A Computer Based Model for the Performance Analysis of a SCRAMJET Propulsion System
by
Chiang-Hwa Ren
B.S.E. Mechanical Engineering and Applied Mechanics, University of Pennsylvania, 1987

SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE OF

Master of Science
in
Aeronautics and Astronautics
at the
Massachusetts Institute of Technology
January 1989
©1989, Chiang-Hwa Ren

The author hereby grants to MIT and the Charles Stark Draper Laboratory, Inc. permission to reproduce
and distribute copies of this thesis document in whole or in part.

Signature of Author ____________________________

Department of Aeronautics and Astronautics
January 1989

Certified by ____________________________
Professor Manuel Martinez-Sanchez, Thesis Supervisor
Department of Aeronautics and Astronautics

Certified by ____________________________
Dr. Phillip D. Hattis, Technical Supervisor
Charles Stark Draper Laboratory

Accepted by ____________________________
Professor Harold Y. Wachman, Chairman
Department Graduate Committee
A Computer Based Model for the Performance Analysis of a SCRAMJET Propulsion System

by

Chiang-Hwa Ren

Submitted to the Department of Aeronautics and Astronautics in partial fulfillment of the requirements for the degree of Master of Science in Aeronautics and Astronautics

This thesis presents a primarily theory based computer oriented two-dimensional model of a SCRAMjet propulsion system developed to analyze the performance and design issues associated with propulsion in the National Aerospace Plane project. The lack of dependence on CFD in this model allows it to rapidly compute performance data for a wide range of design and off-design flight Mach numbers. This is essential in order to reflect the conditions that the actual craft will be expected to fly through. The first run of each test of the model is used as the design condition for establishing the geometry of the propulsion system. Then a series of off-design calculations are performed going from a maximum specified Mach number to a minimum. Data is obtained for conditions in the inlet, combustor, and nozzle. The performance parameters of Isp, thrust coefficient, thrust parameter, and global efficiency are also computed. The key issues of finding the appropriate design Mach number, understanding the effects of flying at different angles of attack, and understanding the effects of varying combustor fuel ratio are addressed in detail.

Thesis Supervisor: Manuel Martinez-Sanchez
Associate Professor of Aeronautics and Astronautics
Acknowledgements

I wish to begin by offering my deepest gratitude to Prof. Martinez-Sanchez for his guidance, support, and great insight, without which this thesis would not be possible. My sincerest thanks also goes to Mark Lewis, Roger Biasca, and the other members of M.I.T.'s hypersonics team for all the invaluable help they have given me during the past year.

To my parents who have given me everything and shared in all my hopes and dreams, I offer my eternal love. My special thanks goes to the rest of my family and my true friends for their continuous support. You mean more to me than all the riches of this world. Finally, my gratitude goes to all the teachers in my life. Knowledge is truly a priceless gift.

This thesis was written under the sponsorship of The Charles Stark Draper Laboratory, Inc. Publication of this thesis does not constitute approval by The Charles Stark Draper Laboratory, Inc. of the findings or conclusions contained herein.
## Contents

<table>
<thead>
<tr>
<th>Acknowledgements</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>1 INTRODUCTION</strong></td>
<td>4</td>
</tr>
<tr>
<td><strong>2 INLET ANALYSIS</strong></td>
<td>6</td>
</tr>
<tr>
<td>2.1 Shock Calculation</td>
<td>7</td>
</tr>
<tr>
<td>2.2 Design Geometry</td>
<td>9</td>
</tr>
<tr>
<td>2.3 Off-Design Conditions</td>
<td>11</td>
</tr>
<tr>
<td>2.4 Boundary Layer Calculation</td>
<td>16</td>
</tr>
<tr>
<td>2.5 Performance of the Inlet</td>
<td>21</td>
</tr>
<tr>
<td><strong>3 COMBUSTOR ANALYSIS</strong></td>
<td>23</td>
</tr>
<tr>
<td>3.1 Chemical Equilibrium</td>
<td>25</td>
</tr>
<tr>
<td>3.1.1 Fuel-Rich Case</td>
<td>26</td>
</tr>
<tr>
<td>3.1.2 Fuel-Lean Case</td>
<td>27</td>
</tr>
<tr>
<td>3.1.3 Stoichiometric Case</td>
<td>28</td>
</tr>
<tr>
<td>3.2 Combustor Geometry</td>
<td>29</td>
</tr>
<tr>
<td>3.3 Inclusion of Boundary Layer Effects</td>
<td>30</td>
</tr>
<tr>
<td><strong>4 NOZZLE ANALYSIS</strong></td>
<td>32</td>
</tr>
<tr>
<td><strong>5 MODELLING AND GLOBAL PARAMETERS</strong></td>
<td>38</td>
</tr>
<tr>
<td><strong>6 RESULTS</strong></td>
<td>42</td>
</tr>
<tr>
<td>6.1 Design Mach Number</td>
<td>42</td>
</tr>
<tr>
<td>6.2 Angle of Attack</td>
<td>55</td>
</tr>
<tr>
<td>6.3 Varying Stoichiometry</td>
<td>61</td>
</tr>
<tr>
<td>6.4 Other Results</td>
<td>65</td>
</tr>
<tr>
<td><strong>7 CONCLUSION</strong></td>
<td>71</td>
</tr>
<tr>
<td>Section</td>
<td>Page</td>
</tr>
<tr>
<td>------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>A  MAIN PROGRAM WITH INLET ANALYSIS</td>
<td>75</td>
</tr>
<tr>
<td>B  COMBUSTOR AND NOZZLE ANALYSIS SUBPROGRAM</td>
<td>95</td>
</tr>
<tr>
<td>C  METHOD OF WAVES SUBPROGRAM</td>
<td>106</td>
</tr>
<tr>
<td>D  SUBROUTINES FOR THE NOZZLE</td>
<td>114</td>
</tr>
<tr>
<td>E  SHOCKS, BOUNDARY LAYER, AND CHEMICAL EQUILIBRIUM ROUTINES</td>
<td>126</td>
</tr>
</tbody>
</table>
Chapter 1
INTRODUCTION

With the revitalization of the field of hypersonics sparked by President Reagan’s announcement in 1986 of the national commitment toward an aerospace plane, it is evident that many areas in hypersonics require substantially more fundamental research and understanding. While always remaining a viable objective, insufficient progress in hypersonic technology has been made since the research in the nineteen sixties which was to the most part halted with the cancellation of the X-plane projects. Therefore, more fundamental research is clearly needed to bring the objective of a hypersonic plane to reality or once and for all prove the impracticality of hypersonic flight in the near future other than by rockets. One key area in determining the possibility and limitations of the proposed hypersonic craft is hypersonic propulsion technology. This is the area in which this thesis focuses.

In achieving an understanding and design of a hypersonic propulsion system, several paths of research can be taken each with its significant limitations. Existing technological limitations in testing facilities such as wind tunnels severely hinders the capability to simulate hypersonic conditions. In terms of wind tunnels, the desired upper end condition of Mach 25 or even just higher than Mach 10 is still basically unachievable. Even conditions near Mach 10 are primarily limited to shock facilities. Without going into great details, it can be said that empirical research in hypersonics is extremely difficult at best. So far, the hypersonics research group at the Massachusetts Institute of Technology has opted to take the theoretical and computational path of research. This is the most effective and rapid way given the conditions in which one can increase one’s fundamental understanding of hypersonic propulsion. This line of research should enhance the more specific and extensive research activities of the U.S. government and private industry. To date, significant studies have been completed by other members of the group in the areas of detailed CFD modelling of the propulsion system, modelling of the chemical kinetic processes in the SCRAMjet, and a thorough study of the problem of inlet flow phenomena.
The research presented in this thesis involves the development of a more general and flexible model, primarily based on theory and computer oriented, of an entire hypersonic propulsion system. This thesis is a result of work built on top of and in conjunction with continuing research initiated by Professor Martinez-Sanchez. By avoiding the computer time consuming approach of CFD, this model achieves a much higher degree of flexibility in terms of the range of conditions that can be analyzed in a limited time frame. By sacrificing certain details, a much greater understanding of the overall behavior and performance of the propulsion system has been achieved. Because of the greater flexibility and speed in terms of computer time of this model, it can be fitted into other models for different areas of study. Most notably, this model can provide part of the data needed for a model analyzing the trajectory and control requirements of the craft. This is the primary interest of our sponsor, Charles Stark Draper Laboratory, Inc.

While the exact design of the hypersonic propulsion system has either not yet been decided or disclosed by the industrial contractors, basic principles do dictate the necessary components of a hypersonic propulsion system. The complete details as to why the following general design is correct is explained quite clearly in other references[1][2]. In concise terms, conventional turbomachinery propulsion systems reach their limitations in the range of Mach 2 to 3. Going to higher Mach numbers, the temperature and forces generated by the inlet air flow exceed the structural and material strength of the blades and other turbine components. Therefore, a RAMjet type system is currently the only real viable option. Basically, the propulsion system is then an air breathing rocket with three general sections; a compression inlet taking in the oxidizer (air), a combustor where the fuel is injected, mixed, and burned, and finally an expansion nozzle. There is really no internal moving parts as in a turbojet. Because the air is driven in by the motion of the craft, a minimum Mach number must be achieved in order for the system to start up. The most important design criterion is that given such a proposed system, the entire forebody of the plane must be used as the inlet, and the aftbody of the craft becomes a part of the external nozzle. The model to be presented analyzes the entire system from the inlet to the combustor, and exiting the nozzle.
Chapter 2
INLET ANALYSIS

The approach to modelling the inlet is to simplify the inlet forebody surface into a two-dimensional model of a two compression ramp. While two dimensional flow is definitely not a perfect representation of the three-dimensional conditions, it does give us a solid estimate of inlet properties and flow behavior with minimal analytical complexity. The hypersonic flow going into this inlet model passes through oblique shocks as it makes the turn at each ramp and entering the inlet. A variable angle of attack for the craft is incorporated by allowing variable turning of the first ramp. The inlet flow is modelled in two regions, an inviscid core flow, and a boundary layer region where the viscous effects are contained.

The geometry of the propulsion system is set for a specific design Mach number. In the case of the inlet, the primary design requirement is shock matching at the lip of the inlet. Once the geometry is fixed, the computation method varies to some degree as one goes to higher or lower than design Mach number, or as one changes the angle of attack, indicating off-design conditions. In the case of the inlet, higher Mach numbers will sweep the shocks into the inlet while shocks at lower Mach numbers will not even reach the lip, thus resulting in some spillage. Changes in $\alpha$ will affect shock conditions also. Increasing $\alpha$ will cause the first shock to be stronger and more swept back. The lowered Mach number flow over the second ramp will cause the second shock to be less swept back. Thus, the increase of $\alpha$ may either lead to the shock coming nearer to or further from the ramp surface. The reverse is true for decreases in the angle of attack. More details and formulations will be given on this after a brief presentation of the shock computation method. Because the inviscid core flow is dominant, the much simpler inviscid shock theory given in many references[3] can be used. The effects of the viscous boundary layer will be included later.
2.1 Shock Calculation

The simplest way to analyze property changes across a shock from region one to region two is to first consider a normal shock. The appropriate equations relating properties across a the normal shock are

\[ \text{Continuity: } \rho_1 u_1 = \rho_2 u_2 \tag{2.1} \]

\[ \text{Momentum: } P_1 - P_2 = (\rho_1 u_1)u_1 - (\rho_2 u_2)u_2 \tag{2.2} \]

\[ \text{Energy: } \frac{\gamma}{\gamma - 1} \frac{P_1}{\rho_1} + \frac{u_1^2}{2} = \frac{\gamma}{\gamma - 1} \frac{P_2}{\rho_2} + \frac{u_2^2}{2} \tag{2.3} \]

Equation (2.3) is a form of the compressible Bernoulli's equation derivable from the energy equation for adiabatic flow of a perfect gas.

\[ \frac{1}{2}(u_1^2 - u_2^2) = C_p(T_2 - T_1) \]

with: \( T = \frac{Pv}{R}, R = C_p - C_v, v = 1/\rho \) Also noting \( M = V/V_a, V_a = \sqrt{\gamma RT}, \frac{P}{\rho} = RT \), this set of equations can be algebraically manipulated to yield the following shock relations:

\[ \frac{u_2}{u_1} = \frac{\rho_1}{\rho_2} = \frac{2 + (\gamma - 1)M_1^2}{(\gamma + 1)M_1^2} \tag{2.4} \]

\[ \frac{P_2}{P_1} = \frac{1 - \gamma + 2\gamma M_1^2}{\gamma + 1} \tag{2.5} \]

\[ \frac{T_2}{T_1} = \frac{P_2\rho_1}{P_1\rho_2} \tag{2.6} \]

Extension to oblique shocks:

With the addition of a turning angle \( \delta \) the shock is swept back to an angle denoted by \( \theta \) so that all the computation can be then treated as if its going across a normal by taking normal components. The goal is then to determine \( \theta \) given \( \delta \).
Figure 2.1: Geometry of a shock

From the geometry $u_1 = u_2 \frac{\cos(\theta - \delta)}{\cos(\theta)}$ Combining this with equation (2.4), the result is then

$$\frac{u_{1n}}{u_{2n}} = \frac{\cos(\theta - \delta) \sin(\theta)}{\cos(\theta) \sin(\theta - \delta)} = \frac{(\gamma + 1) M_1^2 \sin^2 \theta}{2 + (\gamma - 1) M_1^2 \sin^2 \theta}$$

(2.7)

After more algebraic manipulation we arrive at the following form:

$$\tan \delta = \cot \theta \frac{M_1^2 \sin^2 \theta - 1}{1 + M_1^2 \left(\frac{\gamma + 1}{2} - \sin^2 \theta\right)}$$

(2.8)

While an exact solution for $\theta$ in terms of $\delta$ is in theory achievable, the mathematical complexity which includes dealing with real and complex roots makes an iterative solution scheme much more attractive since the range of the correct solution is known. The approach selected is to guess a $\theta$ for a given $\delta$ and then iteratively converge to the right solution. For the initial approximation, the following rough assumptions will be taken:

$$\delta << 1, M_1 \sin \theta \sim 1, \cos \theta \sim \sqrt{1 - \frac{1}{M_1^2}}, \cot \theta \sim \sqrt{M_1^2 - 1}$$

We can then make the following approximations:

$$\theta = \sin^{-1} \frac{1}{M_1} + \epsilon$$

(2.9)

$$\sin \theta = \sin(\sin^{-1} \frac{1}{M_1} + \epsilon) \simeq \frac{1}{M_1} + \epsilon \sqrt{1 - \frac{1}{M_1^2}}$$

(2.10)

$$\cos \theta \simeq \sqrt{1 - \frac{1}{M_1^2}} - \frac{\epsilon}{M_1}$$

(2.11)

$$\cot \theta \simeq \sqrt{M_1^2 - 1 - \epsilon M_1^2}$$

(2.12)
Placing these into equation (2.8) we have:

\[ \tan \delta = \frac{(\sqrt{M_1^2 - 1} - \epsilon M_1^2)(2\epsilon \sqrt{M_1^2 - 1})}{M_1^{2+\frac{1}{2}} - 2\epsilon \sqrt{M_1^2 - 1}} \]  \hspace{1cm} (2.13)

After making all possible small angle approximations and dropping higher order terms we get:

\[ 2\epsilon(M_1^2 - 1) = M_1^2(\gamma + 1) \tan \delta \]  \hspace{1cm} (2.14)

Making the last approximation \( \tan \delta \sim \delta \) we than arrive at a reasonable approximation for \( \theta \) to start the iteration.

\[ \theta = \sin^{-1} \frac{1}{M_1} + \frac{\gamma + 1}{2(M_1^2 - 1)} \delta \]  \hspace{1cm} (2.15)

### 2.2 Design Geometry

With the method to calculate shock angles and changes across the shocks presented, the complete geometry of the inlet can then be obtained. The first calculation should be for design geometry with the condition of shock matching. Once the geometry is fixed then one can continue with calculations at higher or lower than the design Mach value and other changes from the design condition. The formulas for the design condition and the off design conditions all basically come from a control volume approach plus basic geometry. This is more so for the off design conditions.

![Design geometry with shocks matched at the lip](image)

Figure 2.2: Design geometry with shocks matched at the lip

The above diagram shows the geometry at design of the inlet and the control volume.
taken. To start, we note from mass conservation that

$$\frac{CC'}{AA'} = \frac{\rho_0 u_0}{\rho_3 u_3} = \frac{\rho_0 M_0 \sqrt{T_0}}{\rho_3 M_3 \sqrt{T_3}} \quad (2.16)$$

For zero angle of attack, the distance $AA'$ is defined as the design capture height ($Cap$), a design specification. However, as the angle of attack is increased, the flow capture increases. Therefore, the changing $AA'$ is related to the design capture height for $\alpha = 0$ by the following $\alpha$-dependent relations.

$$AA' = Cap \frac{\sin \theta_1}{\sin(\theta_1 - \alpha)} \quad (2.17)$$

For our model, if one designs at a nonzero $\alpha$, then an inlet which is shock matched at that $\alpha$ and satisfying the design capture height when rotated to $\alpha = 0$ will be generated. For most cases the design is done at $\alpha = 0$ and then rotated to the correct $\alpha$. With some basic trigonometry work, the following equations for the design configuration of the inlet can be obtained.

$$A'D = \frac{AA'}{\tan \theta_1} \quad (2.18)$$

$$AB = \frac{AA'}{\sin \theta_2} \left( \frac{\sin(\delta_1 + \alpha + \theta_2)}{\tan \theta_1} - \cos(\delta_1 + \alpha + \theta_2) \right) \quad (2.19)$$

$$BD = \frac{AA'}{\sin \theta_2} \left( \cos(\delta_1 + \alpha) - \frac{\sin(\delta_1 + \alpha)}{\tan \theta_1} \right) \quad (2.20)$$

$$BC = BD \frac{\sin(\pi - \theta_2 - \theta_3 + \delta_2)}{\sin \theta_3} \quad (2.21)$$

To allow flow conditions such as reflected shocks to settle we need to set an adequate duct length. As a design constraint the upper length of the duct will be set to five times the entering height. Page 120 of reference [4] gives information showing that this constraint is reasonable. This constraint primarily shows up in the system diagrams and does not really affect the computed properties except in terms of the boundary layer. Therefore, the lower length of the duct will be

$$\text{lower length} = \text{upper length} + \frac{\text{area ratio} AA'}{\tan(\theta_3 - \delta_1 - \delta_2)} \quad (2.22)$$

The angles of these panels that form the duct are already known.

Thus, the geometry of the inlet can be fixed with a set of design calculations. The pressure, temperature, and Mach ratios from region (0) to (3) are simply the products of the ratios across the three shocks. With ambient conditions given, the combustor entering flow conditions at the end of the duct can be computed.
This inviscid geometry must then be modified by using approximations to the viscous effects. For the purpose of computation the exact boundary layer shape will be approximated by a linear growth. Thus, properties such as boundary layer thickness, displacement thickness, and momentum thickness need only be computed at the end of each panel. The computational methods for both laminar and turbulent boundary layers will be presented in detail later. For now, we proceed with how the thicknesses, once computed, are incorporated into the model.

For the design case, the inviscid calculation provides the external fluid conditions on each panel, which are then used to compute boundary layer thicknesses. The panels are then pushed back to incorporate the displacement thickness while not affecting the geometry of the inviscid core. With this adjustment the design geometry is complete and fixed.

For the off-design conditions the process of incorporating the boundary layer effects will have to be iterative since the shape of the inlet can not be altered. The iteration involves performing the new shock calculations after approximating the boundary layer and adjusting the inviscid flow turn angle to incorporate the new boundary layer thicknesses. The boundary layer is then recomputed and the iteration continues until convergence.

### 2.3 Off-Design Conditions

For off-design conditions slightly more complex control volume approach methods are needed. This implies two conditions, one where the shocks do not reach the lip (Condition 2), and the opposite where the shocks are swept into the inlet (Condition 3). A special case of Condition 3 can occur if only the 2nd shock is swept into the inlet. The diagrams below show the the control volumes used for each off-design shock condition. Note that in the control volume equations the effects of the boundary layer presented earlier are included.
To start the off design analysis we define the quantities $F$ (stream impulse at the duct exit), $G$ (mass flux at duct exit), and $H$ (core total enthalpy). The following definitions
apply to all cases:

\[ F = (P + \rho u^2)_{\text{exit}} \]  
\[ G = (\rho u)_{\text{exit}} \]  
\[ H = C_p T_o + u_o^2/2 = (C_p T + u^2/2)_{\text{exit}} \]  

We can then solve for the exit conditions in terms of F, G, H:

\[ u = \frac{\gamma F}{\gamma + 1 G} + \sqrt{\frac{\gamma F}{\gamma + 1 G}^2 - 2\frac{\gamma - 1}{\gamma + 1} H} \]  
\[ T = \frac{1}{R G} \]  
\[ P = F - G u \]  

At this point we introduce the use of the following nondimensional variables:

\[ \rho' = \frac{\rho}{\rho_0} \]  
\[ u' = \frac{u}{u_0} \]  
\[ P' = \frac{P}{\rho_0 u_0^2} \]  
\[ T' = \frac{C_p T}{u_0^2} \]  

Once the exit conditions are found in terms of the above variables, the actual initial to final property ratios can obtained as follows:

\[ \frac{P_e}{P_0} = \gamma \frac{M_o^2}{P'} \]  
\[ \frac{T_e}{T_0} = (\gamma - 1) \frac{M_o^2}{T'} \]  
\[ \frac{\rho_e}{\rho_0} = \rho'_e \]  
\[ \frac{u_e}{u_0} = u'_e \]  
\[ \text{Mach}_e = \frac{u'_e}{\sqrt{(\gamma - 1)T'_e}} \]  

The next step is to then to determine the specific G,F,H for each condition. With the definition of C_p T it can be realized quickly that H is the same for all conditions.

\[ H' = \frac{H}{u_0^2} = \frac{1}{2} + \frac{1}{(\gamma - 1)M_o^2} \]
For $G$ we note that there is flow spillage when the shock does not reach the lip. Thus the flow capture ratio $C_r$ needs to be computed:

$$G' = \frac{\rho u}{\rho_0 u_0}$$

Applying mass conservation:

$$G' = \frac{AA' C_r}{A_{\text{exit}}}$$

(2.35)

Note that $C_r=1$ when both shocks reach the lip. Therefore, the capture ratio is only significant in computing $G$ for Condition 2 and the Special Case of Condition 3.

To find $F$ one needs to perform a force balance in the $X$ direction on the control volumes. Before we proceed with the full equations, the balance can be performed on the duct only and such a term will be applicable in all the conditions since every condition's control volume incorporates the duct region.

$$F_{\text{duct}} = P_e \gamma M_e^2 \cos \alpha \left[ \frac{L(3) \sin(\phi_3) + L(4) \sin(\phi_4)}{\gamma M_e^2} - (\theta_4 + (\theta_3 - \theta_30)) \right]$$

All the $\theta$s and $\delta$s in this section refers to momentum and displacement thicknesses. The numbers associated with them represent the wall panel to which that boundary layer parameter applies. Panels 1, and 2 represent the two ramps and panels 3 and 4 represent the upper and lower duct surfaces. The lengths $L(3)$ and $L(4)$ indicate the upper and lower duct lengths and the angles $\phi$ indicate the wall displacements due to boundary layer. We now proceed with specific cases.

In the case of Condition 2 the balance of $X$ directional forces gives:

$$P_2 Y_{DJ} - P_4 Y_{JC} - P_4 L(3) \sin \alpha + P_4 L(4) \sin \alpha + F_{\text{duct}} - P_E Y_E \cos \alpha =$$

$$\rho_e u_e^2 (Y_E - \delta_3 - \delta_4 - \theta_3 - \theta_4) \cos \alpha - \rho_2 u_2 u_\infty (D\text{JN} - \delta_2 - \theta_2)$$

The numbers associated with the pressures($P$) indicate the region which those pressures came from, and the $Y$'s represent the $y$ distance between the two points such as from point D to point J. DJN stands for the distance of a line normal to the second panel and intersecting the point D. A unit depth is imposed in the calculation. Please refer to Figure(2.3) for where each term in the above equation comes from. By extracting the appropriate terms to fit the definition of $F$ we get:

$$F = \frac{1}{Y_E \cos \alpha} \left( \frac{1}{\gamma M_0^2} \left( \frac{P_2}{P_0} Y_{DJ} - \frac{P_4}{P_0} Y_{JC} + \frac{P_4 \sin \alpha}{P_0} (L(4) - L(3)) \right) + \frac{F_{\text{duct}}}{\rho_0 u_0^2} \right)$$

$$+ \frac{P_e M_e^2}{P_0 M_0^2} (\delta_3 + \delta_4 + \theta_3 + \theta_4) \cos \alpha + \frac{\rho_2 u_2 u_\infty}{\rho_0 u_0 u_0} (D\text{JN} - \delta_2 - \theta_2)$$

(2.36)
It should be noted that since $P_E$ is in the definition for $F$ and $P_E$ is what we want to finally determine, the solution will have to be an iterative one.

For Condition 3 the the force balance equation becomes:

\[
\begin{align*}
P_0 Y_{AD} - P_1 Y_{AB} - P_2 Y_{BC} - P_E L(3) \sin \alpha + & P_E L(4) \sin \alpha - \rho_1 u_1^2 \theta_1 \cos(\delta_1 + \alpha) \\
- \rho_2 u_2^2 (\theta_2 - \theta_1) \cos(\delta_1 + \alpha + \delta_2) = & \rho_E u_E^2 (Y_E - \delta_3 - \delta_4 - \theta_3 - \theta_4) \cos \alpha - \rho_0 u_0^2 Y_{AD}
\end{align*}
\]

Figure (2.4) helps to indicate where each force component in the above equation applies in the control volume. This then gives $F$ as:

\[
F = \frac{1}{Y_E \cos \alpha} (Y_{AD} + \frac{1}{\gamma M_0^2} (Y_{AD} - \frac{P_1}{P_0} Y_{AB} - \frac{P_2}{P_0} Y_{BC} + \rho_E \sin \alpha (L(4) - L(3))) + \frac{F_{duct}}{\rho_0 u_0^2} \]

\[
\frac{P_E M_0^2}{P_0 M_0^2} (\delta_3 + \delta_4 + \theta_3 + \theta_4) \cos \alpha - \rho_1 u_1^2 \frac{1}{\rho_0 u_0^2} \theta_1 \cos(\delta_1 + \alpha) - \frac{\rho_2 u_2^2}{\rho_0 u_0^2} (\theta_2 - \theta_1) \cos(\delta_1 + \alpha + \delta_2)) (2.37)
\]

In the special case of Condition 3 we note that the inlet flow capture is not complete. Therefore, $Y_{AD}$ must be multiplied by the capture ratio. Also, from the control volume, region (1) has an additional force component other than from the first ramp wall. This extra force from the additive drag region shown in figure(2.5) should be incorporated into the above Condition 3 equations.

The capture ratio is defined as the percentage of flow crossing $AA'$ that actually enters the duct. This can either be computed by mass conservation between the flow in the duct and the flow crossing $AA'$ in region(0), or by geometrically extending the flow path from the duct to $AA'$ as shown in the diagrams. With the key properties of pressure, temperature, density, velocity, and Mach number entering the combustor totally defined for all the conditions, other properties and computations of interest can now be presented.

One condition of interest that usually occurs in Condition 3 is that the shocks of the same family cross each other. While the control volume presented incorporates all these effects, it might be useful from a geometrical stand point to see what results. When shocks of the same family cross, they merge to form a stronger shock where the property changes across that shock reflect the change across the original two shocks. Since we already know the properties at region zero and two, the angle of the resulting merged shock can be easily computed by working back from equation 2.5.

\[
\theta = \sin \left( \frac{\sqrt{\frac{P_0}{\gamma} (\gamma + 1) -(1 - \gamma)}}{\sqrt{2 \gamma M_0}} \right) (2.38)
\]

When shocks do not reach the lip as in Condition 2 we note from the geometry that the flow pressure changes add drag to the body. This shock induced drag can easily be
expressed in terms of an additive drag coefficient:

\[ C_{D_{\text{add}}} = \frac{\text{Drag}}{\frac{1}{2}\rho_0 u_0^2 AA'} = \frac{2}{\gamma M_0^2} \left[ \left( \frac{P_1}{P_0} - 1 \right) Y_{PQ} + \left( \frac{P_2}{P_0} - 1 \right) Y_{QD} \right] \] (2.39)

This drag is significant in that it is not induced by the actual airframe and will be considered as a reduction in thrust.

### 2.4 Boundary Layer Calculation

The control volume analysis presented earlier incorporates the effects of the compressible hypersonic boundary layer in terms of certain layer parameters. This section is then devoted to explaining how the boundary layer parameters used to formulate the approximate linear boundary layers are computed.

The compressible laminar boundary layer region can be analyzed with the use of well known integral variables and a form of the Crocco-Busemann temperature relationship[5]. For compressible flow:

The displacement thickness is defined as:

\[ \delta^* = \int_0^\infty \left( 1 - \frac{\rho u}{\rho_e u_e} \right) dy \] (2.40)

Similarly we have:

Momentum thickness:

\[ \theta = \int_0^\infty \frac{\rho u}{\rho_e u_e} (1 - \frac{u}{u_e}) dy \] (2.41)

Boundary Layer Thickness:

\[ \delta = \int_0^{\delta_{99\%}} dy \] (2.42)

Shape Factor:

\[ H = \frac{\delta^*}{\theta} \] (2.43)

Skin Friction Coefficient

\[ C_f = \frac{2 \tau_w}{\rho_e u_e^2} \approx \frac{0.664 \sqrt{C^*}}{\sqrt{Re_{xe}}} \] (2.44)

\[ Re_{xe} = \frac{\rho_e u_e x}{\mu_e} \]

\( C^* \) in the above equation is the average value of the Chapman-Rubesin parameter[5]. It should be noted that most of the resulting equations, especially as we discuss turbulent flows, are semiempirical in nature.
The Crocco Relations states that the profiles of the velocity and excess total enthalpy are congruent. Expressed in term of \( f' = \frac{\rho}{\rho_e} \), it can be written as

\[
\frac{\rho}{\rho_e} = \frac{T}{T_e} = \frac{T_w}{T_e} + \frac{T_{aw} - T_w}{T_e} f' - \frac{\gamma - 1}{2} M_e^2 f'^2
\]  

(2.45)

\[T_{aw} = T_e + \frac{U_e^2}{2 C_p}\]

This analysis can be made a great deal simpler with the application of the following similarity variables for laminar flat plate boundary layers. The detailed theory behind them is presented in chapter 7 of reference[5].

\[\xi = \rho_e U_e \mu_e x\]

(2.46)

\[\eta = \left(\frac{Re_{ex}}{2}\right)^{1/2} \int_0^x \frac{\rho}{\rho_e} \frac{dy}{x} = \frac{U_e}{\sqrt{2 \xi}} \int_0^x \rho dy\]

(2.47)

Also, the following nondimensionalized variables will be used.

\[\beta = 1 + \frac{\gamma - 1}{2} M^2\]

(2.48)

\[t_w = \frac{T_w}{T_e}\]

(2.49)

\[Re_{\delta e} = \frac{\rho_e U_e \delta}{\mu_e}\]

(2.50)

Equation (2.45) with the above nondimensional variables becomes:

\[
\frac{\rho}{\rho_e} = t_w + (\beta - t_w)f' - (\beta - 1)f'^2
\]

(2.51)

From equation (2.47) we obtain the relationship:

\[dy = \sqrt{\frac{2}{Re_{ex}}} \frac{\rho_e}{\rho} d\eta\]

(2.52)

Substituting the above in equation (2.40) for displacement thickness we have the new equation:

\[
\frac{\delta^*}{x} = \frac{2}{Re_{ex}} \int_0^\infty [t_w + (\beta - t_w - 1)f' - (\beta - 1)f'^2] d\eta
\]

(2.53)

Finally we relate our variables as \( f'(<\eta) \approx f_B'(<\eta_B \sqrt{C^*}) \) and \( \eta = \eta_B \sqrt{C^*} \). Referring to tabular form of the Blasius solution, \( u = 0.99 U_e \) occurs at \( \eta_B = 3.6 \) where \( f' \), \( f'' \), and \( f''' \) are 2.385, 0.9939, and 0.0193 respectively. This then gives after integration

\[
\frac{\delta^*}{x} = \sqrt{\frac{2C^*}{Re_{ex}}} (1.215 t_w + 0.467 (\beta - 1))
\]

(2.54)
By the power law for viscosity in the case of air $C^* \approx (\frac{T_e}{T_*})^{-1/3}$. An empirical estimate for the temperature ratio was presented by Eckert[5] as

$$\frac{T_*}{T_e} \approx 0.5 + 0.5t_w + 0.195(\beta - 1)$$

This approach when applied to the other boundary layer parameters yields

$$\frac{\theta}{x} = \sqrt{\frac{2}{Re_{se}}} \int_0^\infty [f' - f'^2] d\eta$$

(2.55)

$$\frac{\theta}{x} = \sqrt{\frac{2C^*}{Re_{se}}} 0.467$$

$$\frac{\delta}{x} = \sqrt{\frac{2}{Re_{se}}} \int_0^\infty [t_w + (\beta - t_w)f' - (\beta - 1)f'^2] d\eta$$

(2.56)

$$\frac{\delta}{x} = \sqrt{\frac{2C^*}{Re_{se}}} (2.385 + 0.467(\beta - 1) + 1.215t_w)$$

These results allow us to quick approximate laminar boundary layer thickness parameters as a function of plate distance.

As discussed in previous sections, the shape of the boundary layer is approximated by a straight line from an initial to a final thickness across the plate. An interesting effect is that the boundary layer is compressed as it goes across a shock. This, as the results will later show, significantly limits the growth of the boundary layer in our multiramp inlet design. The calculation of final thickness on a plate with initial layer thickness involves extending the length of the plate back with the initial thickness to where such a layer would begin and then computing forward to find the thickness at the end of this modified plate.

The analysis of turbulent boundary layers effects poses many interesting difficulties. First of all, the question of whether transition occurs and if it does where it occurs is an area in which great uncertainty exists. While the exact fixing of the point of turbulent transition is far beyond the scope of this research, the possibility of turbulent flow effects must be incorporated into the model as an option.

The effective velocity approach presented by van Driest produces useful relationships between $Re_{se}, Re_{de}$, and $C_f[6]$. If we introduce the following parameters

$$a = \sqrt{\frac{\beta - 1}{t_w}}$$

$$b = \frac{\beta}{t_w} - 1$$
\[
A = \frac{2a^2 - b}{(b^2 + 4a^2)^{1/2}} \\
B = \frac{b}{(b^2 + 4a^2)^{1/2}}
\]

with \(\beta\) and \(tw\) the same as that for the laminar boundary layer calculations, then we have the following relationships:

\[
\frac{\sin^{-1} A + \sin^{-1} B}{\sqrt{C_f(\beta - 1)}} \approx 4.15 \log(Re_{\infty} C_f \frac{\mu_e}{\mu_w}) + 1.7 \tag{2.57}
\]

for air \(\frac{\mu_e}{\mu_w} \sim \frac{1}{tw^{0.87}}\)

\[
\frac{\sin^{-1} A + \sin^{-1} B}{a\sqrt{C_f \frac{tw}{2}}} \approx 5.6 \log(Re_{\delta_e} \frac{\sqrt{C_f/2}}{\frac{\delta_e}{1/6}}) + 7.4 \tag{2.58}
\]

Thus with the length of the plate and \(Re_{\infty}, C_f\) can be computed by a simple iteration. This then leads to an approximation for \(Re_{\delta_e}\) and \(\delta\). The other thicknesses can then be determined with the approximations:

\[
\frac{\delta^*}{\delta} \approx 0.34 \log(tw + \beta)[1 - 0.11 \log(\frac{Re_{\delta_e}}{10^5})] \tag{2.59}
\]

\[
H = \frac{\delta^*}{\theta} \approx tw + \theta \tag{2.60}
\]

To check the validity of this analysis, figures(2.6 and 2.7) offers a comparison of the skin friction coefficient computed by the above method with that of other theory based and experimental results. For figure(2.7) the value of \(C_f(\text{incompressible})\) is \(C_f(\text{incompressible}) = 0.025 Re^{-1/7}\) [5]. In Wilson’s results, \(tw\) varies from 1.8 at Mach 2 to 21.0 at Mach 10. For our calculation a temperature ratio of \(tw = 1 + \frac{2-1}{2} \text{Mach}^2\) was used.

19
Figure 2.6: Skin friction coefficient for compressible turbulent boundary layer over a zero incidence flat plate with $Re_x \approx 10^7$

[7]

Figure 2.7: Skin friction coefficient computed by the method described
2.5 Performance of the Inlet

With the boundary layer analysis sketched out, the inlet performance parameter called kinetic energy efficiency can be introduced, first with the inviscid definition and then incorporating once again the effect of the boundary layer. First for the inviscid case, the definition of kinetic energy efficiency is

$$\eta_k = \frac{K.E.\text{after ideal re-expansion}}{K.E.\text{original}}$$

For ideal gas with uniform losses, the following expression can be derived [8].

$$\eta_k = \frac{T_t}{T_{t0}} \left[ 1 - \frac{(P_{t0})^{(\gamma-1)/\gamma} - 1}{\frac{\gamma-1}{2} M_o^2} \right]$$ (2.61)

However, $P_t$ and $T_t$, and hence $\eta_k$ varies across the boundary layer, so a better indicator is an averaged value defined as

$$\langle \eta_k \rangle = \frac{\int \eta_k drn}{\int drn}$$

$drn = \rho dy$ multiplied by a constant depth. After substantial manipulation of the above definitions we get the result:

$$\langle \eta_k \rangle = \eta_k - \frac{\beta \delta_H - [1 + (\beta - 1)(1 - \eta_k)] \delta_T}{(\beta - 1)(H - \delta^*)}$$ (2.62)

$$\delta_H = \int_0^\delta \frac{\rho u}{\rho u_e} (1 - \frac{ht}{ht_e}) dy \approx (1 - \frac{T_w}{T_{te}}) \theta$$ (2.63)

$$\delta_T = \int_0^\delta \frac{\rho u}{\rho u_e} (\frac{T}{T_e} - 1) dy \approx \delta^* - \delta_u$$ (2.64)

Following the effective velocity approach for turbulent boundary layer, 

$$\delta_u = \int_0^\delta (1 - \frac{u}{u_e}) dy \approx \frac{2.9744C_f((\beta + tw)/2 - 1) + 2.439\sqrt{C_f/2}}{1 + 5.9488(\beta - 1)(C_f/2)}$$

For laminar flow the term is very simplified if the approximation $u/u_\delta = y/\delta$ is used:

$$\delta_u = \frac{\delta}{2}$$

This basically completes a 2D method for analyzing the inlet. While this inlet model may be considered in many ways crude, it does fulfill a key requirement of rapidly providing information to move on the combustor analysis, while incorporating a wide variety of important effects. The results of this method do agree with results from other simpler, similar, and finer analysis [8].
As a possible alternative to the described shock based inlet analysis, we might consider the application of the newtonian flow theory [9]. While this theory is physically inaccurate, for hypersonic flow where the shocks are very swept back the results may be acceptable. In such a method the pressure rise at the inlet is caused by the flow hitting the ramps. By the inlet geometry, two approaches can be taken. First, we can assume the flow hits the second ramp directly, turns with the new pressure, hits the lip, and then goes into the combustor. Also, we can assume the flow hits the first ramp, turns with the new pressure, hits the second ramp, and then proceeds to hit the lip as before. The primary equation use in newtonian flow analysis is:

\[ P = \lambda \sin^2(\theta) \frac{1}{2} \rho_e U_e^2 + P_e \]

\[ \lambda = \gamma + 1 \]

\( \theta \) in the equation represents the angle of the plate that the flow hits. A comparison between the newtonian flow approach and our shock based approach will be presented later in the results chapter.
Chapter 3

COMBUSTOR ANALYSIS

The supersonic combustion process in a SCRAMjet is an area in which much research is still needed. The exact chemistry that takes place in the combustion chamber is still not perfectly understood even though progress is continuously being made. Reference [10] documents a member of M.I.T.’s group’s contribution to this area. Problems such as the mixing of injected fuel have not been completely resolved.

However, the goal in this model is not to resolve all the combustion problem but instead to find ways to not let these problems bog down the real goal of analyzing the entire propulsion system. Toward this goal, we once again turn to the control volume approach of looking only at entering and exiting conditions. The process in between is assumed to work even though the exact details of how it works are vague and will not be dealt with in this text. This nevertheless still gives us a good prediction of the behavior and conditions to fit into the rest of the model.

The key condition that makes the following analytical approach possible is that the pressure distribution in the combustor can be generalized to relate to the cross sectional area of the combustor by the parametric expression presented by Crocco [8]:

\[
\frac{P_2}{P_1} = \left(\frac{A_1}{A_2}\right)^{\frac{\gamma - 1}{\gamma}}
\]

(3.1)

where \(\epsilon\) is a constant to be determined. From this one can then proceed with the routine conservation of mass and momentum analysis. The mass flow into the combustor is in two parts, the air(oxydizer) taken in, and the fuel injected. Because of the very high rate at which this process must take place, hydrogen has been generally excepted as the only real candidate in terms of fuel. Naturally, the exiting flow will be the sum of these two entering flows. A basic force balance yields:

\[
(m_a + m_f)u_a - m_a u_a - m_f u_f = P_1 A_1 - P_2 A_2 + \int_{\text{exit}} P dA = - \int_{\text{enter}} Adp
\]

(3.2)
Applying the Crocco relation to the integral we get

\[ \int_{\text{exit}}^{\text{exit}} Adp = -\varepsilon P_1 A_1 [1 - \left( \frac{A_1}{A_2} \right) \gamma] \quad (3.3) \]

A control factor that must be given is the mass ratio of the oxydizer to the fuel, represented as OF. From this an expression for \( u_e \) can then be obtained in terms of the (still unknown) \( \varepsilon \).

\[ \dot{m}_a = \left( \frac{O_F}{O_F + 1} \right) \dot{m} \]

\[ u_e = \frac{O_F u_a + u_f}{O_F + 1} + \varepsilon \left( \frac{P_1 A_1}{\dot{m} u_a} \right) u_a [1 - \left( \frac{A_1}{A_2} \right) \gamma] \quad (3.4) \]

\[ \frac{P_1 A_1}{\dot{m} u_a} = \frac{O_F}{(O_F + 1) \gamma M_a^2} \left( \frac{A_1}{A_1 f} \right) \]

\[ \frac{A_1}{A_1 f} = 1 + \frac{M_a T_1 f u_1 f}{O_F M_f T_1 a u_f} \]

Before proceeding, it is necessary to presented some assumptions about the basic chemistry. Although, hydrogen is the preferred fuel, this analysis allows the option to consider a variation of fuel types and mixtures. The primary elements in the oxydizer are oxygen and nitrogen. The primary possible elements in the fuel are carbon, hydrogen, nitrogen, and oxygen. In a stoichiometric condition, reaction (if it was complete) would occur as follows:

\[ \alpha t(C H_{\nu H} N_{\nu N F} O_{\nu O F}) + b_{st}(O_{\nu O O} N_{\nu NO}) \rightarrow \alpha v C O_2 + \frac{\alpha v H}{2} H_2 O + \frac{1}{2}(\alpha v_{N F} + b v_{N O}) N_2 \quad (3.5) \]

For this condition, it is clear that all the oxygen, carbon, and hydrogen are used and that by summing all the amounts of oxygen on both sides the following mole ratio can be established, with its corresponding mass ratio.

\[ \frac{b}{a}_{st} = (2 \nu C + \frac{1}{2} \nu H - \nu_{O F})/\nu_{O O} \quad (3.6) \]

\[ O F_{st} = \left( \frac{b}{a} \right)_{st} \frac{16 \nu_{O O} + 14 \nu_{N O} \text{(molecular weight of ox)}}{12 \nu C + \nu H + 14 \nu_{N F} + 16 \nu_{O F} \text{(molecular weight of fuel)}} \quad (3.7) \]

In a similar manner for nonstoichiometric case, (b/a) can be found once OF is specified.

The next step after expressing \( u_e \) and \( P_e \) in terms of \( \varepsilon \) is to determine the enthalpy at the exit through a total energy balance. The entering conditions are known. One way to make the analysis simpler is to normalize everything with respect to hydrogen. Thus, the total amount of hydrogen \( N_H = a \nu H = 1 \). Scaling the amount of other elements to this, one gets the following:

\[ N_C = \frac{\nu C}{\nu H} \]
\[ N_O = \frac{\nu_{OF} + \frac{1}{2} \nu_{OO}}{\nu_H} \]
\[ N_N = \frac{\nu_{NF} + \frac{1}{2} \nu_{NO}}{\nu_H} \]

Energy balance:
\[
a\nu_H (h_{exit} + \frac{1}{2} M_R \frac{U_e^2}{4180} - M_R h_{loss}) = a (h_f + \frac{1}{2} (0.002) \frac{U_e^2}{4180}) + b (h_{oz} + \frac{1}{2} (0.0289) \frac{U_e^2}{4180}) \quad (3.8)
\]

Please note that all the units are converted to Kcal/mole. \( M_R \) represents kilograms of the products per mole of hydrogen, and is obtainable from the proportion of each basic element. While the proportion of each basic element with \( N_H = 1 \) is given above, what is still not known is the arrangement of the basic elements at the end of the reaction, and hence the static enthalpy \( h_{exit} \) of the products for the given \( P_e \) and \( T_e \). Chemical equilibrium analysis is required for finding the moles of each resulting species at the end of the reaction. For a specific exit temperature there is corresponding exit composition and exit enthalpy. Therefore, we need to iterate in temperature and use chemical equilibrium until the enthalpy matches the correct value resulting from the energy balance.

### 3.1 Chemical Equilibrium

Building from the brief description of the chemical process taking place in the combustion, six types of reaction are postulated at equilibrium [11][12]:

1. \( \frac{1}{2} O_2 + \frac{1}{2} H_2 \rightleftharpoons OH \)
2. \( \frac{1}{2} H_2 \rightleftharpoons H \)
3. \( \frac{1}{2} O_2 \rightleftharpoons O \)
4. \( CO_2 + H_2 \rightleftharpoons CO + H_2O \)
5. \( \frac{1}{2} N_2 + \frac{1}{2} O_2 \rightleftharpoons NO \)
6. \( H_2 + \frac{1}{2} O_2 \rightleftharpoons H_2O \)

The unknowns are the total number of moles after the reaction \( n_{tot} \) and the number of moles of each resulting species \( n_{species} \). For a given temperature the equilibrium constants \( K_p \) for the reactions presented above is known. A new constant in terms of moles instead of partial pressure can then be defined from \( K_p \).

\[
R = \frac{n^\mu_M n^\nu_N}{n^\alpha_A n^\beta_B} = (\frac{n_{tot}}{P})^\mu + \nu - \alpha - \beta K_p
\]

in a reaction \( \alpha A + \beta B \rightleftharpoons \mu M + \nu N \).
It is important to note for later analysis that for reactions 1, 4 and 5 $K_p=R$. Thus, there is no dependence on the unknown $n_{tot}$ in this case.

From all this we can define six $R$:

\[
\begin{align*}
R_1 &= \frac{n_{OH}}{n_{H_2}^{1/2}n_{O_2}^{1/2}} \\
R_2 &= \frac{n_{H}}{n_{H_2}^{1/2}} \\
R_3 &= \frac{n_{O}}{n_{O_2}^{1/2}} \\
R_4 &= \frac{n_{CO}n_{H_2}O}{n_{CO_2}H_2} \\
R_5 &= \frac{n_{NO}}{n_{N_2}^{1/2}n_{O_2}^{1/2}} \\
R_6 &= \frac{n_{H_2}O}{n_{H_2}n_{O_2}^{1/2}}
\end{align*}
\]

The total quantity of each of the four basic elements which was computed earlier can now be expressed in moles of each species as

\[
\begin{align*}
N_H &= 2n_{H_2} + 2n_{H_2}O + n_{OH} + n_H \\
N_O &= n_{H_2}O + 2n_{CO_2} + n_{CO} + n_{OH} + n_O + n_{NO} + 2n_{O_2} \\
N_C &= n_{CO_2} + n_{CO} \\
N_N &= n_{NO} + 2n_{N_2}
\end{align*}
\]

The next phase of the analysis involves a scheme which iteratively converges to the correct values of $n_{species}$. This scheme varies slightly between the three known fuel conditions to take advantage of what is known about each condition.

3.1.1 Fuel-Rich Case

For this case we note that there may be a measurable quantity of unused hydrogen and not very much unused oxygen. Since $n_{tot}$ is not known, the first step is to approximate the quantity of the dominant resulting species through a scheme which is independent of $n_{tot}$. Therefore, the first set of approximations will involve $N_2, H_2, H_2O, CO_2, CO$

Using the definition presented earlier we can achieve the following derivations

\[
G1 = n_{H_2}O + n_{CO_2} = N_O - N_C - 2n_{O_2} - n_{OH} - n_O - n_{NO}
\]
\[ G2 = n_{H2} + n_{H2O} = (N_H - n_{OH} - n_H)/2 \]
\[ Kp4 = \frac{n_{CO}n_{H2O}}{n_{CO2}n_{H2}} \]
\[ 0 = n_{H2O}^2 + n_{H2O} \left( -Kp4(G2 + G1) - N_C + G1 \right) + \frac{Kp4G1G2}{Kp4 - 1} \]

\( n_{H2O} \) is the solution of the quadratic equation and the other quantities can then be easily determined.

\[ n_{N2} = (N_N - n_{NO})/2 \]
\[ n_{H2} = G2 - n_{H2O} \]
\[ n_{CO2} = \frac{N_C}{1 + Kp4 \frac{n_{H2}}{n_{H2O}}} \]
\[ n_{CO} = N_C - n_{CO2} \]

These new values are then used to get a better estimation for \( n_{tot} \). The next set of approximations involve the use of the R's (converted from Kp's):

\[ n_{O2} = (R_6 n_{H2O}/n_{H2})^2 \]
\[ n_{OH} = R_1 n_{H2}^{1/2} n_{O2}^{1/2} \]
\[ n_{H} = R_2 n_{H2}^{1/2} \]
\[ n_{O} = R_3 n_{O2}^{1/2} \]
\[ n_{NO} = R_5 n_{N2}^{1/2} n_{O2}^{1/2} \]

Again, we improve at this point the estimation for \( n_{tot} \). This entire process is repeated until every quantity converges.

3.1.2 Fuel-Lean Case

In this case the excess component is expected to be oxygen. Once again we start with the set of primary species, in this case, \( CO_2, H_2O, O_2, N_2 \). The mole quantity of this group can be quite simply expressed in terms of those of the other compounds, using atom conservation:

\[ n_{CO2} = N_C - n_{CO} \]
\[ n_{H_2O} = \frac{(N_H - n_H - 2n_{H_2} - n_{OH})}{2} \]
\[ n_{O_2} = \frac{(N_O - n_{CO} - n_O - n_{OH} - n_{NO} - 2n_{CO_2} - n_{H_2O})}{2} \]
\[ n_{N_2} = \frac{(N_N - n_{NO})}{2} \]

With these new quantities, a better estimate of \( n_{tot} \) can be achieved and the R's can then be converted from Kp's. The rest of the quantities then follows:

\[ n_{H_2} = R_6 \frac{n_{H_2O}}{n_{O_2}} \]
\[ n_{CO} = R_4 \frac{n_{CO_2} n_{H_2}}{n_{H_2O}} \]
\[ n_{OH} = R_1 n_{H_2}^{1/2} n_{O_2}^{1/2} \]
\[ n_{H} = R_2 n_{H_2}^{1/2} \]
\[ n_{O} = R_3 n_{O_2}^{1/2} \]
\[ n_{NO} = R_5 n_{N_2}^{1/2} n_{O_2}^{1/2} \]

\( n_{tot} \) is recomputed with the new values, and we then iterate until convergence is achieved.

### 3.1.3 Stiochiometric Case

The primary resulting species are \( CO_2, H_2, H_2O, N_2 \). For this case we need to get a crude approximation of \( n_{tot} \) and expressions of the above species in terms of other remaining species.

\[ n_{tot} = n_{CO_2} + n_{H_2} + n_{H_2O} + n_{N_2} \]

\[ n_{tot} = (N_C - n_{CO}) + (N_H - n_H - n_{OH})/2 + (N_N - n_{NO})/2 \]

\( R_6 \) is then obtained from Kp6:

\[ n_{H_2O} + n_{H_2} = \frac{NN_H}{2} = N_H - n_H - n_{OH} \]
\[ n_{CO_2} = NN_C = N_C - n_{CO} \]
\[ n_{H_2O} + 2n_{CO_2} + 2n_{O_2} = NN_O = N_O - n_{CO} - n_O - n_{OH} - n_{NO} \]
\[ 2n_{N_2} = NN_N = N_N - n_{NO} \]
\[ n_{O_2} = \frac{n_{H_2O}^2 R_6^2}{n_{H_2}^2} \]
Using the above expressions, one can derive a cubic equation for \( n_{H_2} \).

\[
\left( \frac{NN_H^2}{2} \right) - (2NN_H)n_{H_2} + \left( \frac{1}{R_6^2} (2NN_C - NN_O + \frac{NN_H}{2}) + 2 \right)n_{H_2}^2 - \frac{1}{R_6^2} n_{H_2}^3 = 0
\]

The rest of the dominant species can then be easily determined:

\[
\begin{align*}
n_{H_2O} & = \frac{NN_H}{2} - n_{H_2} \\
n_{CO_2} & = NN_C \\
n_{CO} & = Kp4 \frac{n_{CO_2}n_{H_2}}{n_{H_2O}} \\
n_{O_2} & = (NN_O - n_{H_2O} - 2n_{CO_2})/2 \\
n_{N_2} & = \frac{NN_C}{2}
\end{align*}
\]

\( n_{tot} \) is estimated as in the other fuel cases in order to convert the Kp's to R's before continuing to the remaining set of computations.

\[
\begin{align*}
n_{OH} & = R_1 n_{H_2}^{1/2} n_{O_2}^{1/2} \\
n_{H} & = R_2 n_{H_2}^{1/2} \\
n_{O} & = R_3 n_{O_2}^{1/2} \\
n_{NO} & = R_5 n_{N_2}^{1/2} n_{O_2}^{1/2}
\end{align*}
\]

As in other cases, we need to improve the estimate of \( n_{tot} \) after each step and iterate until convergence is achieved.

Once the composition at the exit is determined for a give temperature, the total enthalpy which is the sum of the enthalpy of each specie times its number of mole can be easily found. This chemical equilibrium scheme must then be used iteratively with varying temperature until the correct enthalpy is converged upon.

### 3.2 Combustor Geometry

What is not yet known is \( \epsilon \). Its value must be iterated upon using mass conservation until the exit/inlet area ratio has the correct known value. This is done as follows: Once the correct temperature and exit composition has been established, the mole fraction can be computed. The specific heats of individual species, which possess temperature dependent
empirical definitions, can be combined in correct fractions to obtain the frozen total specific heat and thus total $\gamma$. Likewise the total enthalpy is computed. With the total number of moles known we can then convert MR into MM which is grams of exiting material per mole of exit material. With MM, pressure, and temperature the exit density is defined. Finally, a simple mass conservation analysis gives the inlet/exit area ratio. In the Design case where pressure is constant and $\varepsilon$ is zero, the initial area ratio is unknown and not needed. The area ratio resulting from one pass of the described analysis then becomes the Design area ratio. For Off-Design conditions, the area ratio is already established, and, as noted, $\varepsilon$ must be adjusted to reproduce it.

### 3.3 Inclusion of Boundary Layer Effects

It is clear that the boundary layer from the inlet travels into the combustor and thus has an effect on the combustion process. Because of the level of the combustion analysis described earlier, the effects are primarily introduced as an adjustment in mass intake, area, and energy loss of the incoming air. This is accomplished by including the following adjustment factors into the velocity and energy equations presented earlier. The $\theta_s$ and $\delta_s$ in the expressions below refer to the momentum and displacement thicknesses respectively at the upper and lower combustor entrance walls. Percentage of inlet flow mass retained with the introduction of Boundary Layer:

$$\left(1 - \frac{\theta_{\text{upper}} + \theta_{\text{lower}}}{\text{height} - \theta_{\text{upper}} - \theta_{\text{lower}}} \right)$$

Percentage of increase from inlet flow area to actual inlet area:

$$\left(1 + \frac{\delta_{\text{upper}} + \delta_{\text{lower}}}{\text{height} - \delta_{\text{upper}} - \delta_{\text{lower}}} \right)$$

The weakness of these corrections is that they are for the combustor inlet air flow part of the equations only. The more complex effects within combustion process due to factors such as heat lost and wall friction have not been really represented. It is quite clear that there is a great deal of room for improvement.

The percentage energy loss in the incoming air requires a few steps. In isentropic flow stagnation temperature is given as $\frac{T_o}{T_{ci}} = 1 + (\gamma - 1)M_c^2/2$. In the actual flow the velocity drops to zero at the combustor surface where $T = T_w$. Therefore, an expression for percentage energy loss in the combustor before the combustion process can be obtained in terms of stagnation temperatures only.

$$\frac{T_w}{T_o} = \frac{T_w}{T_{ci}(1 + \frac{\gamma-1}{2}M_c^2)}$$
If this loss is contained only in the boundary layer, then the percentage energy contained in the boundary layer with respect to total energy in flow is

\[
\frac{\theta_{upper} + \theta_{lower}}{\text{height} - \theta_{upper} - \theta_{lower}}(1 - \frac{T_w}{T_{ci}}(1 + \frac{\gamma - 1}{2} M_{ci}^2))
\]

With the addition of this correction to the inlet air flow term of the energy balance equation, the effects of the boundary layer from the inlet is to a degree represented in our combustion analysis.
Chapter 4

NOZZLE ANALYSIS

The final component of the propulsion system is the expansion nozzle. As discussed earlier, the design limitations require that the nozzle be formed by the aftbody of the craft. The size and weight constraints all point to the generally accepted fact that the design will have to be for an external flow nozzle. The lack of symmetry introduces some interesting complexities especially in the external flow region. Another key factor that affects performance is the degree of recombination that occurs in the nozzle. At the temperatures reached in the combustor especially at the stoichiometric fuel condition dissociation definitely becomes a nontrivial energy loss factor. Therefore, the degree in which we can recover this energy through recombination in the nozzle will play a significant role in the overall performance of the propulsion system.

As with the other parts, the goal is to seek an analytical approach which provide us with credible information in a computational time scale compatible with the rest of the model. However, before beginning with the analytical techniques it is important to discuss the actual physical behavior of the flow in the nozzle. The exiting flow from the combustor goes through an expansion as it enters the nozzle. The expansion process can then be doubled by having the lower lip of the external nozzle reflect back all the initial expansion wave. Also, physically a flow crossing through an expansion wave and its reflection must once again become parallel. This is naturally desireable since we are looking for thrust. The initial expansion angle can then be selected to give the desired exit pressure assuming complete expansion. If the geometry of the nozzle lip is fixed, it is clear that because of the wide range of off design conditions, the expansion will not always be matched at the lip. For design, if we have pressure matching with the ambient and parallel flow at the exit then is true that the boundary between the nozzle flow and the external will be flat. However if we have under-expansion before the lip in comparison to the ambient, then the boundary will turn outward. On the other side, an over-expansion in comparison to the ambient will make the boundary contour turn inward. This then is the basic behavior
of an external flow nozzle.

From the description above, it is apparent that the changes in the nozzle are not one dimensional. However, following the control volume approach we can once again obtain valuable information about the flow properties at the entrance and exit where the flow is parallel. Assuming an isentropic expansion process, the region where the flow is frozen can be easily analyzed. We first start with the equation for enthalpy which is basically a combination by molar proportions of the individual species' enthalpy curve fits. The $S$'s in the equation below represent the summation of the individual species' curve fit constants with the number of moles of each species.

$$N_{\text{tot}}h = S1 + S2T + S3\theta^2 (\text{where } \theta = \frac{T}{1000})$$

(4.1)

Then for a constant entropy process where $d\theta = 0$:

$$dh = Tds + udP = \frac{RT}{P}dP$$

(4.2)

Combining the two equations and integrating, an iterative solution for $\theta$ arises:

$$\frac{\theta}{\theta_0} = \left(\frac{P}{P_0}\right)^{\frac{1000N_{\text{tot}}}{S2}} e^{-\frac{2S2}{S2}(\theta - \theta_0)}$$

(4.3)

The pressure ratio is known if the condition of pressure matching with the ambient is imposed. Once $\theta$ is found all the other properties, including area ratio, can be computed from standard relationships.

For those conditions where the pressure ratio is unknown but the area ratio is known there is the definition:

$$\frac{P}{P_0} = \frac{T}{T_0}\left(\frac{u}{u_0}\right)\left(\frac{A}{A_0}\right)$$

(4.4)

substituting this into equation(4.3) the result is another iterative definition for $\theta$:

$$\frac{\theta}{\theta_0} = \left(\frac{A}{A_0}\right)\left(\frac{u}{u_0}\right)\frac{1}{(\frac{1000N_{\text{tot}}}{S2})} e^{-\frac{2S2}{S2}(\theta - \theta_0)}$$

(4.5)

Since velocity is defined in terms of enthalpy which is a function of temperature, $u$ must be incorporated into the iteration. Once $\theta$ has been established one then goes back to compute the pressure ratio.

As noted earlier, there could be an equilibrium flow region in the initial part of the nozzle where the pressure is high and recombination does occur. Since entropy is constant, one can determine a corresponding temperature for a given pressure in the flow. This is accomplished by using the chemical equilibrium approach described in the previous chapter and iterating in temperature for a given pressure until the entropy is matched.
with the exit of the combustor. The equilibrium methods also provide the chemical composition from which the other properties could be computed. Thus, if the pressure at the freezing point is known, all the other properties can be computed. Then the conditions at the entrance to the frozen flow region are determined.

To fix the freezing point pressure, a separate analysis was completed involving the use of a chemical kinetics code developed by Roger Biasca. With the code, the time in which equilibrium conditions are reached given a pressure change profile could be computed. Therefore, if we test points down the equilibrium expansion by giving them a step pressure drop, a curve of equilibration time versus the pressure could be obtained. Such curves in fig(4.1) clearly shows that at some pressure and temperature the equilibration time rapidly shoots up showing that the recombination process has basically stopped. If the stoichiometry does not change and the entropy does not change then the process travels along a constant entropy line and the freezing point will be the same no matter where the initial pressure is at. Testing conditions across a range of Mach numbers and fuel ratios, the results showed that the change in combustor exit entropy due to changes other then the fuel ratio is roughly two orders of magnitude less then by a change in fuel ratio. Because of the errors associated with identifying the freezing point from a turning in the curves in figure(4.1) the small changes in freezing point pressure due to changes other then stoichiometry could not be clearly identified. Figure(4.2) shows that the changes due to stoichiometry on the other hand is very clear and significant. Thus, for predictions, freezing point pressure can be considered basically as a function only of the fuel ratio. Testing this across a range of fuel ratios, is is found that the lowest freezing pressure is at stoichiometric fuel condition. Also the increase on both sides is approximately linear. It turns out that for \( S < 1.0 \) \( P_{\text{freeze}}(\text{atm}) = -1.163S + 1.503 \) and \( (S > 1.0)P_{\text{freeze}}(\text{atm}) = 0.388S - 0.049 \) where \( S \) is the number of times stoichiometric fuel, is a reasonable estimate.
Equilibrium Time vs Temperature

Figure 4.1: Equilibrium time for decreasing temperature under expansion around the region of freezing

Freezing Point Pressure vs Fuel Ratio

Figure 4.2: Freezing point pressure for varied stoichiometry with linear curve fit
The next step is then to obtain information on the design geometry of the nozzle. A computationally quick method to accomplish this is the method of waves[3][13][14][15]. The foundation to this method is the fact that small disturbances propagate downstream expanding at an angle $\pm \mu$ from the direction of the stream. The lines going out of the point of disturbance at the angle $\pm \mu$ are known as Mach or characteristic lines. $\mu$ is totally dependent on the Mach number of the flow and can be expressed as $\sin^{-1}(\frac{1}{M_1})$. With this we can then apply the Prandtl-Meyer Function, defined as:

$$\nu = \int \frac{\cot \mu}{w} dw$$

(4.6)

For ideal gas flow this function is easily expressed in closed form:

$$\frac{a^2}{a_0^2} = 1 + \frac{\gamma - 1}{2} M^2$$

(4.7)

$$\nu = \int \cot \mu \frac{dw}{w}$$

$$\cot \mu = \sqrt{M^2 - 1}$$

$$\frac{dw}{w} = \frac{dM}{M} + \frac{da}{a} = \frac{dM}{M} \left( \frac{1}{1 + \frac{\gamma - 1}{2} M^2} \right)$$

This then leads to the definition

$$\nu(M) = \sqrt{\frac{\gamma + 1}{\gamma - 1}} \tan^{-1} \sqrt{\frac{\gamma - 1}{\gamma + 1}} (M^2 - 1) - \tan^{-1} \sqrt{M^2 - 1}$$

(4.8)

where the constant of integration is selected so that $\nu = 0.0$ when $M = 1$. The simplicity of the Prandtl-Meyer function lies in the fact that the change of the function due to an expansion turn is simply that function plus the turn angle.

Before continuing on, it should be clarified that the method of waves varies slightly from the method of characteristics in that the methods of characteristics uses a continuous velocity field where one computes at the lattice points of a grid structure formed by crossing characteristics. The characteristics resulting from the expansion are captured within this grid structure. The method of waves uses a patchwork of uniform property field of cells that are formed by the crossing of waves resulting directly from the expansion. For the two dimensional problem of nozzle contour design, the method of waves is computationally faster and easier to visualize. We begin the method of wave analysis by representing the initial expansion as a set of expansion waves which can be viewed as a series of small equal turns each with its appropriate Mach line summing up to the the complete turn. These waves and their reflections off the bottom lip form a network of
patches in which flow is uniform. The Prandtl-Meyer function for each patch is known since it is simply the initial function plus the number of waves, whose strength is defined as their turn angles, that were crossed before getting to the patch. The flow direction in each patch is also defined since it is also a sum of the waves one crossed before reaching the patch. However, waves of the opposite family change the direction of the flow in the opposite direction. From this, other flow properties within the patch could be computed. The standard design criterion for the upper contour of the nozzle is that wherever the reflected wave hits the upper surface, a compression turn will have to occur to cancel out any further reflections. If the exit condition must be set, such as pressure matching, then it is relatively easy to find the initial turn angle for the nozzle design.

\[
TurnAngle = \frac{\nu_{\text{initial}} - \nu_{\text{final}}}{2}
\]  

(4.9)

If the analysis is applied to a preset nozzle contour, characteristics of the same family may tend to merge into shocks thus causing the method to fail. For this and other computational reasons, the complete method of waves is only used for design purposes in this model.

Once the geometry is established, the off-design issue of external flow contour effect can be addressed. With the area ratio obtained in the control volume analysis, one can determine whether the flow is over or under-expanded. For either case the primary effect will be an additive drag term similar to that discussed for the inlet. For the over-expanded case the flow contour is approximated by a straight line from the lip to the correct exit area. The additive drag term is then computed by a Prandtl-Meyer expansion of the external flow over that line. For the underexpanded case, the last expansion wave extends beyond the lip. Therefore, we can approximate the internal pressure at the lip by flow going through two less than the design number of expansion waves. The \(\gamma\) in that region is near that for the exit condition, and therefore, the other properties can be calculated in that region. With the internal and external properties at the lip know one can then perform a pressure balance to compute the departure angle. The pressure on the inside goes through an isentropic expansion while the external compression is accomplished by a shock. Once the angle of the dividing streamline at its initial point as it leaves the nozzle lip is determined, a parabolic curve is fitted and a newtonian flow analysis is performed for the additive drag computation. With the additive drag term approximated, there is enough information to compute the performance of the whole system.
Chapter 5
MODELLING AND GLOBAL PARAMETERS

Now that the key components of the propulsion system has been modelled, the final step is to assemble everything together so as to give information leading to an increased understanding of system performance. Studying the types of information that are desired, it is judged that two program versions of the model should be constructed. The first version will be very automatic, taking in only the minimal control parameters necessary at the beginning. The purpose of this version is to enable the program to automatically and rapidly complete a series of runs through the variation of a specified control parameter such as Mach number. The second version is then a much more expanded program allowing the user to adjust parameters throughout the propulsion process and select alternative methods of analysis. This version therefore allows the user to closely study the operations at a single state. All the properties from the changes across each shock to the resulting chemical compositions as well as graphics of the geometry are displayed.

For the core model the essential parameters are the following: $\delta_1, \delta_2, \alpha$, design Mach number, ambient pressure and temperature, laminar or turbulent inlet boundary layer conditions, wall temperature, hydrogen injection temperature, oxidizer/fuel ratio, and combustor length. The nozzle is designed for pressure matching with the ambient.

If we then consider some very very basic concepts of trajectory, it is clear that the flight Mach number is tied to altitude and thus pressure to some degree. The relations between pressure and altitude can be expressed crudely and simply as those for an isothermal atmosphere:

\[
\int_0^P dP = \int_0^h pgdh \quad \text{where} \quad \rho = \frac{P}{RT}
\]

\[
P = P_0 e^{-kg/RT}
\]

For this model the temperature will be considered constant even though the temperature is recognized to vary with altitude.
To maintain a simple constant altitude flight, force balance along the axis of lift gives

\[ C_L \frac{1}{2} \rho V^2 S = (mg - \frac{mV^2}{r}) \cos(\alpha) \]  

(5.3)

This then leads to a definition of the required altitude for a given flight Mach number.

\[ h = \frac{-RT}{g} \ln\left[ \frac{\frac{m_g}{M_0 \gamma} - \frac{RT_m}{r}}{C_L S \frac{\rho_0}{2}} \cos(\alpha) \right] \]  

(5.4)

The new constraint introduced is \( C_L \), coefficient of lift, which is a property of the entire air frame. At hypersonic conditions, \( C_L \)'s dependance on Reynolds number is very small. Thus, \( C_L \) is basically only tied to the angle of attack and the relatively fixed shape of the airfoil and plane. With \( C_L \) and the nondimensionalizing parameter \( S \) (cord length or wing area) plus an approximation of the craft's mass specified, the cruise altitude and ambient pressure can be roughly approximated for a given flight Mach number. The effects of the other parameters can then be studied as the program completes the design Mach number run to set the geometry and then completes a series of below and above design condition runs. The resulting trajectory in terms of altitude and atmospheric pressure is given in figures (5.1) and (5.2).
Figure 5.1: Altitude vs Flight Mach Number

Figure 5.2: Atmospheric Pressure vs Flight Mach Number
The analysis presented earlier primarily gives information specific to each section. Yet, it is clear that performance can be better understood through the formation of some key global parameters. The most well known of these parameters is perhaps the specific impulse $I_{SP}$. From the properties at the entrance and exit of the propulsion system, thrust (T) and $I_{SP}$ can be defined in the following manner:

$$ T = \dot{m}_a[(1 + f)u_e - u_0] + (P_e - P_0)A_e - (C_{Dadd} + C_{Dadd, Nozzle})\frac{1}{2}\rho_0u_0^2AA' $$

$$ I_{SP} = \frac{T}{\dot{m}_g} $$

$$ I_{SP} = \frac{OF}{g}[(1 + \frac{1}{OF})u_e - u_0] + (P_e - P_0)\frac{A_eOF}{AA'\rho_0u_0CG} - (C_{Dadd} + C_{Dadd, Nozzle})\frac{u_0OF}{2gCR} $$

The generated thrust can be expressed in other forms. Nondimensionalizing similarly to the drag and lift coefficient, we get an expression for the thrust coefficient:

$$ C_F = \frac{T}{\frac{1}{2}\rho_0u_0^2AA'} = 2CR[(1 + \frac{1}{OF})\frac{u_e}{u_0} - 1] - \frac{2(P_e - P_0)A_e}{\rho_0u_0^2AA'} - (C_{Dadd} + C_{Dadd, Nozzle}) $$

Alternatively, the thrust parameter, defined as the ratio of thrust to inlet pressure force results in the expression

$$ \frac{F}{P_0A_0} = [(1 + \frac{1}{OF})\frac{u_e}{u_0} - 1]CM_0^2 + \frac{P_e - P_0}{P_0}\frac{A_e}{AA'} - \frac{(C_{Dadd} + C_{Dadd, Nozzle})\gamma M_0^2}{2} $$

The total efficiency of the system can be defined as the ratio of the thrust power over the power inputted through combustion. The thrust power is simply the product of the thrust and the velocity of the craft, and the power generated is the mass rate of fuel injected times the heat of reaction per unit fuel mass.

$$ \eta = \frac{Tu_0}{\dot{m}_f h} = \frac{gI_{SP}u_0}{h} $$

This completes the description of the analytical approach behind our hypersonic propulsion model. What remains is the presentation of the results of testing this model.
Chapter 6

RESULTS

Three primary categories of tests were performed on the propulsion model presented in order to achieve an indication of how the actual system should be designed and how the craft should be flown. For all the tests, the program performs the first run at a design Mach number to fix the geometry of the system. Then the system is tested through a range of off design Mach numbers starting from the maximum specified Mach number to the minimum at an increment also specified by the user. For the results to be presented, several variables which are also at the option of the user were also kept constant. These quantities are inlet wall temperature (1000°), fuel injection temperature (300°), and combustor wall temperature (2000°). There are no particular reasons for these exact numbers except they are roughly in the range of the real or desired conditions and thus provides a fair set of common reference conditions. Trials have shown that the results will not be greatly affected unless there is a severe deviation of these figures. The boundary layer conditions at the inlet were set to be laminar on the first ramp and the lip with the transition to turbulence occurring at the second ramp. The difficulty of the transition point has been discussed earlier. The decision to set the transition at the second ramp is not totally unreasonable since depending on the actual size of the system, $Re_a$ of the range of $10^7$ to $10^8$ have been computed. The diagrams presented later will show that even with a turbulent boundary layer, boundary layer growth is very minimal once the second ramp is reached.

6.1 Design Mach Number

The first set of tests on the model addresses the question of which Mach number one should design for. Figures(6.1 to 6.10) show the results for various design Mach numbers with the ramp angles frozen at seven degrees each. The results for the other tests are presented in this similar format. From the inlet, the combustor entrance pressure, temperature, average
kinetic energy efficiency, and combustor entrance Mach number were plotted versus the range of off-design Mach numbers for that particular design. From the combustor, the exit temperature and pressure were plotted. Finally, the performance parameters of Isp, thrust coefficient, thrust parameter, and the global efficiency were plotted.

As figure(6.1) shows, freezing the ramp angles and allowing the inlet opening and the rest of the geometry to vary results in a combustor inlet pressure of about the same for the design range of six to sixteen when combined with the trajectory calculation. The maximum off-design Mach number is limited by the fact that a single step is taken to go from the design Mach number to the maximum Mach number. Convergence problems can be avoided and higher Mach numbers reached if several steps in between are taken. The minimum Mach number is governed by convergence problems existing in the inlet and the combustor when the shocks miss the lip by a large distance. The second important reason for the Mach number limit is that the Isp quickly drops below the point of usefulness beyond the limits.

The problem of such a design approach is that while the inlet pressure is maintained at around the same range for different design Mach numbers, the inlet temperature soars to very high values. This then brings up problems such as dissociation, and increases the materials and cooling problems of the craft's surface.

The average kinetic energy efficiency peaks at the design condition and drops a few percentage points going to higher or lower Mach numbers. The peaks decrease slightly going from lower to higher design Mach numbers, but the drop off rate for off design is significantly less at higher design Mach numbers. As expected, peak Isp and thrust tends to be around the design region, dropping off significantly for off design conditions. The peak Isp decreases with design Mach number. However, the thrust parameter and global efficiency both increases with the design Mach number.

We have seen that with the first approach, the inlet temperature for high design Mach numbers reaches very high levels. The alternative is then to reduce the ramp angles as design Mach number increases. This serves to suppress inlet temperature as well as pressure. Also as a result, the combustor entrance Mach is much higher which may increase the mixing problem. As expected, with smaller ramp angles thus smaller disturbance, the average kinetic energy efficiency is higher. The Isp, Cf, thrust parameter, and global efficiency are found to be the same or slightly lower than that of the constant ramp angles results. Therefore, this approach with temperature suppression might be more desirable. The ramp angles used are 8 degrees per ramp at Mach 6 design, 7 degrees per ramp at Mach 10 design, 6 degrees per ramp at Mach 14 design, and 5 degrees per ramp at Mach
16 design. These results are presented in figures (6.11 to 6.20).
Figure 6.1: Combustor inlet pressure for varying design Mach number as indicated, with constant ramp angles

Figure 6.2: Combustor inlet temperature
Figure 6.3: Average kinetic energy efficiency of the inlet

Figure 6.4: Combustor inlet Mach number vs flight Mach number
Figure 6.5: Combustor exit temperature

Figure 6.6: Combustor exit pressure
Figure 6.7: Isp (sec) for varying design Mach numbers

Figure 6.8: Thrust coefficient
Figure 6.9: Thrust parameter

Figure 6.10: Global efficiency
Figure 6.11: Combustor inlet pressure for varying design Mach number, with varying ramp angles

Figure 6.12: Combustor inlet temperature with suppressed temperature curves
Figure 6.13: Average kinetic energy efficiency for the inlet

Figure 6.14: Combustor inlet Mach number vs flight Mach number
Figure 6.15: Combustor exit temperature

Figure 6.16: Combustor exit pressure
Figure 6.17: Isp(sec) for varying design Mach numbers

Figure 6.18: Thrust coefficient
Figure 6.19: Thrust parameter

Figure 6.20: Global efficiency
6.2 Angle of Attack

The second question addressed is the effect of flying at an angle of attack on performance. For this study, a system designed for Mach 10 with stoichiometric fuel ratio is tested at increasing angles of attack through a range of off-design Mach numbers. The pressure plots, figure (6.21), indicate that increasing the angle of attack lowers the Mach number at which the shocks miss the lip. Also, it is observed that the angle of attack has much less effect on pressure if the shocks miss the lip. This can be partly attributed to the fact that the changes in the flow capture due to angle of attack are not as great for conditions where the shocks miss the lip when compared to conditions where the shocks are swept in. The interesting result of changing the angle is that for the conditions when the shocks are swept in beyond the lip there is very little to no effect on average kinetic energy efficiency, Isp, Cf, and thrust parameter. However, since we have nondimensionalized with the current value for the capture height AA', thrust actually increases with $\alpha$. 
Figure 6.21: Combustor inlet pressure for varying angle attack in degrees as indicated

Figure 6.22: Combustor inlet temperature
Figure 6.23: Average kinetic energy efficiency of the inlet. Efficiency decrease with angle of attack at Mach 16.

Figure 6.24: Combustor inlet Mach number vs flight Mach number.
Figure 6.25: Combustor exit temperature

Figure 6.26: Combustor exit pressure
Figure 6.27: Isp(sec) for varying angle attack

Figure 6.28: Thrust coefficient
Figure 6.29: Thrust parameter

Figure 6.30: Global efficiency
6.3 Varying Stoichiometry

The third test addresses the question of the effect of oxidizer fuel ratio. Once again, a Mach 10 design was used, and the selected O/F conditions to be tested were 34.32 (stoichiometric), 67.16 (2x stoichiometric), 11.44 (3x stoichiometric), 51.48 (2/3x stoichiometric), 68.64 (1/2x stoichiometric). As expected, thrust increases with the amount of fuel. Also, the Isp results show that the stoichiometric condition gave the best Isp results in the range of Mach 6 to 16. Fuel lean conditions tended to achieve Isp's just as high as that for the stoichiometric condition in the design region, but the Isp drops off much faster going to lower and higher off design Mach numbers. Fuel rich results are much lower in Isp in the design region. However, since its Isp decreases slower than that for the stoichiometric condition, the Isp for fuel rich rises above that for stoichiometric at very high Mach numbers.
Figure 6.31: Combustor exit temperature for varying fuel ratio

Figure 6.32: Isp(sec) for varying fuel ratio
Figure 6.33: Thrust coefficient

Figure 6.34: Thrust parameter
Figure 6.35: Global efficiency
6.4 Other Results

The other results to be presented are comparisons of the newtonian flow analysis of the inlet to the analysis by shocks as described in Chapter (2). These are presented in figures (6.36 to 6.38). The results show that the newtonian flow computation is much more sensitive to flight Mach number so that the pressure increase due the Mach number dominates over pressure decrease due to the altitude. This is not so for our more accurate shock computation approach to the inlet. Also included are diagrams of some of the geometries discussed earlier and specially enlarged views of areas such as the boundary layer.
Figure 6.36: Combustor inlet pressure

Figure 6.37: Combustor inlet temperature
Figure 6.38: Isp(sec) vs flight Mach number
METHOD OF WAVES NOZZLE DESIGN

Figure 6.39: X and Y in meters

Figure 6.40: System: Mach 10 Design, Unit capture height
Figure 6.41: Inlet: Mach 10 Design

Figure 6.42: Inlet: Mach 16 Off-Design conditions for Mach 10 design
Figure 6.43: Inlet: Mach 8 Off-Design conditions for Mach 10 design
Chapter 7

CONCLUSION

As the results show, the performance of a SCRAMjet propulsion system is very sensitive to flight conditions. The design region tends to produce the ideal performance, with performance decreasing dramatically as one goes toward the extremes of off-design conditions. Considering that the actual craft is expected to fly at such a large range of off-design conditions, the design challenge is obviously enormous. As an alternative, variable geometry at places such as the inlet and the nozzle lips might provide the solution to some of the off-design difficulties. Apart from varying the geometry, it has been shown that other factors such as the angle of attack and the fuel ratio do have a significant effect on performance in different flight regions. For example, we may want to vary the angle of attack to prevent spillage at low Mach numbers and vary the fuel ratio at high Mach numbers to obtain a higher Isp. The results also showed that there are many trade-offs and that no single parameter may be the primary design criterion. While maximizing Isp may be ideal for fuel usage, at various points in the trajectory we may not get enough thrust. While designing the nozzle for full expansion may give the highest thrust, the size of the nozzle and the additive drag resulting from the external expansion will make such a design totally impractical.

Another issue that has greatly affected our research is that insufficient information has been obtained about the government’s actual plan for the design of the National Aerospace Plane. Due to the fact that most of such information are classified, our inability to design the model to reflect more precise conditions and problems is apparent. However, because many of the design issues relevant to the hypersonic craft are fundamental ones, our more general and flexible model has significantly extended our understanding of the key areas of difficulty that will have to be overcome if the National Aerospace Plane is to become a reality. On a positive note, our research has not identified any single problem so large as to render hypersonic flight of this nature totally unfeasible. The difficulties that has manifested itself so far all seemed to be solvable within the limits of current and near
future technology.

Finally, it is obvious that a great deal more work can be done in this line of research. The designed model can be further enhanced to deal with other issues such as heat transfer and the effects variable geometry discussed earlier. Further, much more detailed tests can be performed with the model. As already mentioned, this model will in the future be fitted into another larger trajectory model to provide vital propulsion system data. Hopefully, the research presented will only be a starting point, sparking interests in greater and more detailed research in the future.
Bibliography


Appendix A

MAIN PROGRAM WITH INLET ANALYSIS

C ********************************************************************************************
C THIS IS THE MAIN PROGRAM WHICH INCLUDES THE INLET ANALYSIS AND
C CONTAINS ALL THE CONTROL STRUCTURE FOR THE MODEL
C *******************************************************************************************

C DESIGN OF HYPERSONIC INLET

INTEGER N, I, II, J, S, COND, CONDB, DESIGN, CONDC, LAM, PTS
INTEGER POINTS(1:10), RUN, CONDS, SPECIAL, LOCK, BASE, NEWT
REAL M(0:5), DEL(1:6), DELO(1:5), X(1:5), MD, TA, PA, MACHA, AR
REAL PR(0:5), DR(0:5), TR(0:5), TW(1:5), PRR, DRR, TRR, STORECT(1:100)
REAL XAD, YAD, XAB, YAB, AB, XBC, YBC, YDH, BC, XDJN, YDJN, ALF
REAL ALFOLD, AA, BD, DALF, MLOW, MHIGH, TH2, TWC, OF, CLEW, NLEW, LR
REAL G, F, H, XDJ, YDJ, YJC, CDADD, CR, UE, PE, TE, DE, ME, PTR
REAL U2, U2X, U1, V1, V2, Y, PI, XI(1:200), YY(1:200), ETAK, ETAKV
REAL THKB(1:4), THKM(1:4), THKD(1:4), CF(1:4), HH(1:4), CTEMP
REAL T, P, TKO, THK(0:9), R, L(1:4), THKH(1:4), THKT(1:4)
REAL MA, MB, BA, BB, MC, ANG(1:4), LAMCON(1:4), ISP, TCF, TP, EFF
REAL THKMD(1:4), YEUD, FD, FR1, FR2, F1, F2, DMOME, K, YEXIT
REAL XQD, YQD, XPD, YPD, XCEU, LASTP, LASTT, STOREIS(1:200)
REAL STOREM(1:200), STOREP(1:200), STORET(1:200), STOREME(1:200)
REAL STOREEF(1:200), CAP, MCI, TCI, PCI, A12D, UD, DEMO, AM
REAL XKA, YKA, XKB, YKB, KK, KC, STOREEF(1:200), ARR, WAR, DM
REAL MASS, CD, CL, AREA, ALT, STORECF(1:200), STORETP(1:200)
REAL NLIP, EXANG, STORECP(1:200), CPRES, ZZ
CHARACTER*20 PLTITL
COMMON /PASS/ MCI, TCI, PCI, XX, YY, PTS, DESIGN, LEN, ALF, CDADD, AM
COMMON /PASS2/ A12D, UO, DEMO, CAP, ARR, WAR, TA, PA, MACHA, DALF, CR
COMMON /PASS3/ TH2, TWC, OF, CLEI, ILEI, LR, YEXIT, ISP, TCF, TP, EFF
COMMON /PASS4/ LOCK, BASE, CTEMP, CPRES, MLIP, EXANG
PI = 3.1415927
RUN = 1
ZZ = 2.4
LOCK = 2.0
II = 1
Y = 1.4
CALL GR_INIT(5, 6, '
')
CONDB = 1
C GETS THE DESIGN SPECIFICATIONS FROM THE USER **********************
PRINT *, 'PLEASE GIVE CAPTURE HEIGHT'
READ *, CAP

DO WHILE (CONDB .EQ. 1)
BASE = 1
LOCK = 2.0
PRINT *, 'PLEASE ENTER DEL1, DEL2'
READ *, DEL(1), DEL(2)
DEL(1) = DEL(1) * PI / 180.0
DEL(2) = DEL(2) * PI / 180.0
PRINT *, 'CRAFT MASS, WING AREA, CL'
READ *, MASS, AREA, CL
PRINT *, 'PLEASE GIVE COMBUSTOR LENGTH'
READ *, CLEN

DO WHILE (LOCK .EQ. 2)
CDADD = 0.0
CR = 1.0
POINTS(RUN) = 1
DO 100 I = 1, 200, 1
XX(I) = 0.0
100 CONTINUE
YY(I) = 0.0

CONTINUE

PRINT *, 'THE FIRST RUN IS A DESIGN POINT CALCULATION'
PRINT *, 'PLEASE DESIGN MACH, MIN, MAX'
READ *, MD, MLOW, MHIGH
M(O) = MD
MACHA = M(O)
PRINT *, 'MACH INCREMENT?'
READ *, DM
PRINT *, '(1)STANDARD(2)NEWTONIAN 2PANEL(3)NEWTONIAN 1PANEL'
READ *, NEWT
DESIGN = 1
COND(S) = 2
COND = 1
PRINT *, 'PLEASE INDICATE LAM OR TURB COND ON 4 PANELS (1)=LAM'
READ *, LAMCON(1), LAMCON(2), LAMCON(3), LAMCON(4)
PRINT *, 'PLEASE GIVE ANGLE OF ATTACK'
READ *, ALF
ALF = ALF * PI/180.0
DALF = ALF

PRINT *, 'PLEASE INPUT THE FOUR WALL TEMPERATURES'
READ *, TW(1), TW(2), TW(3), TW(4)
PRINT *, 'COMBUSTOR AND NOZZLE INFORMATION'
PRINT *, 'PLEASE GIVE HYDROGEN INJECTION PLUS WALL TEMP'
READ *, TH2, TWC
PRINT *, 'PLEASE GIVE (O/F) BY MASS'
READ *, OF

C //////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
C LOOP TO RUN THROUGH DESIGN AND OFF-DESIGN CONDITIONS
DO WHILE( COND .EQ. 1)
IF (COND .NE. 1) ALF = 0.0
IF ((COND .EQ. 1) .AND. (POINTS(RUN) .EQ. 1)) THEN
ALF = DALF
END IF
IF (POINTS(RUN) .GT. 1) THEN
DALF = 0.0
END IF
CDADD = 0.0
CR = 1.0
C ALTITUDE AND AMBIENT PRESSURE COMPUTATIONS *********************
ALT = MASS * (1.0/(M(O)*M(O) * 10250) - 1.0/6.37E6)*COS(ALF)
ALT = -7321.43*LOG(ALT/(CL * AREA * 0.72153))
PR(O) = 1.0354E5 * EXP(-1.0*ALT/7321.43)
PA = PR(O)
TR(O) = 250.0
TA = 250.0
DO 110 I = 9, 19, 1
YY(I) = 0.0
XX(I) = 0.0
110 CONTINUE
IF (DESIGN .NE. 1) THEN
DO 115 I = 1, 200, 1
YYY = YY(I)
XXX = XX(I)-XAD
XX(I) = XXX*COS(DALF)+YYY*SIN(DALF)+XAD
YY(I) = -1.0*XXX*SIN(DALF) + YYY*COS(DALF)
115 CONTINUE
AR = AR * AA / YY(1)
AA = YY(1)
XX(20) = XX(21) - 0.5
YY(20) = YY(20)
END IF
DEL(3) = DEL(1) + DEL(2)
COND = 1
C ENTERING CONDITIONS
UO = M(O) * SQRT(1.4 * 287.0 * TR(O))
DEN0 = PR(O) / (287.0 * TR(O))
LASTP = 0.0
LASTT = 0.0
DO WHILE(CONDC .EQ. 1)
CALL SHOCK((DEL(1)+ALF), M(0), X(1))
CALL RATIO(M(0), X(1), (DEL(1)+ALF), M(1), PR(1), DR(1), TR(1))
IF (DESIGN .EQ. 1) THEN
AA = SIN(X(1)) * CAP / SIN(X(1)-ALF)
END IF
CALL SHOCK(DEL(2), M(1), X(2))
CALL RATIO(M(1), X(2), DEL(2), M(2), PR(2), DR(2), TR(2))
U1 = M(1)/M(0) * SQRT(TR(1)) * COS(DEL(1))
V1 = U1 * TAN(DEL(1))
U2 = M(2)/M(0) * SQRT(TR(1)*TR(2)) * COS(DEL(3))
V2 = U2 * TAN(DEL(3))
C CHECK IF CONDITIONS ARE CORRECT
SPECIAL = 10
IF (DESIGN .NE. 1) THEN
IF (((X(1) - ATAN(YY(21)/(XX(76)-XX(21)))) .GT. 0.0) THEN
KC = ATAN(YY(22)/ABS(XX(76)-XX(22)))
IF (((XX(76)-XX(22)) .LT. 0.0) THEN
KC = KC + PI/2.0
END IF
IF (((X(2)+DELO(1)) - KC) .GT. 0.0) THEN
DESIGN = 2
ELSE
SPECIAL = 1
DESIGN = 3
MA = -1.0*TAN(DELO(1)+ALF+X(2))
BA = YY(22) - MA*XX(22)
MB = -1.0*TAN(ALF)
BB = YY(76) - MB*XX(76)
XKA = (BB - BA)/(MA-MB)
YKA = MA * XKA + BA
MA = -1.0 * TAN(DELO(1)+ALF)
BA = YKA - MA*XKA
MB = -1.0 * TAN(X(1))
79
\[ BB = YY(21) - MB*XX(21) \]
\[ XKB = (BB - BA)/(MA - MB) \]
\[ YKB = MA * XKB + BA \]
\[ KK = (AA - YKB)/AA \]
\[ CDADD = 2.0/(1.4*M(0)*M(0))*(PR(1)-1.0)*YKB \]
\[ END IF \]
\[ ELSE \]
\[ KK = 1.0 \]
\[ YKB = 0.0 \]
\[ END IF \]
\[ END IF \]
\[ IF (DESIGN .NE. 3) THEN \]
\[ CALL SHOCK(DEL(3), M(2), X(3)) \]
\[ CALL RATIO(M(2), X(3), DEL(3), M(3), PR(3), DR(3), TR(3)) \]
\[ C DESIGN CONDITION ****************************************************** \]
\[ IF (DESIGN .EQ. 1) THEN \]
\[ PRR = PR(1) * PR(2) * PR(3) \]
\[ TRR = TR(1) * TR(2) * TR(3) \]
\[ DRR = DR(1) * DR(2) * DR(3) \]
\[ ETAK = 1 - (TRR / PRR**((0.4/1.4) - 1.0)) / (0.2 * M(0)**2.0) \]
\[ AR = (1/DRR) * (M(0)/M(3)) * SQRT(1/TRR) \]
\[ ME = M(3) \]
\[ C INLET GEOMETRY \]
\[ YAD = AA \]
\[ XAD = AA / TAN(X(1)) \]
\[ AB = \sin(DEL(1)+ALF+X(2))/TAN(X(1))-\cos(DEL(1)+ALF+X(2)) \]
\[ AB = AB * AA / \sin(X(2)) \]
\[ YAB = AB * \sin(DEL(1)+ALF) \]
\[ XAB = AB * \cos(DEL(1)+ALF) \]
\[ BD = AA * (\cos(DEL(1)+ALF)-\sin(DEL(1)+ALF)/TAN(X(1)))/\sin(X(2)) \]
\[ BC = BD * \sin(PI-X(2)-X(3)+DEL(2))/\sin(X(3)) \]
\[ XBC = XAB + BC * \cos(ALF+DEL(1)+DEL(2)) \]
\[ YBC = YAB + BC * \sin(ALF+DEL(1)+DEL(2)) \]
\[ L(1) = AB \]
\begin{align*}
L(2) &= BC \\
L(3) &= 5.0 \times AR \times AA \\
L(4) &= L(3) + AR \times AA / \tan(X(3) - DELO(1) - DELO(2)) \\
\text{ELSE} \\
\text{IF}(\text{NEWT} .LT. 2) \text{ THEN} \end{align*}

C OFF DESIGN ( MO < MO DESIGN) ***********************************************

\begin{align*}
\text{CALL SHOCK(DEL(3), M(3), X(4))} \\
\text{CALL RATIO(M(3), X(4), DEL(3), M(4), PR(4), DR(4), TR(4))} \\
MA &= -1.0 \times \tan(DELO(1) + ALF + DELO(2)) \\
BA &= YY(23) - MA \times XX(23) \\
MB &= \tan(X(3) - ALF - DELO(1) - DELO(2)) \\
BB &= YY(76) - MB \times XX(76) \\
XDJ &= (BB - BA) / (MA - MB) \\
YDJ &= MA \times XDJ + BA \\
MB &= -1.0 / MA \\
BB &= YY(76) - MB \times XX(76) \\
XDJN &= (BB - BA) / (MA - MB) \\
YDJN &= MA \times XDJN + BA \\
DJN &= \sqrt{(XDJN - XX(76))^2 + (YDJN - YY(76))^2} \\
YJC &= YDJ - YY(23) \\
U2 &= M(2) / M(O) \times \sqrt{TR(1) \times TR(2)} \\
U2X &= U2 \times \cos(DELO(1) + ALF + DELO(2)) \\
CR &= \text{DR}(1) \times \text{DR}(2) \times U2 \times (DJN - \text{THKD}(2)) / AA \\
MA &= -\tan(DELO(1) + ALF + DELO(2) + ANG(2)) \\
BA &= YY(76) - MA \times XX(76) \\
MB &= -\tan(X(2) + DELO(1) + ALF + ANG(1)) \\
BB &= YY(22) - MB \times XX(22) \\
XQD &= (BB - BA) / (MA - MB) \\
YQD &= MA \times XQD + BA \\
MA &= -\tan(DELO(1) + ALF + ANG(1)) \\
BA &= YQD - MA \times XQD \\
MB &= -\tan(X(1) + ANG(1) + DELO(1) + ALF) \\
BB &= YY(21) - MB \times XX(21) \\
XPQ &= (BB - BA) / (MA - MB) \\
\end{align*}
YPQ = MA * XPQ + BA
YPQ = YPQ - YQD

CDADD = 2.0/(1.4*M(0)*M(0)*AA)*((PR(1)-1.0)*YPQ+(PR(2)-1.0)*YQD)

G = CR * AA / (YEUD - THKD(3) - THKD(4))
FD = (L(3)*SIN(ANG(3)) + L(4)*SIN(ANG(4)))/(1.4*M(0)*M(0))
FD = FD - (THKM(3)+THKM(4)-THKMO(3))
FD = FD * PRR * ME * ME * COS(ALF)/ (M(0) * M(0))

DNOME = PRR*ME*ME*(THKD(3)+THKD(4)+THKM(3)+THKM(4))

DNOME = DNOME * COS(ALF)/(M(0)*M(0))
F = PR(1)*PR(2)*YDJ - PR(1)*PR(2)*PR(3)*PR(4)*YJC
F = F + PR(1)*PR(2)*PR(3)*PR(4)*SIN(ALF)*L(4)-L(3))
F = F/(1.4*M(0)*M(0)) + DR(1)*DR(2)+U2+U2X*(DJM-THKD(2)-THKM(2))
F = (F + FD + DMOME)/(YEUD*COS(ALF))

ENDIF
ENDIF
ELSE

C OFF DESIGN ( MO > MO DESIGN) ******************************************************

IF (NEWT .LT. 2) THEN

G = AA*KK / (YEUD - THKD(3) - THKD(4))
FD = (L(3)*SIN(ANG(3)) + L(4)*SIN(ANG(4)))/(1.4*M(0)*M(0))
FD = FD - (THKM(3)+THKM(4)-THKMO(3))
FD = FD * PRR * ME * ME * FD * COS(ALF)/ (M(0) * M(0))

DNOME = DNOME * (THKD(3)+THKD(4)+THKM(3)+THKM(4)) * COS(ALF)
FR1 = PR(1)*M(1)*M(1)*THKM(1)*COS(DELO(1)+ALF)/(M(0)*M(0))
FR2 = PR(1) * PR(2) * M(2) * M(2) * (THKM(2) - THKMO(2))
FR2 = FR2 * COS(DELO(1)+ALF+DELO(2)) / (M(0) * M(0))
F = AA*KK-(AA-YY(22)-YKB)*PR(1) - (YY(22)-YY(23))*PR(1)*PR(2)
F = (F + PRR*SIN(ALF)*L(4)-L(3)) / (1.4 * M(0) * M(0))
F = (AA*KK + F + FD + DMOME - FR1 - FR2) / (YEUD*COS(ALF))

ENDIF
ENDIF

IF (DESIGN .NE. 1) THEN
IF (NEWT .LT. 2) THEN
  F1 = 7.0 / 12.0 * F / G
  H = 0.5 + 1.0 / (0.4 * M(O) * M(O))

  F2 = H / 3.0
  UE = F1 + SQRT(F1 * F1 - F2)
  PE = F - G * UE
  TE = 3.5 * PE * UE / G
  DE = G / UE
  ME = UE / SQRT(0.4 * TE)
  PRR = 1.4 * M(O) * M(O) * PE
  TRR = 0.4 * M(O) * M(O) * TE
  DRR = DE

C OPTION TO USE NEWTON FLOW ANALYSIS *******************************
ELSE
  IF (NEWT .EQ. 2) THEN
    PR(1) = 1.4 * 287.0 * TR(O) * ZZ * (SIN(Del(1) + ALF))**2 * DENO
    PR(1) = 0.5 * PR(1) * M(O) * M(O) + PR(O)
    PR(1) = PR(1) / PR(O)
    M(1) = SQRT(5.0 * ((1.0 / PR(1))**0.2857 * (1.0 + 0.2 * M(O) * M(O)) - 1.0))
    TR(1) = (1.0 + 0.2 * M(O) * M(O)) / (1.0 + 0.2 * M(1) * M(1))
    DR(1) = PR(1) / TR(1)
    PR(2) = 1.4 * 287.0 * TR(1) * TR(O) * ZZ * (SIN(Del(2)))**2 * DENO * DR(1)
    PR(2) = 0.5 * PR(2) * M(1) * M(1) + PR(1) * PR(0)
    PR(2) = PR(2) / (PR(1) * PR(O))
  ELSE
    PR(2) = 1.4 * 287.0 * TR(O) * ZZ * (SIN(Del(1) + Del(2) + ALF))**2 * DENO
    PR(2) = 0.5 * PR(2) * M(O) * M(O) + PR(O)
    PR(2) = PR(2) / PR(O)
    PR(1) = 1.0
    TR(1) = 1.0
    M(1) = M(O)
    DR(1) = 1.0
  END IF
\[ M(2) = \sqrt{5.0 \times ((1.0/PR(2))^{0.2857} \times (1.0 + 0.2 \times M(1) \times M(1)) - 1.0)} \]
\[ TR(2) = (1.0 + 0.2 \times M(1) \times M(1)) / (1.0 + 0.2 \times M(2) \times M(2)) \]
\[ DR(2) = \frac{PR(2)}{TR(2)} \]
\[ PR(3) = 1.4 \times 287.0 \times TR(2) \times TR(1) \times TR(0) \times (\sin(DEL(2) + DEL(1) + ALF))^{2} \]
\[ PR(3) = PR(3) \times 2 \times 0.5 \times M(2) \times M(2) \times DEMO \times DR(1) \times DR(2) \times PR(0) \times PR(1) \times PR(2) \]
\[ PR(3) = \frac{PR(3)}{(PR(2) \times PR(1) \times PR(0))} \]
\[ M(3) = \sqrt{5.0 \times ((1.0/PR(3))^{0.2857} \times (1.0 + 0.2 \times M(2) \times M(2)) - 1.0)} \]
\[ TR(3) = \frac{(1.0 + 0.2 \times M(2) \times M(2))}{(1.0 + 0.2 \times M(3) \times M(3))} \]
\[ DR(3) = \frac{PR(3)}{TR(3)} \]
\[ PRR = PR(1) \times PR(2) \times PR(3) \]
\[ TRR = TR(1) \times TR(2) \times TR(3) \]
\[ DRR = DR(1) \times DR(2) \times DR(3) \]
\[ ME = M(3) \]
\[ END IF \]
\[ ETAK = 1.0 - \frac{(TRR / PRR^{0.4/1.4}) - 1.0}{(0.2 \times M(0) \times M(0))} \]
\[ K = 0.2 \times M(0) \times M(0) \]
\[ ETAKV = (1 + K) \times (THK(3) + THK(4)) \]
\[ ETAKV = ETAKV + (1 + K \times (1 - ETAK)) \times (THKT(3) + THKT(4)) \]
\[ ETAKV = ETAK - \frac{ETAKV}{(K \times YEUD - THKD(3) - THKD(4))} \]
\[ END IF \]

**C BOUNDARY LAYER CALCULATIONS **************************************************

\[ THK(0) = 0.0 \]
\[ THK(1) = 0.0 \]
\[ THK(2) = 0.0 \]
\[ T = TR(0) \times TR(1) \]
\[ P = PR(0) \times PR(1) \]
\[ LAM = LAMCON(1) \]
\[ CALL LAYER(L(1), TW(1), T, M(1), P, X(1), (DEL(1) + ALF), LAM, THK) \]
\[ THKB(1) = THK(1) \]
\[ THKM(1) = THK(2) \]
\[ THKD(1) = THK(3) \]
\[ CF(1) = THK(4) \]
\[ \text{HH}(1) = \text{THK}(5) \]
\[ \text{ANG}(1) = \text{THK}(6) \]
\[ \text{THK}(0) = \text{THK}(1) \]
\[ \text{THKO} = \text{THK}(2) \]
\[ \text{THK}(2) = \frac{\text{THK}(2) \cdot \sin(X(2) - \text{DEL}(2))}{\sin(X(2)) \cdot (M(1)/M(2))} \]
\[ \text{THK}(1) = \frac{\text{THK}(1) \cdot \text{THK}(2)}{\text{THKO}} \]
\[ \text{THKMO}(2) = \text{THK}(2) \]
\[ T = \text{TR}(0) \cdot \text{TR}(1) \cdot \text{TR}(2) \]
\[ P = \text{PR}(0) \cdot \text{PR}(1) \cdot \text{PR}(2) \]
\[ \text{LAM} = \text{LAMCON}(2) \]
\[ \text{CALL LAYER}(L(2), TW(2), T, M(2), P, X(2), \text{DEL}(2), \text{LAM}, \text{THK}) \]
\[ \text{THKB}(2) = \text{THK}(1) \]
\[ \text{THKM}(2) = \text{THK}(2) \]
\[ \text{THKD}(2) = \text{THK}(3) \]
\[ \text{CF}(2) = \text{THK}(4) \]
\[ \text{HH}(2) = \text{THK}(5) \]
\[ \text{ANG}(2) = \text{THK}(6) \]
\[ \text{THK}(0) = \text{THK}(1) \]
\[ \text{THKO} = \text{THK}(2) \]
\[ \text{THK}(2) = \frac{\text{THK}(2) \cdot \sin(X(3) - \text{DEL}(3))}{\sin(X(3)) \cdot (M(2)/M(3))} \]
\[ \text{THK}(1) = \frac{\text{THK}(1) \cdot \text{THK}(2)}{\text{THKO}} \]
\[ \text{THKMO}(3) = \text{THK}(2) \]
\[ T = \text{TR}(0) \cdot \text{TRR} \]
\[ P = \text{PR}(0) \cdot \text{PRR} \]
\[ \text{LAM} = \text{LAMCON}(3) \]
\[ \text{CALL LAYER}(L(3), TW(3), T, M(3), P, X(3), \text{DEL}(3), \text{LAM}, \text{THK}) \]
\[ \text{THKB}(3) = \text{THK}(1) \]
\[ \text{THKM}(3) = \text{THK}(2) \]
\[ \text{THKD}(3) = \text{THK}(3) \]
\[ \text{CF}(3) = \text{THK}(4) \]
\[ \text{HH}(3) = \text{THK}(5) \]
\[ \text{THKH}(3) = \text{THK}(7) \]
\[ \text{THKT}(3) = \text{THK}(8) \]
\[ \text{ANG}(3) = \text{THK}(6) \]
\[ \text{THK}(0) = 0.0 \]
THK(1) = 0.0
THK(2) = 0.0
LAM = LAMCON(4)
CALL LAYER(L(4), TW(4), T, ME, P, X(3), DEL(3), LAM, THK)
THKB(4) = THK(1)
THKM(4) = THK(2)
THKD(4) = THK(3)
CF(4) = THK(4)
HH(4) = THK(5)
THKH(4) = THK(7)
THKT(4) = THK(8)
ANG(4) = THK(6)

IF (DESIGN .EQ. 1) THEN
CONDC = 2
ELSE
CONDC = 1
IF (ABS(LASTP - PRR) .LT. 0.02) THEN
IF (ABS(LASTT - TRR) .LT. 0.02) THEN
CONDC = 2
END IF
END IF
LASTP = PRR
LASTT = TRR
DEL(1) = DELO(1) + ANG(1)
DEL(2) = DELO(2) + ANG(2) - ANG(1)
DEL(3) = DELO(3) + ANG(2) + ANG(4)
ENDIF
ENDIF
DO C NOW WE REWORK THE GEOMETRY TO INCORPORATE THE B.L. ***************
IF (DESIGN .EQ. 1) THEN
XX(1) = 0.0
XX(2) = XAB
XX(3) = XBC
\[
XX(4) = XBC + L(3) \cdot \cos(\alpha) \\
XX(6) = XAD \\
XX(7) = XAD + L(4) \cdot \cos(\alpha) \\
YY(1) = AA \\
YY(2) = AA - YAB \\
YY(3) = AA - YBC \\
YY(4) = YY(3) - L(3) \cdot \sin(\alpha) \\
YY(6) = 0.0 \\
YY(7) = YY(6) - L(4) \cdot \sin(\alpha) \\
MC = \tan(X(3) - \Delta L(3) - \alpha) \\
XX(20) = -0.5 \\
YY(20) = AA \\
XX(21) = XX(1) \\
YY(21) = AA \\
\Delta L(1) = \Delta L(1) - \Delta \theta(1) \\
\Delta L(2) = \Delta L(2) - \Delta \theta(2) + \Delta \theta(1) \\
\Delta L(3) = \Delta L(1) + \Delta L(2) - \Delta \theta(4) \\
MA = -1.0 \cdot \tan(\Delta L(1) + \alpha) \\
BA = YY(1) - MA \cdot XX(1) \\
MB = -1.0 \cdot \tan(X(2) + \Delta L(1) + \alpha) \\
BB = YY(2) - MB \cdot XX(2) \\
XX(22) = (BB - BA) / (MA - MB) \\
YY(22) = MA \cdot XX(22) + BA \\
MA = -1.0 \cdot \tan(\Delta L(1) + \alpha + \Delta L(2)) \\
BA = YY(22) - MA \cdot XX(22) \\
MB = \tan(X(3) - \Delta L(3) - \alpha) \\
BB = YY(6) - MB \cdot XX(6) \\
XX(23) = (BB - BA) / (MA - MB) \\
YY(23) = MA \cdot XX(23) + BA \\
XX(24) = XX(4) + \Delta H(3) \cdot \sin(\alpha) \\
XX(76) = XX(6) \\
XX(77) = XX(7) - \Delta H(4) \cdot \sin(\alpha) \\
YY(24) = YY(4) + \Delta H(3) \cdot \cos(\alpha) \\
YY(76) = YY(6) \\
YY(77) = YY(7) - \Delta H(4) \cdot \cos(\alpha) 
\]
YEUD = AR + AA + THKD(3) + THKD(4)
K = 0.2 * M(0) * M(0)
ETAKV = (1 + K) * (THKH(3) + THKH(4))
ETAKV = ETAKV + (1 + K * (1 - ETAK)) * (THKT(3) + THKT(4))
ETAKV = ETAK - ETAKV / (K*(YEUD - THKD(3) - THKD(4)))
LASTP = PRR
LASTT = TRR
L(1) = SQRT((XX(22)-XX(21))**2.0 + (YY(22)-YY(21))**2.0)
L(2) = SQRT((XX(23)-XX(22))**2.0 + (YY(23)-YY(22))**2.0)
L(3) = SQRT((XX(24)-XX(23))**2.0 + (YY(24)-YY(23))**2.0)
L(4) = SQRT((XX(77)-XX(76))**2.0 + (YY(77)-YY(76))**2.0)
ELSE
MA = -1.0 * TAN(DEL(1)+ALF)
BA = YY(21) - MA * XX(21)
MB = -1.0 * TAN(X(2)+DEL(1)+ALF)
BB = YY(22) - MB * XX(22)
XX(2) = (BB - BA) / (MA - MB)
YY(2) = MA * XX(2) + BA
MA = TAN(X(3) - (DEL(1)+ALF+DEL(2)))
IF (DESIGN .EQ. 3) THEN
MA = MC
ENDIF
BA = YY(76) - MA * XX(76)
MB = -1.0 * TAN(DEL(2)+DEL(1)+ALF)
BB = YY(2) - MB * XX(2)
XX(3) = (BB - BA) / (MA - MB)
YY(3) = MA * XX(3) + BA
XX(4) = XX(24) - THKD(3) * SIN(ALF)
YY(4) = YY(24) - THKD(3) * COS(ALF)
XX(7) = XX(77) + THKD(4) * SIN(ALF)
YY(7) = YY(77) + THKD(4) * COS(ALF)
ENDIF
C SHOCK GEOMETRY
XX(10) = XX(1)
YY(10) = YY(1)

XX(11) = XX(1) + AA / TAN(X(1))

YY(11) = 0.0

XX(12) = XX(2)

YY(12) = YY(2)

XX(13) = XX(2) + YY(2) / TAN(X(2)+DEL(1)+ALF)

YY(13) = 0.0

IF (DESIGN .EQ. 3) THEN

MA = -1.0 * TAN(X(1))

BA = YY(10) - XX(10) * MA

MB = -1.0 * TAN(X(2)+DEL(1)+ALF)

BB = YY(12) - XX(12) * MB

XX(16) = (BB - BA)/(MA - MB)

YY(16) = MB * XX(16) + BB

X(S) = ASIN(((PR(1)*PR(2)*2.4 -0.4)/4.8)**0.5/M(0))

MA = -1.0 * TAN(X(S))

BA = YY(16) - XX(16) * MA

MB = -1.0 * TAN(ALF)

BB = YY(76) - XX(76) * MB

XX(17) = (BB - BA)/(MA - MB)

YY(17) = MA * XX(17) + BA

IF(YY(16) .GT. YY(11)) THEN

XX(11) = XX(16)

YY(11) = YY(16)

XX(13) = XX(16)

YY(13) = YY(16)

ELSE

XX(16) = 0.0

YY(16) = 0.0

XX(17) = 0.0

YY(17) = 0.0

END IF

ENDIF

ENDIF

IF(DESIGN .NE. 3) THEN
\[ MA = (YY(3) - YY(2)) / (XX(3) - XX(2)) \]
\[ BA = YY(2) - MA * XX(2) \]
\[ MB = \tan(XX(3)-\text{DEL}(1)-\text{ALF}-\text{DEL}(2)) \]
\[ BB = YY(6) - MB * XX(6) \]
\[ XX(14) = XX(6) \]
\[ YY(14) = YY(6) \]
\[ XX(15) = (BB - BA) / (MA - MB) \]
\[ YY(15) = MB * XX(15) + BB \]

END IF

IF (DESIGN .EQ. 2) THEN
\[ XX(16) = XX(15) \]
\[ YY(16) = YY(15) \]
\[ MA = -1.0 * \tan(XX(4)+\text{ALF}) \]
\[ BA = YY(16) - MA * XX(16) \]
\[ MB = -1.0 * \tan(\text{ALF}) \]
\[ BB = YY(76) - MB * XX(76) \]
\[ XX(17) = (BB - BA)/(MA - MB) \]
\[ YY(17) = MA * XX(17) + BA \]
END IF

C DISPLAY RESULTS

\[ PCI = PR(0) * PRR / 101300.0 \]
\[ TCI = TR(0) * TRR \]
\[ MCI = ME \]
IF (COND .EQ. 1) THEN
\[ STOREM(II) = M(0) \]
\[ STOREP(II) = PRR * PA / 101300.0 \]
\[ STORET(II) = TRR * TA \]
\[ STOREME(II) = ME \]
\[ STOREEF(II) = ETAKV \]
END IF

COND = 0
ARR = AR
CALL COMMNOZ(THKD(3),THKD(4),THKM(3),THKM(4),YEUR)
IF (COND S .EQ. 1) THEN
STOREIS(II) = ISP
STORECF(II) = TCF
STORETP(II) = TP
STOREEFT(II) = EFF
STORECT(II) = CTEMP
STORECP(II) = CPRES
POINTS(RUN) = POINTS(RUN) + 1
II = II + 1
END IF
IF ((M(O) .EQ. MD) .AND. (COND S .NE. 1)) THEN
M(O) = MHIGH
COND S = 1
COND D = 1
ELSE
M(O) = M(O) - DM
IF (M(O) .LT. MLOW) THEN
COND D = 2
ELSE
COND D = 1
END IF
END IF
IF(M(O) .LT. MD) THEN
DESIGN = 2
ELSE
DESIGN = 3
END IF
BASE = 2.0
MACHA = M(O)
END DO
C ///////////////////////////////////////////////////////////////////////////
COND D = 0
DO WHILE (COND D .NE. 10)
PRINT *, 'SPECIAL OPTIONS'
91
PRINT *, '--------------------'
PRINT *, '(1) DIAGRAM OF THE SYSTEM'
PRINT *, '(2) PRESSURE RATIO VS MACH'
PRINT *, '(3) TEMP RATIO VS MACH'
PRINT *, '(4) EFFICIENCY VS MACH'
PRINT *, '(5) MACH EXIT VS MACH IN'
PRINT *, '(6) ISP VS MACH'
PRINT *, '(7) CF VS MACH'
PRINT *, '(8) THRUST PAR VS MACH'
PRINT *, '(9) GLOBE EFFICIENCY'

PRINT *, '(10) CONTINUE ON'
PRINT *, '(11) CLEAR STORAGE'
PRINT *, '(12) COMBUSTION TEMPERATURE VS MACH'
PRINT *, '(13) COMBUSTION PRESSURE VS MACH'
READ *, COND
IF (COND .EQ. 1) THEN
    CALL DIAG(XX, YY, PTS)
END IF

IF (COND .EQ. 2) THEN
    PLTITL = 'MACH'PRESSURE(ATM)'
    CALL CURVE(STOREM, STOREP, POINTS,PLTITL)
END IF

IF (COND .EQ. 3) THEN
    PLTITL = 'MACH' TEMPERATURE(K)'
    CALL CURVE(STOREM, STORET, POINTS,PLTITL)
END IF

IF (COND .EQ. 4) THEN
    PLTITL = 'MACH' EFFICIENCY'
    CALL CURVE(STOREM, STOREEF, POINTS,PLTITL)
END IF

IF (COND .EQ. 5) THEN
    PLTITL = 'MACH' ALT(M)'
    CALL CURVE(STOREM, STOREME, POINTS,PLTITL)
END IF

92
IF (COND .EQ. 6) THEN
PLTITL = 'MACH''ISP(sec)' 
CALL CURVE(STOREM, STOREIS, POINTS, PLTITL)
END IF
IF (COND .EQ. 7) THEN
PLTITL = 'MACH''CF'
CALL CURVE(STOREM, STORECF, POINTS, PLTITL)
END IF
IF (COND .EQ. 8) THEN
PLTITL = 'MACH''THRESH PAR'
CALL CURVE(STOREM, STORETP, POINTS, PLTITL)
END IF
IF (COND .EQ. 9) THEN
PLTITL = 'MACH''GLOBAL EFF'
CALL CURVE(STOREM, STOREEFT, POINTS, PLTITL)
END IF
IF (COND .EQ. 12) THEN
PLTITL = 'MACH''COMB.TEMP(K)'
CALL CURVE(STOREM, STORECT, POINTS, PLTITL)
END IF
IF (COND .EQ. 13) THEN
PLTITL = 'MACH''COMB.PRES(ATM)'
CALL CURVE(STOREM, STORECP, POINTS, PLTITL)
END IF
IF (COND .EQ. 11) THEN
DO 550 I = 1, 100, 1
STOREM(I) = 0.0
STOREP(I) = 0.0
STORET(I) = 0.0
STOREME(I) = 0.0
STOREEFT(I) = 0.0
550
END IF
STORECT(I) = 0.0
STORECP(I) = 0.0

550 CONTINUE
END IF
END DO
RUN = RUN + 1
PRINT *, 'DO YOU WISH TO UNLOCK THE DESIGN PARAMETERS'
READ *, LOCK
END DO
CONDDB = 2
PRINT *, 'DO YOU WISH TO TRY ANOTHER DESIGN CALC?'
PRINT *, '(1) YES'
PRINT *, '(2) NO '
READ *, CONDB
END DO
END
Appendix B

COMBUSTOR AND NOZZLE ANALYSIS SUBPROGRAM

C ******************************************************
C COMBUSTOR-NOZZLE ANALYSIS
C TAKES VALUES FROM INLET PROGRAM
C ******************************************************

SUBROUTINE COMNOZ(THKDU,THKDL,THKMU,THKML,HIGH)
REAL A(1:10,1:3), B(1:6,1:3), W0, WC, WH, FI, P, MR, HR
REAL MCI, TCI, PCI, UACI, TH2, UFC1, MACHA, CR, CDADD, TWC, WLEN
REAL NUC, NUH, NUWF, NUOF, NUOX, X0, OFST, OF, HOX, HF
REAL HLP, YO, UCE, CE, R, T, F, H, W(1:10), WTOT, X(1:10), D
REAL CPAV, CP(1:10), S(1:10), BR, GAMMA, DACI, STOT, MM, A12, CTEMP
REAL MCE, SS, CEW, CEO, A21, A12D, A12O, SO(1:10), EX, HEIGHT
REAL TO, PO, STUTO, GAMMO, MMO, TH0, TA, PA, PR, UE, DE, ISP, V
REAL CEQ, UE1, UEO, PE, BBAR, CBAR, THE, THE1, C1, YEXIT
REAL AR, C2, C3, C4, SA, SB, SC, PRFR, DFR, TFR, UFR, LR, TAU
REAL GAMBAR, DO, TFHR, UO, DEHO, CAP, ARR, WAR, TCF, XXX, YYY
REAL XX(1:200), YY(1:200), HIGH, CLEN, MMTOT, ANG, L, PIN, MO
REAL Q, QO, K1, K2, KA, KB, DX, DY, FX, FY, MCM, FXTOT, FYTOT
REAL THKDU, THKDL, THKMU, THKML, DELBAR, THMBAR, DBAR, TP, EFF
REAL AN, AA, AO, E, K, STOREIS(1:100), PI, TQUESS, ALF, DALF
REAL V1, V2, P2, MW, NWALL, MOZANG, MOZADD, WAR2, NHEI
REAL TLIP, GAMME, GAMML, SS, MLIP, WALLIP, PIECE, XMAX, YMAX
REAL MLIP, PLIP, ULIP, PANEL(0:10,1:3), CPRES

INTEGER I, J, IDES, COND, ITEXP, CONDA, NCAL, ISEP, IT, CONDB
INTEGER PTS, DESIGN, CONDS, BASE, LOCK

95
COMMON A, B
COMMON /STAT/ NO, NC, MM, MH, FI, P, MR, HR, SA, SB, SC
COMMON /SPEC/ SQ, W, NTOT
COMMON /PASS/ MCI, TCI, PCI, XX, YY, PTS, DESIGN, LEN, ALF, CDADD, AM
COMMON /PASS2/ A1D, UO, DEMO, CAP, ARR, WAR, TA, PA, MACHA, DALF, CR
COMMON /PASS3/ TH2, TWC, OF, CLEN, NLEN, LR, YEXIT, ISP, TCF, TP, EFF
COMMON /PASS4/ LOCK, BASE, CTEMP, CPRES, MLIP, MRLIP
COND = 0
PI = 3.1415927

C READ IN STARTUP DATA FROM CSTART.DAT FILE

OPEN(UNIT = 1, FILE = 'CSTART.DAT', STATUS='OLD')

20 FORMAT (F10.5, F10.5, F10.5)
30 FORMAT (F10.5)

DO 105 I = 1, 10, 1
    DO 90 J = 1, 3, 1
    READ (UNIT = 1, FMT = 30) A(I,J)
90 CONTINUE
105 CONTINUE

DO 150 I = 1, 6, 1
    DO 140 J = 1, 3, 1
    READ (UNIT = 1, FMT = 30) B(I,J)
140 CONTINUE
150 CONTINUE

DO 200 I = 1, 10, 1
    READ (UNIT = 1, FMT = 30) SG(I)
200 CONTINUE

C

T Guess = -100.0

DELBAR = (THKDU + THKDL) / (HIGH - THKDU - THKDL)

THMBAR = (THKMU + THKML) / (HIGH - THKDU - THKDL)

DHBAR = THMBAR *(1.0 - TCI/ TWC/ (1.0 + 0.2 * MCI * MCI))

UACI = MCI * SQRT(1.4 * 287 * TCI)

DACI = PCI * 101300.0 * 28.9 / (8314 * TCI)
UFCI = SQRT(1.4 * 4157 * TH2)
NUC = 0
NUH = 2.0
NUWF = 0
NUOF = 0
NUOX = 2.0
NUOX = 7.52
XO = (2.0 * NUC + NUH/2.0 - NUOF) / NUOX
OFST = XO * (16.0 * NUG + 14.0 * NUOX)
OFST = OFST / (12.0 * NUC + NUH + 14.0 * NUWF + 16.0 * NUOF)
FI = OFST / OF
HOX = A(5,1) + 3.76*A(10,1)+(A(5,2) + 3.76 * A(10,2)) / TCI / 1000
HOX = HOX + (A(5,3) + 3.76*A(10,3)) * (TCI / 1000) * (TCI / 1000)
HF = A(1,1) + A(1,2) * TH2 / 1000
HF = HF + A(1,3) * (TH2 / 1000) * (TH2 / 1000)
HLP = 0.0
NN = 1
NC = NUC/NUH
YO = OF * (12.0 * NUC + NUH + 14.0 * NUWF + 16.0 * NUOF)
YO = YO / (16.0 * NUOX + 14.0 * NUOX)
NO = (NUOF + YO * NUOX) / NUH
NW = (NUWF + YO * NUOX) / NUH
MR = (NN + 12.0 * NC + 16.0 * NO + 14.0 * NW) / 1000
COND = 10

ITEXP = 0
CE = 0.0

C DESIGN CALCULATIONS REQUIRE ONLY ONE PASS /////////////////////////////////////////////////////////////////
DO WHILE (COND .NE. 1)
COND = 10
IF (BASE .EQ. 1) THEN
UCE = UACI * ((OF*1.0-THMBAR) + UFCI/UACI) / (OF+1)
HR = HF + 0.002 * UFCI/UFCI / 8374
HR = HR + YO * (1.0-DEBAR)*HOX + 0.13728 * UACI/UACI / 8374)
HR = HR / NUH - MR * HLP - MR * UCE*UCE / 8374
HL = MR * HLP
P = PCI
CALL ENTHALPY(T, H, N, NTOT, TGUESS)
COND = 1
C ITERATE UNTIL THE RIGHT AREA RATIO IS ACHIEVED ///////////////
ELSE
R = 28.9 * TH2 * UACI / (2.0 * TCI * UFCI)
EX = 1.0 / (CE - 1.0)
BR = 1.0 - A12D**EX
UCE = UACI * (OF*(1.0-THMBAR) + UFCI / UACI)
UCE = UCE + UACI * (BR*CE*(OF*(1.0+DELBAR)+R)/(1.4*MCI*MCI))
UCE = UCE / (OF+1)
HR = HF + 0.002 * UFCI * UFCI / 8374.0
HR = HR + YO *(1.0-DBBAR)*(HOX + 0.13728 * UACI*UACI / 8374)
HR = HR / MUH - MR * HLP - MR * UCE * UCE / 8374
EX = CE / (CE - 1.0)
P = PCI * A12D**EX
CALL ENTHALPY(T, H, N, NTOT, TGUESS)
END IF
C /////////////////

DO 1000 I = 1, 10, 1
X(I) = N(I) / NTOT
1000 CONTINUE
CPAV = 0
DO 1010 I = 1, 10, 1
CP(I) = A(I,2) + 2 * A(I,3) * T / 1000.0
CPAV = CPAV + CP(I) * N(I)
1010 CONTINUE
CPAV = CPAV / NTOT
GAMMA = CPAV/ (CPAV - 1.986)
MM = MR / NTOT * 1000
D = P * 101300.0 * MM / (8314 * T)
STOT = 0
DO 1020 I = 1, 10, 1
IF (N(I) .LT. 0.00000007) THEN
S(I) = 1
ELSE
S(I) = S(I) - 1.986 * LOG(P * N(I) / NTUT) + A(I,2) * LOG(T/1000)
S(I) = S(I) + 2.0 * A(I,3) * (T / 1000 - 1.0)
ENDIF
STOT = STOT + N(I) * S(I)
CONTINUE
A12 = D/DACI * UCE/UACI
A12 = A12*(OF/(1.0-DELBAR)+28.9*TH2*UACI/(2*TCI*UFCl))/(OF+1)
A21 = 1/A12
MCE = UCE / SQRT(GAMMA * 8314.0 / WM * T)
IF (BASE .EQ. 1) THEN
A12D = A12
ENDIF
IF (BASE .NE. 1) THEN
IF (ITEXP .GT. 0) THEN
IF (((ABS(1-A12/A12D) .LT. 0.0001) .OR. (A12 .EQ. A120))THEN
COND = 1
IF (A12 .EQ. A120) THEN
PRINT *, 'DESIGN AREA IN COMBUSTOR IS NOT REACHED'
ENDIF
ELSE
ITEXP = ITEXP + 1
SL = (CE - CEO) / (A12 - A120)
CEN = CEO + SL * (A12D - A120)
CEO = CE
CE = CEN
A120 = A12
ENDIF
ELSE
ITEXP = 1
CEO = CE
A120 = A12
CE = CE + 0.2
END IF
ELSE
COND = 1
END IF
END DO
C COMBUSTER ANALYSIS COMPLETED //////////////////////////////////////////////////

IF (DESIGN.EQ.1) THEN
XX(25) = SQRT((XX(77)-XX(76))**2 + (YY(77)-YY(76))**2)+CLEN
YY(25) = A21 * HIGH

XX(5) = XX(25)
YY(5) = YY(25) - THKDU
ELSE
XX(5) = XX(25) - SIN(ALF) * THKDU
YY(5) = YY(25) - COS(ALF) * THKDU
XX(8) = XX(78) + SIN(ALF) * THKDL
YY(8) = YY(78) + SIN(ALF) * THKDL
END IF
AO = SQRT((YY(25) - YY(78))**2.0 + (XX(25) - XX(78))**2.0)
TO = T
TFR = TO
CTEMP = TO
TLIP = TO
PO = P
CPRES = PO
STOTO = STOT
GAMMO = GAMMA
GAMME = GAMMO
GAMML = GAMMA
MMO = MM
THO = T / 1000
HRO = HR - HLP

PR = PA / (PO * 101300.0)
PE = PA
```plaintext
C EQUILIBRIUM FLOW CALCULATION

SS = (1.0/NO)/2.0
IF(SS .GE. 1.0) THEN
  PFR = 0.38781 * SS - 0.04851
ELSE
  PFR = -1.16342 * SS + 1.50272
END IF
IF(PFR .LT. PO) THEN
  IF(PFR .LT. (PA / 101300.0)) THEN
    PFR = PA / 101300.0
  END IF
  CALL TFIND(TFR, PFR, GAMMA, STOTO)
GAMME = GAMMA
GAMML = GAMMA
UFR = SQRT(UCE * UCE - (HR - HRO) * 4180 / 0.5 / MR)
DFR = PFR * 101300.0 * MM/(8314 * TFR)
PRFR = PFR / PO
T = TFR
DE = PE * MM / (8314 * T)
DO = PO * 101300.0 * MMO / (8314 * TO)
UE = UFR
ELSE
  TFR = TO
  PFR = PO
  DFR = PFR * 101300.0 * MM / (8314 * TFR)
  PRFR = 1.0
  UFR = UCE
END IF

C FROZEN NOZZLE CALCULATIONS FORM THIS POINT ON

IF(PFR .GT. (PA / 101300.0)) THEN
  BBAR = SB / NTOT
  CBAR = SC / NTOT
THE = 1
```
THFR = TFR / 1000.0
IT = 0
C1 = 1.986 / BBAR
C2 = 2.0 * CBAR / BBAR
THE1 = THE
THE = THFR * (PR/PRFR)**C1 * EXP(-C2 * (THE - THFR))
DO WHILE ( ABS(1 - THE / THE1) .GE. 0.0001)
THE1 = THE
THE = THFR * (PR/PRFR)**C1 * EXP(-C2 * (THE1 - THFR))
IT = IT + 1
IF ( IT .GT. 40) THEN
PRINT * , 'CONVERGENCE PROBLEM EXISTS'
END IF
END DO
T = THE * 1000
DE = PE * MM / (8314 * T)
DO = PO * 101300.0 * MMO/(8314 * TO)
UE = BBAR + CBAR * (THFR + THE)
UE = SQRT(UFR * UFR + 8360000.0 / MM * UE*(THFR - THE))
END IF

WAR2 = DO * UCE / DE / UE
ME = UE / (SQRT(GAMME * PE / DE))
HEIGHT = HIGH /A12D
Y = GAMMO
VO = SQRT((Y+1)/(Y-1))*ATAN(SQRT((Y-1)/(Y+1) *(MCE*MCE-1.0)))
VO = VO - ATAN(SQRT(MCE*MCE-1.0))

IF (DESIGN .EQ. 1) THEN
WAR = WAR2
VF = SQRT((Y+1)/(Y-1))*ATAN(SQRT((Y-1)/(Y+1) *(ME*ME - 1.0)))
VF = VF - ATAN(SQRT(ME*ME-1.0))
EXANG = (VF - VO)/2.0
CALL MDZ(MCE,STOTO,(PO*101300.0),TO,GAMMO,HEIGHT,MACHA,TA,
& PA, DESIGN, PANEL,EXANG)
\[ xx(78) = xx(25) + \text{PANEL}(0,1) \]
\[ yy(78) = 0.0 \]
\[ xx(8) = xx(78) \]
\[ yy(8) = yy(78) + \text{THKDL} \]
\[ xx(28) = xx(25) + \text{PANEL}(2,1) \]
\[ xx(27) = xx(25) + \text{PANEL}(3,1) \]
\[ xx(28) = xx(25) + \text{PANEL}(4,1) \]
\[ xx(29) = xx(25) + \text{PANEL}(5,1) \]
\[ xx(30) = xx(25) + \text{PANEL}(6,1) \]
\[ yy(26) = \text{PANEL}(2,2) \]
\[ yy(27) = \text{PANEL}(3,2) \]
\[ yy(28) = \text{PANEL}(4,2) \]
\[ yy(29) = \text{PANEL}(5,2) \]
\[ yy(30) = \text{PANEL}(6,2) \]
\[ pts = 5 \]
\[ nlen = \text{PANEL}(6,1) \]
\[ mlip = \text{PANEL}(0,1) \]
\[ narlip = (mlip * \tan(exang)) + \text{HEIGHT} / \text{HEIGHT} \]
\[ \text{END IF} \]

\[ \text{IF}(narl2 .gt. nar) \text{ THEN} \]
\[ vf = vo + (8/5) * \text{exang} \]
\[ \text{gamml} = \text{gamme} \]
\[ \text{CALL MACH}(mlip, me, vf, \text{gamml}) \]
\[ plip = (1 + (\text{gamme} - 1) * me * me / 2)/(1 + (\text{gamml} - 1) * mlip * mlip / 2) \]
\[ plip = \text{pe} * \text{plip} ** ((\text{gamml} / (\text{gamml} - 1.0))) \]
\[ \text{CALL VERGE}((plip / 101300.0), (pa / 101300.0), \text{gamml}, \text{mlip}, \text{& MACHA, ALF, DD}) \]
\[ dd = \text{ABS}(dd) \]
\[ \text{IF} (dd .gt. 0.01) \text{ THEN} \]
\[ \text{piece} = \text{ABS}(\text{height} * (\text{narl2} - \text{nar})) \]
\[ xmax = (nlen - mlip) * \cos(\text{alf}) - \text{piece} * \sin(\text{alf}) \]
\[ ymax = (nlen - mlip) * \sin(\text{alf}) + \text{piece} * \cos(\text{alf}) \]
\[ \text{CALL DRAG} (\text{xmax}, \text{ymax}, \text{dd}, \text{pa}, \text{macha}, \text{alf}, \text{fd}) \]
\[ nozadd = 2.0 * \text{fd} / (1.4 * \text{pa} * \text{macha} ** 2.0 * \text{high} / \text{arr}) \]
IF(IAR2 .LT. NAR) THEN

NOZANG = ATAn(ABS((NAR - IAR2)*HEIGHT)/(NLEN - NLIP))
Nwall = (NLEN - NLIP) / COS(NOZANG)

NOZANG = ABS(NOZANG)

V1 = 2.4494 * ATan(SQRT(0.16667 * (MACHA * MACHA - 1.0)))
V1 = V1 - ATan(SQRT(MACHA * MACHA - 1.0))
V2 = V1 + NOZANG - ALF

CALL MACH(M2, MACHA, V2, 1.4)
P2 = PA * ((1.0 + 0.2*M2*M2)/(1.0 + 0.2*MACHA*MACHA))**3.5
NOZADD = 2.0*(P2/PA-1)*SIN(NOZANG)*NWALL
NOZADD = ABS(NOZADD) /(1.4*MACHA**2.0 * HIGH / ARR)

END IF

ISP = (UE*(1 + 1.0/OF) - UO) * OF / 9.8
ISP = ISP + (PE-PA)*ARR*WAR2/OF/(A12D*DENO*UO*9.8*CR)
ISP = ISP - (CDADD+NOZADD) * UO * OF / (19.6 + CR)

TCF = 2.0 * CR *(UE/UO * (1.0+1.0/OF) - 1.0)

TP = ((1.0 + 1.0/OF) * UE/UO - 1.0) * CR * 1.4 * MACHA**2.0

EFF = 9.8 * ISP * UO / 120E6

C *****************************************************************************
C NOZZLE AND FINAL RESULTS
C *****************************************************************************

PRINT *, 'NOZZLE AND FINAL RESULTS'
PRINT *, '-------------------------------'
PRINT *, 'PE', PE, 'UE', UE, 'TE', T
PRINT *, 'TFR', TFR
PRINT *, 'ISP', ISP
PRINT *, 'CF', CF
PRINT *, 'WAR', WAR, 'WAR2', WAR
PRINT *, 'NOZZLE EXIT RESULT **********************'

C ROTATE COMBUSTOR AND NOZZLE TO THE ATTACK ANGLE

IF (DESIGN .EQ. 1) THEN
  DO 3005 I = 25,200, 1
  IF (((I .LT. 76) .OR. (I .GT. 77)) THEN
    XXX = XX(I)
    YYY = YY(I)
    XX(I) = COS(ALF) * XXX + SIN(ALF) * YYY + XX(76)
    YY(I) = -1.0 * SIN(ALF) * XXX + COS(ALF) * YYY
  END IF
  3005 CONTINUE
  XXX = XX(5)
  YYY = YY(5)
  XX(5) = COS(ALF) * XXX + SIN(ALF) * YYY + XX(76)
  YY(5) = -1.0 * SIN(ALF) * XXX + COS(ALF) * YYY
  XXX = XX(8)
  YYY = YY(8)
  XX(8) = COS(ALF) * XXX + SIN(ALF) * YYY + XX(76)
  YY(8) = -1.0 * SIN(ALF) * XXX + COS(ALF) * YYY
END IF
CLOSE(UNIT = 1)
RETURN
END
Appendix C

METHOD OF WAVES SUBPROGRAM

C ****************************************************************
C THIS PROGRAM IS A METHOD OF WAVES ANALYSIS OF THE NOZZLE
C ****************************************************************

SUBROUTINE NOZ(MO,SCON,PO,TO,YO,HEIGHT,E,TA,PA,DESIGN,PANEL,DD)
INTEGER DESIGN,FLOW,I,J,LINE,UPPER, COUNT1, COUNT2
INTEGER POINTS(1:20), PT, POSC, POSA, POSB, SIDE
REAL GRID(0:10,0:10,1:11), VA, MO, SCON, PO, TO, YO, NS, HEIGHT
REAL E, TA, PA, DEL(1:2,1:10), PANEL(0:10,1:3), DD
REAL MA, MB, BA, BB, X, Y
REAL A(1:10,1:3), B(1:6,1:3), NO, NC, NW, WH, FI, P, MR, HR
REAL SA, SB, SC, PI
REAL XP(1:100), YP(1:100), ANGIN, ANGOUT, DANG
REAL W(1:10), SO(1:10), NTOT
COMMON A, B
COMMON /STAT/ NO, NC, NW, WH, FI, P, MR, HR,
SA, SB, SC
COMMON /SPEC/ SO, N, NTOT

PI = 3.1415927
WS = 5
FLOW = 2
ANGIN = DD
C START-UP COMPUTATIONS
POSA = 0
POSB = 0
PANEL(1,1) = 0.0
PANEL(1,2) = HEIGHT
PANEL(1,3) = ANGIN
VA = SQRT((YO+1)/(YO-1)) * ATAN(SQRT((YO-1)/(YO+1)*(MO*MO-1)))
VA = VA - ATAN(SQRT(MO*MO-1.0))
GRID(0,0,1) = 0.0
GRID(0,0,2) = 0.0
GRID(0,0,3) = VA
GRID(0,0,4) = 0.0
GRID(0,0,5) = MO
GRID(0,0,6) = ASIN(1.0/MO)
GRID(0,0,7) = YO
GRID(0,0,8) = TO
GRID(0,0,9) = PO
DANG = ANGIN / NS
SIDE = 1
DO 6000 I = 1, NS, 1
POSA = I
DEL(1,I) = DANG
GRID(POSA,POSB,1) = 0.0
GRID(POSA,POSB,2) = HEIGHT
GRID(POSA,POSB,3) = GRID(POSA-1,POSB,3) + DEL(1,I)
GRID(POSA,POSB,4) = GRID(POSA-1,POSB,4) + DEL(1,I)
CALL INFO(POSA,POSB,FLOW, GRID, SCON, SIDE)
GRID(POSA,POSB,11) = ((GRID(POSA-1,POSB,4)-GRID(POSA-1,POSB,6)) + (GRID(POSA,POSB,4)-GRID(POSA,POSB,6))) / 2.0
6000 CONTINUE

NMAX = NS
DO WHILE (POSB .LT. NS)
POSB = POSB + 1
POSA = POSB
DO WHILE (POSA .LE. NMAX)
IF (POSA .EQ. POSB) THEN
MA = TAN(GRID(POSA,POSB-1,11))
BA = GRID(POSA,POSB-1,2) - MA * GRID(POSA,POSB-1,1)
GRID(POSA,POSB,2) = 0.0
GRID(POSA,POSB,1) = (GRID(POSA,POSB,2) - BA) / MA  
DEL(2,POSB) = DANG  
ELSE  
MA = TAN(GRID(POSA,POSB-1,11))  
BA = GRID(POSA,POSB-1,2) - MA * GRID(POSA,POSB-1,1)  
MB = TAN(GRID(POSA-1,POSB,10))  
BB = GRID(POSA-1,POSB,2) - MB * GRID(POSA-1,POSB,1)  
GRID(POSA,POSB,1) = (BB - BA)/(MA - MB)  
GRID(POSA,POSB,2) = MA * GRID(POSA,POSB,1) + BA  
PRINT *, GRID(POSA,POSB-1,11),GRID(POSA-1,POSB,10),  
& GRID(POSA,POSB,1), GRID(POSA,POSB,2)  
END IF  
IF ((DESIGN NE. 1) .AND.(POSA.EQ.POSB).AND.  
& (GRID(POSA,POSB,1).GT.PANEL(0,1))) THEN  
ELSE  
GRID(POSA,POSB,3) = GRID(POSA,POSB-1,3) + DEL(2,POSB)  
GRID(POSA,POSB,4) = GRID(POSA,POSB-1,4) - DEL(2,POSB)  
IF (POSA .EQ. POSB) THEN  
SIDE = 2  
ELSE  
SIDE = 1  
END IF  
CALL INFO(POSA,POSB,FLOW,GRID,SCOM,SIDE)  
GRID(POSA,POSB,10) = (((GRID(POSA,POSB-1,4)+GRID(POSA,POSB-1,6))+  
& (GRID(POSA,POSB,4)+GRID(POSA,POSB,6)))/ 2.0  
GRID(POSA,POSB,11) = (((GRID(POSA-1,POSB,4)-GRID(POSA-1,POSB,6))+  
& (GRID(POSA,POSB,4)-GRID(POSA,POSB,6)))/ 2.0  
END IF  
POSA = POSA + 1  
END DO  
POSA = POSA - 1  
MA = TAN(GRID(POSA,POSB,10))  
BA = GRID(POSA,POSB,2) - MA * GRID(POSA,POSB,1)  
MB = TAN(PANEL(POSB,3))  
BB = PANEL(POSB,2) - MB * PANEL(POSB,1)  

108
\[ I = (BB - BA) / (MA - MB) \]
\[ Y = MA * I + BA \]

IF (DESIGN .EQ. 1) THEN
\[ PANEL(POSB+1,1) = X \]
\[ PANEL(POSB+1,2) = Y \]
\[ PANEL(POSB+1,3) = PANEL(POSB,3) - 1.0 * DEL(2,POSB) \]
ELSE
IF (X .NE. PANEL(POSB+1,1)) THEN
IF (X .LT. PANEL(POSB+1,1)) THEN
\[ DEL(1,POSA+1) = DEL(2,POSB) \]
\[ DEL(1,POSA+2) = -1.0 * DANG \]
\[ GRID(POSA+1,POSB,1) = X \]
\[ GRID(POSA+1,POSB,2) = Y \]
\[ GRID(POSA+2,POSB,1) = PANEL(POSB+1,1) \]
\[ GRID(POSA+2,POSB,2) = PANEL(POSB+1,2) \]
ELSE
\[ DEL(1,POSA+1) = -1.0 * DANG \]
\[ DEL(1,POSA+2) = DEL(2,POSB) \]
\[ MB = TAM(PANEL(POSB+1,3)) \]
\[ BB = PANEL(POSB+1,2) - MB * PANEL(POSB+1,1) \]
\[ GRID(POSA+2,POSB,1) = (BB - BA) / (MA - MB) \]
\[ GRID(POSA+2,POSB,2) = MA * GRID(POSA+2,POSB,1) + BA \]
\[ GRID(POSA+1,POSB,1) = PANEL(POSB+1,1) \]
\[ GRID(POSA+1,POSB,2) = PANEL(POSB+1,2) \]
END IF

\[ GRID(POSA+1,POSB,3) = GRID(POSA,POSB,3) + DEL(1,POSA+1) \]
\[ GRID(POSA+1,POSB,4) = GRID(POSA,POSB,3) + DEL(1,POSA+1) \]

CALL INFO(POSA+1,POSB,FLOW,GRID,SCON)

\[ GRID(POSA+1,POSB,11) = ((GRID(POSA,POSB,4)-GRID(POSA,POSB,6)) + & (GRID(POSA+1,POSB,4)-GRID(POSA+1,POSB,6))) \]
\[ GRID(POSA+2,POSB,3) = GRID(POSA+1,POSB,3) + DEL(1,POSA+2) \]
\[ GRID(POSA+2,POSB,4) = GRID(POSA+1,POSB,4) + DEL(1,POSA+2) \]

CALL INFO(POSA+2,POSB,FLOW,GRID,SCON)
\[ GRID(POSA+2,POSB,11) = ((GRID(POSA+1,POSB,4)-GRID(POSA+1,POSB,6))) \]
& + (GRID(POSA+2,POSB,4) - GRID(POSA+2,POSB,6)))
END IF
END IF
END DO
PANEL(0,1) = GRID(NS, NS, 1)
PANEL(0,2) = 0.0
C ORGANIZATION FOR GRAPICS

LINE = 1
POINTS(LINE) = 0
PT = 1
DO 6400 I = 1,NS,1
POSA = I
POSB = 0
DO WHILE (POSB .LE. POSA)
XP(PT) = GRID(POSA,POSB,1)
YP(PT) = GRID(POSA,POSB,2)
PT = PT + 1
POINTS(LINE) = POINTS(LINE) + 1
POSB = POSB + 1
END DO
IF (NMAX .GT. NS) THEN
UPPER = NS + (POSA - 1) * 2
ELSE
UPPER = NS
END IF
POSB = POSA
POSC = POSA + 1
DO WHILE (POSC .LE. UPPER)
XP(PT) = GRID(POSC,POSB,1)
YP(PT) = GRID(POSC,POSB,2)
PT = PT + 1
POINTS(LINE) = POINTS(LINE) + 1
POSC = POSC + 1
END DO
XP(PT) = PANEL(POSB+1,1)  
YP(PT) = PANEL(POSB+1,2)  
PT = PT + 1  
POINTS(LINE) = POINTS(LINE) + 1  
LINE = LINE + 1  
POINTS(LINE) = 0  

6400 CONTINUE  
IF (WMAX .GT. WS) THEN  
COUNT1 = 0  
COUNT2 = 0  
DO 6410 I = (NS+1),NMAX,1  
POSA = I  
COUNT2 = COUNT2 + 1  
IF (COUNT2 .EQ. (COUNT1 + 2)) THEN  
COUNT1 = COUNT2  
POSB = 1 + COUNT1 / 2  
END IF  
DO WHILE (POSB .LE. WS)  
XP(PT) = GRID(POSA,POSB,1)  
YP(PT) = GRID(POSA,POSB,2)  
PT = PT + 1  
POINTS(LINE) = POINTS(LINE) + 1  
POSB = POSB + 1  
END DO  
LINE = LINE + 1  
POINTS(LINE) = 0  

6410 CONTINUE  
END IF  
DO 6450 I = 1,(NS+1),1  
XP(PT) = PANEL(I,1)  
YP(PT) = PANEL(I,2)  
PT = PT + 1  
POINTS(LINE) = POINTS(LINE) + 1  

6450 CONTINUE  
LINE = LINE + 1
POINTS(LINE) = 2
XP(PT) = PANEL(1,1)
YP(PT) = 0.0
PT = PT + 1
XP(PT) = PANEL(0,1)
YP(PT) = PANEL(0,2)

CALL GRIDGRAPH(XP, YP, LINE, POINTS)
END

C *********************************************************
C COMPUTATION OF PROPERTIES AFTER FINDING V AND ANGLE
C *********************************************************
SUBROUTINE INFO(U,D, FLOW, GRID, SCON, SIDE)
INTEGER U, D, FLOW, J, I, SIDE, K, C
REAL GRID(0:10,0:10,1:11), SCON
REAL CNAV, GAMMA, GAMMB
REAL A(1:10,1:3), B(1:6,1:3), NO, NC, NW, WH, FI, P, MR, HR
REAL SA, SB, SC, PI, SO(1:10), W(1:10), CP(1:10), NDTOT

COMMON A, B
COMMON /STAT/ NO, NC, NW, WH, FI, P, MR, HR, SA, SB, SC
COMMON /SPEC/ SO, N, NDTOT
IF (SIDE .EQ. 1) THEN
K = 1
C = 0
ELSE
K = 0
C = 1
END IF

IF (FLOW .EQ. 1) THEN
GAMMA = GRID(U-K,D-C,7)
CALL CHEM (GRID(U,D,6),GRID(U,D,8),GRID(U,D,9),GRID(U-K,D-C,5),
& GRID(U-K,D-C,8),GRID(U-K,D-C,9), GAMMA, GRID(U,D,3), SCON)

GRID(U,D,6) = ASIN(1.0/GRID(U,D,5))
GRID(U,D,7) = GAMMA
ELSE
GAMMA = GRID(U-K,D-C,7)
GAMMB = 100.0
DO WHILE(ABS(1.0-GAMMA/GAMMB) .GT. 0.0001)
GAMMB = GAMMA
CALL MACH(GRID(U,D,5), GRID(U-K,D-C,5), GRID(U,D,3),GAMMB)
GRID(U,D,6) = ASIN(1.0/GRID(U,D,5))
GRID(U,D,8) = (1+(GAMMB-1.0)/2.0*GRID(U-K,D-C,8)**2)
& * GRID(U-K,D-C,8)
GRID(U,D,8) = GRID(U,D,8)/(1.0+(GAMMB-1.0)/2.0*GRID(U,D,5)**2)
CPAV = 0.0
DO 6008 J = 1, 10, 1
CP(J) = A(J,2) + 2.0 * A(J,3) * GRID(U,D,8)/1000.0
CPAV = CPAV + CP(J) * N(J)
6008 CONTINUE
CPAV = CPAV / NTOT
GAMMA = CPAV / (CPAV - 1.986)
END DO
GRID(U,D,7) = GAMMA
GRID(U,D,9) = GRID(U,D,7) / (GRID(U,D,7) - 1.0)
GRID(U,D,9) = GRID(U-K,D-C,9)/
& ((GRID(U,D,8)/GRID(U-K,D-C,8))**GRID(U,D,9))
END IF
END
Appendix D

SUBROUTINES FOR THE NOZZLE

C **********************************************************************
C THIS ROUTINE COMPUTES ADDITIVE DRAG FOR AN OVER EXPANDED NOZ
C **********************************************************************

SUBROUTINE DRAG(XMAX,YMAX,DD,P1,M1,ALF,FD)
INTEGER I, J, K
REAL X, Y, DX, DY, DS,ANG,DD,P1,M1, P, XMAX,YMAX,FX,ALF,FD
REAL A1, A2
A1 = TAN(DD)
A2 = (YMAX - A1 * XMAX) / (XMAX * XMAX)
DX = XMAX / 300.0
X = 0.0
Y = 0.0
FX = 0.0
DO WHILE (X .LE. XMAX)
  X = X + DX
  DY = (A1 * X + A2 * X * X) - Y
  ANG = ATAN(DY/DX)
  DS = SQRT(DX * DX + DY * DY)
  P = P1 * (1.2*1.4*M1*SIN(ANG)**2.0+1.0)
  FX = (P - P1)* DS * SIN(ANG-ALF) + FX
  Y = Y + DY
END DO
FD = FX
END

114
SUBROUTINE VERGE(P1,P2,GAM,M1,M2,ALF,DD)
INTEGER NSTEP, ICOUNT, I, J
REAL P1,P2,GAM,M1,M2,ALF, DD, FA, FB, F, Y
REAL DA, DB, V1, V2

Y = GAM
V1 = SQRT((Y+1)/(Y-1))*ATAN(SQRT((Y-1)/(Y+1)*(M1*M1-1.0)))
V1 = V1 - ATAN(SQRT(M1*M1-1.0))
Y = 1.4
V2 = SQRT((Y+1)/(Y-1))*ATAN(SQRT((Y-1)/(Y+1)*(M2*M2-1.0)))
V2 = V2 - ATAN(SQRT(M2*M2-1.0))

FA = 1.0
FB = 1.0
F = 1
NSTEP = 1
ICOUNT = 0
DA = 0.5
DB = 0.11
DO WHILE((FA * FB) .GE. 0.0)
DA = DA + 0.1
DB = DB - 0.1
CALL FORM(V1,V2,M1,M2,P1,P2,GAM,ALF,DA,FA)
CALL FORM(V1,V2,M1,M2,P1,P2,GAM,ALF,DB,FB)
END DO

DO WHILE((ABS(F) .GT. 0.0001) .AND. (NSTEP .LT. 50))
DD = ( DA * FB - DB * FA)/(FB - FA)
CALL FORM(V1,V2,M1,M2,P1,P2,GAM,ALF,DD, F)
IF((F * FB) .GE. 0.0) THEN
DB = DD
FB = F
ICOUNT = ICOUNT + 1
IF (ICOUNT .GE. 2.0) THEN
DD = (DA + DB)/2.0
CALL FORM(V1,V2,M1,M2,P1,P2,GAM,ALF,DD, F)
IF ((F * FA) .GE. 0.0) THEN
DA = DD
FA = F
ICOUNT = 0
END IF
END IF
ELSE
DA = DB
FA = FB
DB = DD
FB = F
END IF
NSTEP = NSTEP + 1
END DO
END

C ******************************************
C FIND ERROR IN PRESSURE CALCULATIONS
C ******************************************
SUBROUTINE FORM(V1,V2,M1,M2,P1,P2,GAM,ALF,DD,F)
INTEGER I, J
REAL V1, V2, M1, M2, P 1, P2, GAM, ALF, DD, F, Y
V3A = V1 + DD
V3B = V2 - DD - ALF
CALL MACH(M3A, M1, V3A, GAM)
CALL MACH(M3B, M2, V3B, 1.4)
Y = GAM
P3A = (1.0+(Y-1.0)/2.0*M1*M1)/(1.0+(Y-1.0)/2.0*M3A*M3A)
P3A = P1*(P3A**(Y/(Y-1.0)))
Y = 1.4
IF((DD+ALF) .GE. 0.0) THEN
CALL SHOCK((DD+ALF),M2,SANG)
CALL RATIO(M2,SANG,DD,M3B,PR,DR,TR)
P3B = P2 * PR
ELSE
P3B = (1.0+(Y-1.0)/2.0*M2*M2)/(1.0+(Y-1.0)/2.0*M3B*M3B)
P3B = P2*(P3B**((Y/(Y-1.0)))
END IF
F = P3A - P3B
END

C **********************************************
C ITERATIVE SCHEME TO FIND M GIVEN V
C **********************************************

SUBROUTINE MACH(M, MS, V, Y)
REAL M, MS, V, Y, MA, MB, FA, FB, E, F
INTEGER N, I, J, NSTEP, ICOUNT
FA = 1.0
FB = 1.0
F = 1
MA = MS
MB = MS + 3.0
E = 0.00001
NSTEP = 1
ICOUNT = 0
DO WHILE((FA * FB) .GE. 0.0)
MB = MB + 0.1
MA = MA - 0.1
CALL ERRORM(Y, V, MA, FA)
CALL ERRORM(Y, V, MB, FB)
END DO
DO WHILE ( (ABS(F) .GT. E) .AND. (NSTEP .LT. 50))
M = (MA * FB - MB * FA) / (FB - FA)
call ERRORM(Y, V, M, F)
IF((F * FB) .GE. 0.0) THEN
MB = M
FB = F
ICOUNT = ICOUNT + 1
IF ( ICOUNT .GE. 2.0) THEN
M = ( MA + MB) /2.0
 call ERRORM(Y, V, M, F)
IF((F * FA) .GE. 0.0) THEN
MA = M
FA = F
ICOUNT = 0
END IF
END IF
ELSE
MA = MB
FA = FB
MB = M
FB = F
END IF
NSTEP = NSTEP + 1
END DO
IF (NSTEP .GT. 45) THEN
PRINT *, 'NO CONVERGENCE IN SHOCK ANGLE CALCULATION'
END IF
END

C ************************************************
C FIND ERROR IN THE APPROXIMATED M
C ************************************************

SUBROUTINE ERRORM(Y, V, M, F)
REAL Y, V, M, F
IF (M .LE. 1.0) M = 1.0
IF (Y .LE. 1.0) Y = 1.01
F = SQRT((Y+1)/(Y-1)) * ATAN(SQRT((Y-1)/(Y+1) * (M*M-1)))
F = V - (F - ATAN(SQRT(M*M-1)))
END

C ***************************************************
C CHEM
C ***************************************************
SUBROUTINE CHEM(M2, T2, P2, M1, T1, P1, GAMMA, MU, SCON)
REAL M2, T2, P2, M1, T1, P1, GAMMA, MU, SCON
REAL FA, FB, F, XA, XB, X, ER, D1
REAL A(1:10,1:3), B(1:6,1:3), NO, NC, NW, NH, FI, P, MR, HR
REAL SA, SB, SC, SO(1:10), N(1:10), NTOT, PP
INTEGER NSTEP, ICOUNT

COMMON A, B
COMMON /STAT/ NO, NC, NW, NH, FI, P, MR, HR, SA, SB, SC
COMMON /SPEC/ SO, N, NTOT
FA = 1.0
FB = 1.0
F = 1
XA = GAMMA
XB = GAMMA + 0.1
ER = 0.00001
NSTEP = 1
ICOUNT = 0
DO WHILE((FA * FB) .GE. 0.0)
XA = XA - 0.05
XB = XB + 0.05
CALL CALC(M2, T2, P2, M1, T1, P1, XA, MU)
PP = P2 / 101300.0
CALL ERRO(T2, PP, D1, SCON, FA)

119
CALL CALC(M2, T2, P2, M1, T1, P1, XB, MU)
PP = P2 / 101300.0
CALL ERRO(T2, PP, D1, SCAM, FB)
END DO

DO WHILE ((ABS(F) .GT. ER) .AND. (NSTEP .LT. 50))
X = (XA * FB - XB * FA) / (FB - FA)
CALL CALC(M2, T2, P2, M1, T1, P1, X, MU)
PP = P2 / 101300.0
CALL ERRO(T2, PP, D1, SCAM, F)
IF((F*FB) .GE. 0.0) THEN
XB = X
FB = F
ICOUNT = ICOUNT + 1
IF ( ICOUNT .GE. 2.0) THEN
X = (XA + XB) / 2.0
CALL CALC(M2, T2, P2, M1, T1, P1, X, MU)
PP = P2 / 101300.0
CALL ERRO(T2, PP, D1, SCAM, F)
IF((F*FA) .GE. 0.0) THEN
XA = X
FA = F
ICOUNT = 0
END IF
END IF
ELSE
XA = XB
FA = FB
XB = X
FB = F
END IF
NSTEP = NSTEP + 1
END DO
GAMMA = X
END
C CALC MACH, P, AND T
C
SUBROUTINE CALC(M2, T2, P2, M1, T1, P1, GAM, MU)
REAL M2, T2, P2, M1, T1, P1, GAM, MU
CALL MACH(M2, M1, MU, GAM)
T2 = (1.0+(GAM-1.0)/2.0*M1*M1)*T1/(1.0+(GAM-1.0)/2.0*M2*M2)
P2 = P1 / ((T1/T2)**(GAM/(GAM-1.0)))
END

C GIVE INITIAL STATE FIND CONDITION AFTER EQUILIBRIUM FLOW
C
SUBROUTINE EQUIL(T1, P1, T2, P2, DP, LIMIT, REC, GAMMA, SCON, & UCE, HRO)
REAL T1, P1, T2, P2, DP, LIMIT, GAMMA, SCON, UCE, HRO
REAL DP, N(1:10), NTOT, SO(1:10), VEL
REAL NO, NC, NN, NH, FI, P, MR, HR, SA, SB, SC
INTEGER I, J, K
COMMON /STAT/ NO, NC, NN, NH, FI, P, MR, HR, SA, SB, SC
COMMON /SPEC/ SO, N, NTOT
OPEN(UNIT = 2, FILE='NOZ2.DAT', STATUS='NEW')
FORMAT (2X F10.5,F10.5,X F10.5,F10.5)
P2 = P1
T2 = T1
COND = 1
WRITE(UNIT=2, FMT=3) T1, P1, UCE, (1.0/NO)
DO WHILE(COND .EQ. 1)
P2 = P2 - DP
CALL TFIND(T2, P2, GAMMA, SCON)
VEL = SQRT(UCE*UCE-(HR-HRO)*4180/0.5/MR)
IF (ABS((N(7) + N(8))/NTOT) .LE. REC) THEN
COND = 2
END IF
IF (P2 .LE. LIMIT) THEN
  COND = 2
END IF

WRITE(UNIT=2,FMT=3) T2, P2, VEL, (1.0/N0)
END DO
CLOSE(UNIT=2)
END

C ****************************************************
C GIVEN PRESSURE ITERATE IN TEMP UNTIL ENTROPY IS MATCHED
C ****************************************************

SUBROUTINE TFIND(T2, P2, GAMMA, SCON)

REAL T2, P2, GAMMA, SCON

REAL FA, FB, F, XA, XB, EPS, X

INTEGER ICOUNT, ISTEP

REAL A(1:10,1:3), B(1:6,1:3), NO, NC, NW, NE, FI, P, MR, HR

REAL SA, SB, SC

COMMON A, B

COMMON /STAT/ NO, NC, NW, NE, FI, P, MR, HR, SA, SB, SC

FA = 1.0
FB = 1.0
F = 1
XA = T2 - 200.0
XB = T2 + 10.0
EPS = 0.001

ISTEP = 1
ICOUNT = 0

DO WHILE ((FA*FB) .GE. 0.0)
  CALL ERRO(XA, P2, GAMMA, SCON, FA)
  CALL ERRO(XB, P2, GAMMA, SCON, FB)
  XA = XA - 100.0
  XB = XB + 10.0
END DO
DO WHILE ((ABS(F) .GT. EPS) .AND. (ISTEP .LT. 50))
  X = (XA * FB - XB * FA) / (FB - FA)
  CALL ERRO(X, P2, GAMMA, SCON, F)
  IF ((F * FB) .GE. 0.0) THEN
    XB = X
    FB = F
    ICOUNT = ICOUNT + 1
    IF (ICOUNT .GE. 2.0) THEN
      X = (XA + XB) / 2.0
      CALL ERRO(X, P2, GAMMA, SCON, F)
    END IF
    END IF
  ELSE
    XA = XB
    FA = FB
    XB = X
    FB = F
    END IF
  ISTEP = NSTEP + 1
END DO
T2 = X
END

C ******************************************************
C DIFFERENCE IN ENTROPY TO CONSTANT
C ******************************************************
SUBROUTINE ERRO(T2, P2, GAMMA, SCON, F)
REAL T2, P2, GAMMA, SCON, F, N(1:10), NTOT
REAL CPAV, CP(1:10), D1, D2, TH
INTEGER ITER, I, J
REAL A(1:10,1:3), B(1:6,1:3), NO, NC, NN, NH, FI, P, MR, HR
REAL SA, SB, SC, SO(1:10), STOT, S(1:10)

COMMON A, B
COMMON /STAT/ NO, NC, NN, NH, FI, P, MR, HR, SA, SB, SC
COMMON /SPEC/ SO, N, NTOT
P = P2
CALL ERRC(T2, D1, N, NTOT, D2)
TH = T2 / 1000.0
CPAV = 0.0
DO 6400 I = 1, 10, 1
CP(I) = A(I,2) * 2.0*A(I,3) * TH
CPAV = CPAV + CP(I) * N(I)
6400 CONTINUE
CPAV = CPAV/ NTOT
GAMMA = CPAV / (CPAV - 1.986)

STOT = 0.0
DO 6950 I = 1, 10, 1
IF (N(I) .LT. 0.0000007) THEN
S(I) = 1.0
ELSE
S(I) = SO(I) - 1.986*LOG(P*N(I)/NTOT) + A(I,2)*LOG(TH)
S(I) = S(I) + 2.0 * A(I,3) * (TH - 1.0)
END IF
STOT = STOT + N(I) * S(I)
6950 CONTINUE
HR = SA + SB * TH + SC * TH * TH
F = SCOM - STOT
END

C *******************************************************
C GRID GRAPH
C *******************************************************
SUBROUTINE GRIDGRAPH(XP, YP, LINE, POINTS)
REAL XP(1:500), YP(1:500)
INTEGER LINE, POINTS(1:20)
CHARACTER*8 PLTITL
INTEGER NLINE, INDGRR, N(1:3,1:200), I(1:200), J
PLTITL = 'X-Y'
NLINE = LINE
INDGRR = 31
DO 6910 J = 1, LINE, 1
   N(1,J) = POINTS(J)
   N(2,J) = 1
   N(3,J) = 3
   I(J) = 2
6910 CONTINUE
CALL GR_LINE_BETTER(I, NLINE, PLTITL, INDGRR, XP, YP, N)
END
Appendix E

SHOCKS, BOUNDARY LAYER, AND CHEMICAL EQUILIBRIUM ROUTINES

C ******************************************************************************
C SUBROUTINE FOR COMPUTING SHOCK ANGLE GIVEN TURNING ANGLE
C ******************************************************************************

SUBROUTINE SHOCK(A, M, X)
REAL A, M, X, XA, XB, FA, FB, E, Y, F
INTEGER N, I, J, NSTEP, ICOUNT
FA = 1.0
FB = 1.0
F = 1
Y = 1.4
XA = 1.0/M + (Y + 1) * M * M + A/(4.0 * (M * M - 1))
XB = 0.9
E = 0.00001
NSTEP = 1
ICOUNT = 0
DO WHILE((FA * FB) .GE. 0.0)
    XB = XB + 0.1
    CALL ERROR(A, M, XA, FA)
    CALL ERROR(A, M, XB, FB)
END DO

DO WHILE ( (ABS(F) .GT. E) .AND. (NSTEP .LT. 50))
\[
X = \frac{(X_A \cdot FB - X_B \cdot FA)}{(FB - FA)}
\]

CALL ERROR(A, N, X, F)

IF((F \cdot FB) \geq 0.0) THEN

\[
X_B = X
\]

FB = F

ICOUNT = ICOUNT + 1

IF (ICOUNT \geq 2.0) THEN

\[
X = \frac{(X_A + X_B)}{2.0}
\]

CALL ERROR(A, N, X, F)

IF((F \cdot FA) \geq 0.0) THEN

\[
X_A = X
\]

FA = F

ICOUNT = 0

ENDIF

ENDIF

ELSE

\[
X_A = X_B
\]

FA = FB

\[
X_B = X
\]

FB = F

ENDIF

NSTEP = NSTEP + 1

END DO

IF (NSTEP \gt 45) THEN

PRINT *, 'NO CONVERGENCE IN SHOCK ANGLE CALCULATION'

ENDIF

ENDIF

END

C ******************************************************

C FIND ERROR IN THE APPROXIMATED ANGLE

C ******************************************************

SUBROUTINE ERROR(A, M, X, F)

REAL A, M, X, F, A2, Y

Y = 1.4

A2 = M \cdot M \cdot \sin(X) \cdot \sin(X) - 1.0
A2 = A2 / (1.0 + M * M * ((Y+1)/2 - SIN(X) * SIN(X))) / TAN(X)
F = A2 - TAN(A)

END

C *****************************************************
C FIND ALL THE RATIOS ACROSS THE SHOCK
C *****************************************************

SUBROUTINE RATIO(M1, X, A, M2, PR, DR, TR)
REAL Mi, X, A, M2, PR, DR, TR, Mn, Mt, Y, M2T, M2N
Y = 1.4
Mn = M1 * SIN(X)
Mt = M1 * COS(X)
PR = (1.0 - Y + 2 * Y * Mn * Mn) / (Y + 1.0)
DR = (Y + 1.0) * Mn * Mn / (2.0 + (Y - 1.0) * Mn * Mn)
TR = PR / DR
M2T = Mt * SQRT(1.0 / TR)
M2N = SQRT(((Y-1.0) * Mn * Mn + 2)(2 * Y * Mn * Mn - (Y -1.0)))
M2 = SQRT(M2T * M2T + M2N * M2N)
END

C *****************************************************
C ROUTINE TO CALCULATE BOUNDARYLAYER PARAMETERS
C *****************************************************

SUBROUTINE LAYER(LEN, TO, TI, M, PI, S, DEL, LAM, THK)
REAL LEN, TO, TI, M, PI, S, DEL, THK(0:9)
REAL OI, XO, DX, REY, REYIX, REYIT, VERG, CF, B1, B2, A, B
REAL DI, MUI, UI, BOT, K, X, TW, Z, KK
INTEGER I, J, LAM

X = LEN
TW = TO
TW = TW / TI
OI = 1 + 0.2 * M * M
DI = PI / (287 * TI)
MUI = 0.000000145 * TI**1.5 / (TI + 110)
UI = M * SQRT(1.4 * 287 * TI)
REY = DI * UI / WUI
XO = 0.0
IF (LAM .EQ. 1) THEN

BOT = 0.5 + 0.5 * TW * 0.195*(OI - 1.0)
BOT = BOT**((1.0/6.0)
DX = SQRT(REY) * BOT * THK(1)
DX = (DX / (3.373 + 0.66*(OI-1) + 1.215 * TW))**2.0
X = X + (DX - XO)
THK(1) = (2.385 + 0.467*(OI-1) + 1.215*TW)/BOT*SQRT(2.0*X/REY)
THK(2) = 0.467 / BOT * SQRT(2.0 * X / REY)
THK(3) = (1.215 * TW + 0.467*(OI-1)) / BOT * SQRT(2.0*X / REY)
THK(4) = 0.664 / (BOT * SQRT(REY * X))
THK(5) = THK(3) / THK(2)
THK(8) = THK(3) - THK(1) / 2.0
ELSE
B1 = SQRT((OI - 1.0) / TW)
B2 = OI / TW - 1.0
THK(5) = TW + OI
IF (THK(2) .NE. 0.0) THEN
REYIT = 1000000
DO 600 I = 1, 5, 1
THK(3) = THK(5) * THK(2)
THK(1) = 0.406 * (LOG10(TW+OI))**0.9
THK(1) = THK(1) - 0.037 * LOG10(REYIT/100000.0)
THK(1) = THK(3) / THK(1)
REYIT = REY * THK(1)
600 CONTINUE
CF = 0.03
VERG = 1.0
DO WHILE (VERG .GT. 0.0001)
Z = CF
CF = 5.6 * LOG10(REYIT * Z / (TW**(7.0/6.0)))+7.4
\[ CF = \left( \text{ASIN}(A) + \text{ASIN}(B) \right) / (B1 \times \sqrt{TW} \times CF) \]
\[ \text{VERG} = \text{ABS}(1.0 - CF / Z) \]
\[ \text{END DO} \]
\[ CF = 2 \times CF \times CF \]
\[ \text{REYIX} = \left( \frac{\text{ASIN}(A) + \text{ASIN}(B)}{\sqrt{CF \times (O1 - 1.0)}} - 1.7 \right) / 4.15 \]
\[ \text{REYIX} = \exp(\text{REYIX}) \times TW^{(2.0/3.0)} / CF \]
\[ DX = \text{REYIX} / \text{REY} \]
\[ X = X + (DX - XO) \]
\[ \text{END IF} \]
\[ CF = 0.001 \]
\[ \text{VERG} = 1.0 \]
\[ \text{DO WHILE (VERG > 0.0001)} \]
\[ Z = CF \]
\[ CF = 4.15 \times \log_{10}(\text{REY} \times X \times Z / (TW^{(2.0/3.0)})) + 1.7 \]
\[ CF = \left( \frac{\text{ASIN}(A) + \text{ASIN}(B)}{CF} \right) \]
\[ CF = CF \times CF / (OI - 1.0) \]
\[ \text{VERG} = \text{ABS}(1.0 - CF / Z) \]
\[ \text{END DO} \]
\[ \text{REYIT} = 0.0 \]
\[ \text{THK}(4) = CF \]
\[ \text{REYIT} = \left( \frac{\text{ASIN}(A) + \text{ASIN}(B)}{(B1 \times \sqrt{\text{THK}(4) \times TW} / 2.0)} \right) \]
\[ \text{REYIT} = \exp((\text{REYIT} - 7.4) \times 2.0326/5.6) \]
\[ \text{REYIT} = \text{REYIT} \times TW^{(7.0/6.0)} / \sqrt{\text{THK}(4)/2.0} \]
\[ \text{THK}(1) = \text{REYIT} / \text{REY} \]
\[ \text{THK}(3) = 0.406 \times \log_{10}(TW + OI)^{0.9} \]
\[ \text{THK}(3) = \left( \text{THK}(3) - 0.037 \times \log_{10}(\text{REY} \times \text{THK}(1)/100000.0) \right) \times \text{THK}(1) \]
\[ \text{THK}(2) = \text{THK}(3) / (TW + OI) \]
\[ KK = CF / (2.0 \times 0.41 \times 0.41) \]
\[ \text{THK}(8) = \left( KK \times \left[ (O1+TW)/2.0-1.0 \right] + \sqrt{\text{KK}} \right) / (1+KK \times (O1-1.0)) \]
\[ \text{THK}(8) = \text{THK}(3) - \text{THK}(1) \times \text{THK}(8) \]
\[ \text{END IF} \]
\[ \text{THK}(6) = \text{ATAN}(\text{THK}(3) / X) \]
\[ \text{THK}(7) = \text{THK}(3) / \text{THK}(6) \times (1.0 - TW / OI) \]
\[ \text{END} \]
SUBROUTINE ENTHALPY(X, H, N, NTOT, TGUSS)
REAL X, H, N(1:10), NTOT, TGUSS
REAL FA, FB, F, XA, XB, EPS
INTEGER ICOUNT, NSTEP
REAL A(1:10,1:3), B(1:6,1:3), NO, NC, MM, MH, FI, P, MR, HR
REAL SA, SB, SC
COMMON A, B
COMMON /STAT/ NO, NC, MM, MH, FI, P, MR, HR, SA, SB, SC
FA = 1.0
FB = 1.0
F = 1
IF (TGUSS .GT. 0.0) THEN
XA = TGUSS - 200.0
XB = TGUSS + 300.0
ELSE
XA = 2200.0
XB = 2700.0
END IF
EPS = 0.01
NSTEP = 1
ICOUNT = 0
DO WHILE ((FA*FB) .GE. 0.0)
CALL ERRC(XA, FA, N, NTOT, H)
CALL ERRC(XB, FB, N, NTOT, H)
XA = XA - 100.0
XB = XB + 100.0
END DO
DO WHILE ((ABS(F) .GT. EPS) .AND. (NSTEP .LT. 50))
X = (XA * FB - XB * FA) / (FB - FA)
CALL ERRC(X, F, N, NTOT, H)
IF ((F * FB) .GE. 0.0) THEN
XB = X
FB = F
ICOUNT = ICOUNT + 1
IF (ICOUNT .GE. 2.0) THEN
X = (XA + XB) / 2.0
CALL ERRC(X, F, N, NTOT, H)
IF ((F * FA) .GE. 0.0) THEN
XA = X
FA = F
ICOUNT = 0
END IF
ELSE
XA = XB
FA = FB
XB = X
FB = F
END IF
NSTEP = NSTEP + 1
END DO
IF (NSTEP .GT. 45) THEN
PRINT *, 'NO CONVERGENCE IN ENTHALPY CALCULATION'
END IF
END
C ************************************************
C SPECIFIC ENTHALPY
C ************************************************
SUBROUTINE ERRC(X, F, N, NTOT, H)
REAL X, F, N(1:10), NTOT, H
REAL T, TH, G1, G2, BB, C, SG, PF, NNO, NNC, NNN, NH
REAL S1, S2, S3, K(1:6), WOLD(1:10), M
REAL F6, PAR, PP, QQ, RAD, BS, XX, KK
INTEGER ITER, I, J
REAL A(1:10,1:3), B(1:6,1:3), NO, NC, NW, NH, FI, P, MR, HR
REAL SA, SB, SC
COMMON A, B
COMMON /STAT/ NO, NC, NW, NH, FI, P, MR, HR, SA, SB, SC
IF(T .LT. 300.0) THEN
  T = 300.0
END IF
T = X
TH = T /1000
COND = 10
DO 2000 I = 1, 6, 1
  K(I) = EXP((-B(I,1)-B(I,2)*TE-B(I,3)*TH*TE)/T)
2000 CONTINUE
DO 2005 I = 1, 10, 1
2005 CONTINUE
C FUEL RICH CASE //////////////////////////////////////////////////
IF(FI .GE. 1.12) THEN
  ITER = 0
  DO 2010 I = 5, 9, 1
    N(I) = 0
2010 CONTINUE
DO WHILE((COND .NE. 1) .AND. (ITER .LE. 80))
  G1 = NO - NC - N(8) - N(6) - N(9) - 2.0 * N(5)
  G2 = (NW - N(7) - N(6)) / 2.0
  BB = (K(4) * (G1+G2) + NC - G1)/2.0/(K(4) - 1)
  C = K(4) * G1 + G2/ (K(4) - 1.0)
  SG = 1.0
  IF (K(4) .LT. 1) THEN
    SG = -1.0
  END IF
  N(2) = C /(BB + SG * SQRT(BB*BB - C))
  N(10) = (NW - N(9))/2.0
\( m(1) = g2 - m(2) \)

IF \( (m(2) \leq 0.0) \) THEN

\( m(3) = m(2) / (1.0 + k(4) * m(1)/m(2)) \)

END IF

\( m(4) = m(2) - m(3) \)

IF \( (m(2) < 0.0) \) THEN \( m(2) = 0.0 \)

IF \( (m(10) < 0.0) \) THEN \( m(10) = 0.0 \)

IF \( (m(1) < 0.0) \) THEN \( m(1) = 0.0 \)

IF \( (m(3) < 0.0) \) THEN \( m(3) = 0.0 \)

IF \( (m(4) < 0.0) \) THEN \( m(4) = 0.0 \)

\( ntot = 0.0 \)

DO 2020 \( i = 1, 10, 1 \)

\( ntot = ntot + m(i) \)

2020 CONTINUE

\( pf = \sqrt{p/ntot} \)

IF \( (m(1) \neq 0.0) \) THEN

\( m(5) = k(6) * pf * m(2)/m(1) \)

END IF

\( m(5) = m(5) * m(5) \)

\( m(6) = k(1) * \sqrt{m(5) * m(1)} \)

\( m(7) = k(2) * \sqrt{m(1)} / pf \)

\( m(8) = k(3) * \sqrt{m(5)} / pf \)

\( m(9) = k(5) * \sqrt{m(10) * m(5)} \)

\( ntot = 0.0 \)

DO 2030 \( i = 1, 10, 1 \)

\( ntot = ntot + m(i) \)

2030 CONTINUE

\( cond = 1 \)

\( i = 1 \)

DO WHILE ( \( i \leq 10 \) )

IF \( (m(i) \leq 0.0) \) THEN

\( m = 0.0 \)

ELSE

\( m = \text{ABS}((m(i) - mold(i)) / m(i)) \)
END IF
IF (M(I) .GT. 0.0001) THEN
END IF
END IF
I = I + 1
END DO
IF (COND .NE. 1) THEN
DO 2035 J = 1, 10, 1
NOLD(J) = I(J)
2035 CONTINUE
END IF
ITER = ITER + 1
END DO
END IF

C FUEL LEAN CASE ///////////////////////////////////////////////////////////////
IF( FI .LT. .9) THEN
ITER = 0
NNH = NH
NNC = NC
NNO = NO
NNN = NN
DO WHILE( (COND .NE. 1) .AND. (ITER .LE. 80))
N(3) = NNC
N(2) = NNH / 2.0
N(5) = (NNO - 2.0 * NNC -NNH / 2.0) / 2.0
N(10) = NNH / 2.0
IF (N(3) .LT. 0.0) N(3) = 0.0
IF (N(2) .LT. 0.0) N(2) = 0.0
IF (N(5) .LT. 0.0) N(5) = 0.0
IF (N(10) .LT. 0.0) N(10) = 0.0
NTOT = 0.0
DO 3020 I = 1, 10, 1
NTOT = NTOT + N(I)
3020 CONTINUE

PF = SQRT(P/NTOT)

IF (N(5) .NE. 0.0) THEN
N(1) = K(6) * PF * N(2) / SQRT(N(5))
ENDIF

IF (N(2) .NE. 0.0) THEN
N(4) = K(4) * N(3) * N(1)/N(2)
ENDIF

IF (N(1) .LT. 0.0) THEN
N(1) = 0.0
ENDIF

N(6) = K(1) * SQRT(N(5) * N(1))
N(7) = K(2) * SQRT(N(1)) / PF
N(8) = K(3) * SQRT(N(5)) / PF
N(9) = K(5) * SQRT(N(10)*N(5))

NTOT = 0
DO 3030 I = 1, 10, 1
NTOT = NTOT + N(I)
3030 CONTINUE

COND = 1
I = 1
DO WHILE ( I .LE. 10)
IF (N(I) .EQ. 0.0) THEN
M = 0.0
ELSE
M = ABS((N(I) - MOLD(I)) / N(I))
ENDIF
IF (N(I) .GT. 0.0001) THEN
IF (M .GT. 0.0001) THEN
COND = 10
I = 11
ENDIF
ENDIF
I = I + 1
END DO
IF (COND .NE. 1) THEN
DO 3035 J = 1, 10, 1
NOLD(J) = N(J)
3035 CONTINUE
NWH = NH - N(7) - 2.0 * N(1) - N(6)
NWC = NC - N(4)
NNO = NO - N(4) - N(8) - N(6) - N(9)
NWH = NH - N(9)
END IF
ITER = ITER + 1
END DO
END IF

C NEAR STOICHIOMETRIC CONDITIONS /////////////////
IF ((FI .LT. 1.12) .AND. (FI .GT. .9)) THEN
ITER = 0
NWH = NH
NNO = NO
NWC = NC
NWH = NH
NTOT = NWC + NWH / 2.0 + NWH/2.0
DO WHILE (((COND .NE. 1) .AND. (ITER .LE. 80))
F6 = K(6) * K(6) * P / NTOT
PAR = NNO - 2.0 * NWC - NWH/2.0 - 2.0 * F6
PP = 2.0 * NWH * F6 - PAR * PAR / 3.0
QQ = 2.0/ 27.0 * PAR**3
QQ = QQ - 2.0/3.0 * NWH * PAR * F6 - F6 * NWH * NWH/2.0
RAD = QQ * QQ / 4.0 + PP**3 / 27.0
IF (RAD .LT. 0.0) THEN
RAD = 0.0
END IF
BS = -QQ/2.0 + SQRT(RAD)
KK = ABS(BS)**0.3333
KK = SIGN(KK,BS)
BS = -QQ/2.0 - SQRT(RAD)
KK = KK + SIGN(ABS(BS)**0.3333),BS

N(1) = KK - PAR / 3.0

N(2) = NWH / 2.0 - N(1)

N(3) = NNC

IF (N(2) .NE. 0.0) THEN

N(4) = N(3) * K(4) * N(1) / N(2)

END IF

N(5) = (NNO - 2.0 * NNC - N(2)) / 2.0

IF(N(1) .LT. 0.0) N(1) = 0.0

IF(N(2) .LT. 0.0) N(2) = 0.0

IF(N(3) .LT. 0.0) N(3) = 0.0

IF(N(4) .LT. 0.0) N(4) = 0.0

IF(N(5) .LT. 0.0) N(5) = 0.0

N(10) = NWW / 2.0

IF(N(10) .LT. 0.0) N(10) = 0.0

NTOT = 0.0

DO 4020 I = 1, 10, 1

NTOT = NTOT + N(I)

4020 CONTINUE

PF = SQRT(P / NTOT)

N(6) = K(1) * SQRT(N(5) * N(1))

N(7) = K(2) * SQRT(N(1)) / PF

N(8) = K(3) * SQRT(N(5)) / PF

N(9) = K(5) * SQRT(N(10) * N(5))

NTOT = 0.0

DO 4030 I = 1, 10, 1

NTOT = NTOT + N(I)

4030 CONTINUE

COND = 1

I = 1

DO WHILE (I .LE. 10)

IF (N(I) .EQ. 0.0) THEN

M = 0.0

ELSE

M = ABS((N(I) - NOLD(I)) / N(I))
END IF
IF (M(I) .GT. 0.0001) THEN
IF (M .GT. 0.0001) THEN
  COND = 10
  I = 11
END IF
END IF
END IF
I = I + 1
END DO
IF (COND .NE. 0.0) THEN DO 4035 J = 1, 10, 1
  NOLD(J) = M(J)
4035 CONTINUE

  NHH = NH - N(7) - N(8)
  NNC = NC - N(4)
  NNO = NO - N(4) - N(8) - N(6) - N(9)
  NNH = NH - N(9)
END IF
ITER = ITER + 1
END DO
END IF
C CONTINUE ON AFTER GOING THROUGH APPROPRIATE CASE
SA = 0.0
SB = 0.0
SC = 0.0
DO 5000 I = 1, 10, 1
  SA = SA + N(I) * A(I,1)
  SB = SB + N(I) * A(I,2)
  SC = SC + N(I) * A(I,3)
5000 CONTINUE
H = (SA + SB * TH + SC * TH * TH) / MR
F = H - HR / MR

END