Estimating Stagnation-Point Radiative Heating Using Maslen's Inverse Method and High-Temperature Equilibrium Air Properties

A project present to
The Faculty of the Department of Aerospace Engineering
San Jose State University

in partial fulfillment of the requirements for the degree
Master of Science in Aerospace Engineering

By

Alexander A. Carlozzi

December 2014

approved by

Dr. Papadopoulos
Faculty Advisor

San José State
UNIVERSITY

The Designated Thesis Committee Approves the Thesis Titled

ESTIMATING STAGNATION-POINT RADIATIVE HEATING USING

MASLEN'S INVERSE METHOD AND HIGH-TEMPERATURE

EQUILIBRIUM AIR PROPERTIES

by

Alexander A. Carlozzi

APPROVED FOR THE DEPARTMENT OF AEROSPACE

ENGINEERING

SAN JOSE STATE UNIVERSITY

December 2014

Dr. Periklis Papadopoulos	Department of Aerospace Engineering
Dr. Nikos Mourtos	Department of Aerospace Engineering
Dr. Jorge Bardina	NASA Ames Research Center

ABSTRACT

ESTIMATING STAGNATION-POINT RADIATIVE HEATING USING MASLEN'S

INVERSE METHOD AND HIGH-TEMPERATURE EQUILIBRIUM AIR

PROPERTIES

by

Alexander A. Carlozzi

This paper attempts to estimate stagnation point radiative heat flux for hypervelocity reentry vehicles. To begin, a shock shape is assumed. Using the inverse method developed by Maslen, the body which supports such a shock is calculated. High temperature equilibrium theory that accounts for chemical reactions in air is incorporated into Maslen's Method. Using the calculated air temperature and density behind the bow-shock, radiative heat flux is then calculated at the stagnation point. When a non-grey transparent gas is assumed in the shock-layer, a radiative heat flux at the stagnation point on the vehicle with a nose radius equal to that of the Apollo Command Module is calculated to be $3.3 \times 10^8 /^2$, using a free-stream velocity of 36,000 feet per second at an altitude of 170,000 feet. This is approximately 18 % less than the flux calculated for an infinite slab of comparable gas conditions and shock-layer thickness. When an emitting and absorbing shock-layer are modeled with radiative cooling, agreement with rigorous analysis from previously published data are in agreement on the order of 1.2 to 13.7 % difference.

ACKNOLEDGEMENTS

Tables of Contents

List of Figures

List of Tables

Chapter 1. Introduction

There are three mechanisms for transferring heat from one location to another. Heat transfer by conduction occurs through the collision of particles due to a temperature gradient. If a person were to hold a metal rod at one end and place the other end in a pot of boiling water, the end being held by the person gets hotter and hotter. Atoms on the "warmer" end have on average more kinetic energy than the atoms on the "cooler" end. The atoms with more kinetic energy will collide with their "cooler" neighbors thereby transferring some of their kinetic energy. Although the atoms in the metal rod do not move from one location to another, their energy does. This is the mechanism of heat transfer by conduction.

Heat transfer by means of mass motion of a fluid from one physical location to another is the mechanism of convection. Heat is transferred from the Earth's surface to its atmosphere by convection. Thermal expansion of the air at the surface will cause a change in density. This change in density causes the air to rise through the buoyancy effect. This is an example of "free/natural" convection. These thermal updrafts are what allow unpowered glider aircraft to climb and sustain flight.

The third mechanism for transferring heat from one location to another is radiation. A person can walk outside and feel the warmth of the sun's radiation, or feel heat from standing in from of a fireplace. Electromagnetic waves or light will transfer heat from one location to another. An object that has a temperature above absolute zero will emit electromagnetic radiation. An object at room temperature will emit light with

wavelengths greater than the wavelengths of visible light. A person at 98.6 degrees Fahrenheit will emit infrared radiation, which cannot be seen by the human eye. However, as the temperature of an object is raised, the wavelengths of the emitted light will decrease. An incandescent lamp filament at 3000 degrees Celsius will emit enough radiation in the "visible" spectrum to appear "white-hot" to the human eye. It is this third mechanism of heat transfer that relates to the topic at hand.

The rate at which energy is radiated from an object is proportional to its surface area, and its temperature raised to the fourth power. It is also proportional to the nature of the objects surface or its emissivity. This is a dimensionless constant with a value between 0 and 1. Young and Freedman [1] define emissivity as the ratio of the rate of radiation from a particular surface to the rate of radiation from an equal area of an ideal (blackbody) radiating surface at the same temperature. The rate of radiative heat transfer can be expressed by the equation below.

$$\underline{\quad\quad} = ^4 \tag{1.1}$$

This relationship is known as the Stefan-Boltzmann law, and is the Stefan-Boltzmann constant. Emissivity or "e" is large for dark surfaces and less for light surfaces. A body that can be considered an ideal emitter is also an ideal absorber. A surface that absorbs all the radiation incident on it is called an ideal "blackbody". The emissivity of a blackbody is equal to 1. For simplicity, this paper assumes a reentry vehicle experiencing radiative heating to have the properties of a blackbody.

2

Chapter 2.   Background

Why is radiative heating important to the design of a reentry vehicle or its thermal protection system (TPS) in modern Aerospace applications? To answer this question, the physics of hypersonic flows and high-temperature gas must be explored. A reentry body such as the Apollo command shown below in figure 1, will penetrate Earth's atmosphere at extremely high velocities and Mach numbers.
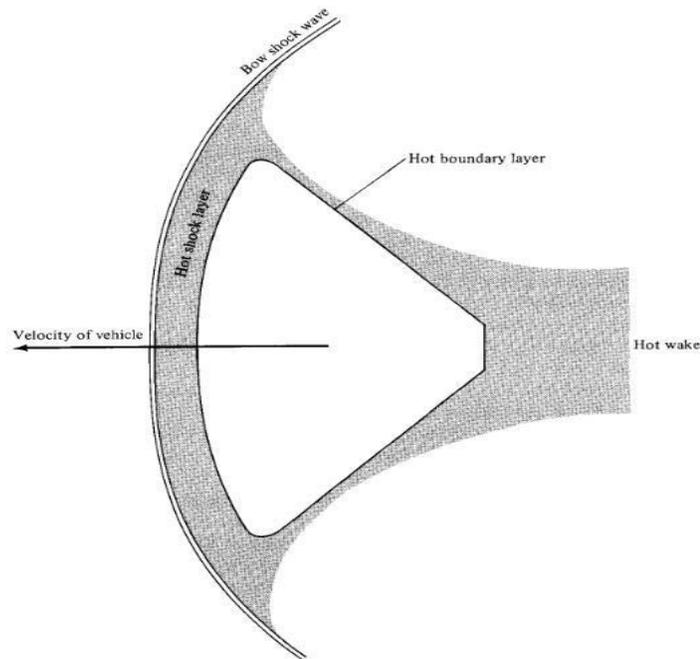

Figure 1. Apollo command module in a high temperature flow field [2].

From aerodynamic theory, a shockwave will form in front of the vehicle. Shock waves are thin regions of a flow where very severe changes in flow properties occur. For

example, as a fluid element moves through a shock wave, Mach number decreases, static pressure increases, static temperature increases, flow velocity decreases, total pressure decreases, and total temperature remains constant if the assumption of an ideal gas (intermolecular forces are ignored) is made. One might ask why shock waves form in supersonic/hypersonic flow. Consider gas molecules in a subsonic flow colliding with a sharp nosed object like a Pitot tube. These collisions set up a disturbance pattern in the flow. Weak pressure waves (sound waves) will communicate this disturbance to other areas of the flow. Because the flow is subsonic, these disturbances will be communicated upstream of the object to all regions of the flow. However, when the free-stream flow velocity is greater than the speed of sound, pressure disturbances that propagate away from an object at the speed of sound cannot move upstream! They coalesce at a small-finite distance from the front of the Pitot tube to form a shock-wave. Free-stream areas ahead of the shockwave cannot sense any disturbances from the object. Simply put, flow ahead of the shock wave has no idea the object is even there. The presence of the Pitot tube is communicated only to regions downstream of the shockwave. Hence, a shockwave can be thought of as a boundary between upstream regions of undisturbed flow, and downstream regions of disturbed flow. Solid objects in supersonic/hypersonic flow will always produce shockwaves.

Because the reentry velocity of the Apollo spacecraft is very large, the bow shockwave in front of the body will be very strong. This will produce extremely high static temperatures immediately behind the shock. As will be discussed later, a point in time on the Apollo reentry trajectory saw a free stream velocity of 36,000 feet per second

4

(ft/sec) at an altitude of 170,000 feet (ft). This produced an air temperature behind the normal part of the bow shock in excess of 11,000 K. At temperatures of this magnitude (hotter than the surface of the sun), air actually breaks down in a process called dissociation. Air is made up of mostly O_2 and N_2 molecules. These molecules break apart into O and N atoms and even ionize into O^+ and N^+ ions and electrons. This means the fluid behind the bow shock is a chemically reacting gas. Because the vehicle itself is bathed in a shock-layer of such extreme temperatures, it will inherently absorb large amounts of heat.

The aerodynamic heating (related to convective heat transfer mentioned above) of a vehicle is the result of friction in the boundary layer. Frictional forces that are responsible for skin friction drag also heat the air in the boundary layer. This "heating" of the boundary layer causes heat flow to the vehicle surface. For hypervelocity reentry vehicles, the problem of heating relates to the very survival of the vehicle, as well as its design and shape. In the analysis of aerodynamic heating, a dimensionless heat transfer coefficient called the Stanton Number is defined by equation 2.1 below.

$$= \frac{\dot{q}}{\rho_\infty V_\infty (h_0 - h)} \tag{2.1}$$

V_∞ and ρ_∞ are the free-stream velocity and density, h_0 is the total enthalpy, h is the enthalpy at the surface of the vehicle, and S is the vehicle's reference area. Equation 2.1 can be rewritten in the following form below.

$$\overline{} = (h - h_w) \tag{2.2}$$

5

Total enthalpy is defined by equation 2.3 below.

$$h_0 = h + \frac{V^2}{2} \tag{2.3}$$

Since the free-stream velocity is quite large, the free-stream static enthalpy is relatively small in comparison. It can be assumed for a hypervelocity reentry that total enthalpy can be expressed by equation 2.4 below.

$$h_0 = \frac{V^2}{2} \tag{2.4}$$

Because the surface temperature must be less than the melting temperature of the surface itself, and h_0 is associated with maximum temperatures behind the shockwave, it can be assumed that $h_0 \gg h \approx 0$. Using these assumptions, equation 2.2 can be expressed in the form of equation 2.5 below.

$$\frac{}{} = \frac{1}{2}^3 \tag{2.5}$$

Because there is a direct relationship between aerodynamic heating and skin friction (both caused by friction in the boundary layer) called Reynolds Analogy, the Stanton Number can be approximated as equation 2.6 below.

$$\approx \frac{1}{2} \tag{2.6}$$

is defined as the mean skin friction coefficient averaged over the vehicle's surface.

Using equation 2.6, equation 2.5 can be expressed as equation 2.7 below.

$$\underline{\quad} = \underline{\frac{1}{4}}\,^3 \tag{2.7}$$

As mentioned above, aerodynamic heating has a profound impact on the design and shape of a reentry vehicle. It is desirable to develop a relationship between aerodynamic heating and the general shape of the body. Assuming the case of a ballistic entry, Newton's second law of motion can be written as equation 2.8 below.

$$- \;= \quad \underline{\qquad} \quad \underline{\quad}^{\,*} = - \;\underline{\quad} = - \;\frac{1}{2}\,^2 \tag{2.8}$$

The chain rule from Calculus says that equation 2.8 can be written in the form below.

$$\underline{\quad} = \underline{\qquad}^{\infty} = \underline{\qquad}\left(- \;\frac{1}{2}\,^2 \;\right) \tag{2.9}$$

Equations 2.7 and 2.9 can be equated and simplified to form equation 2.10 below.

$$\underline{\qquad}\left(- \;\frac{1}{2}\,^2 \;\right) = \frac{1}{4}\,^3$$

$$= - \;\frac{1}{2}\;\underline{\qquad}^2 \tag{2.10}$$

Equation 2.10 can now be integrated from the beginning of the entry trajectory where $Q = 0$ and the free-stream velocity is the entry velocity, to the end of the trajectory where $Q =$ and the free-stream velocity is equal to zero. is expressed by equation 2.11 below.

$$= \frac{1}{2}\,\underline{\quad}\,(\frac{1}{2}\,^2) \tag{2.11}$$

Equation 2.11 says that the total heat absorbed by a reentry vehicle is related to its kinetic energy at entry, as well as the ratio of its skin friction drag to its total drag. The total drag of a non-lifting vehicle is expressed by equation 2.12 below.

$$= +$$

<div align="right">(2.12)</div>

is defined as pressure drag due to flow separation. Therefore, if total heat absorbed

by the vehicle is to be minimized, then equation 2.13 below needs to be minimized.

$$\underline{\quad} = \underline{\quad\quad} \tag{2.13}$$

A reentry vehicle that is slender with a sharp nose configuration will correspond

to a relationship expressed by equation 2.14 below.

$$\approx \qquad \underline{\quad} \approx 1 \tag{2.14}$$

A reentry vehicle that takes the shape of a blunt nosed body corresponds to equation
2.15.

$$\approx \qquad \underline{\quad} \ll 1 \tag{2.15}$$

Examining equations 2.14 and 2.15 leads to the obvious conclusion that to minimize

equations 2.13 and 2.11, a blunt nosed body must be selected. This is precisely why the

Apollo Command Module employed a blunt nose design. It should be remembered that

in equation 2.11 corresponds to total aerodynamic (convective) heating.

Therefore, a blunt nose design will minimize aerodynamic heating. When radiative

8

heating is explored in later sections, it will be shown that reducing the nose

radius actually minimizes the total radiative heat absorbed by the vehicle.

To answer the question of why radiative heating is so important to the design and

survival of hypervelocity reentry vehicles, figure 2 below shows a comparison between

convective and radiative heating versus free-stream velocity.
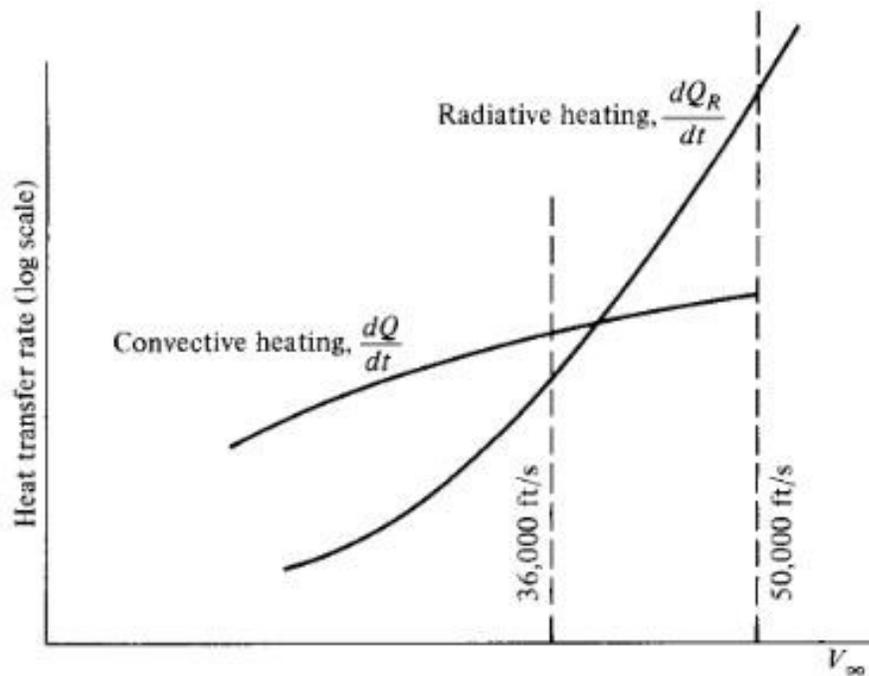
Figure 2. Comparing radiative and convective heating rates [2].

For low earth orbit (LEO) reentries, total heat absorbed by the body is comprised

primarily of convective heating. On the lunar return entries of Apollo where free-stream

velocities were on the order of magnitude of 36,000 ft/sec, the total heat rate absorbed

consisted of more than 30% radiative. As seen in figure 2, the rate of convective heating

is still greater than that of radiative on a Lunar return. On future Mars return entries

where free-stream velocities will be on the order of magnitude of 50,000 ft/sec, total heat

absorbed will be comprised primarily of radiative heating. At such high velocities, air

temperatures behind the normal portion of the bow-shock will exceed 11,000 K. This

means that the gas in the shock layer will radiate heat/energy in all directions. This

phenomenon is depicted in figure 3 below.
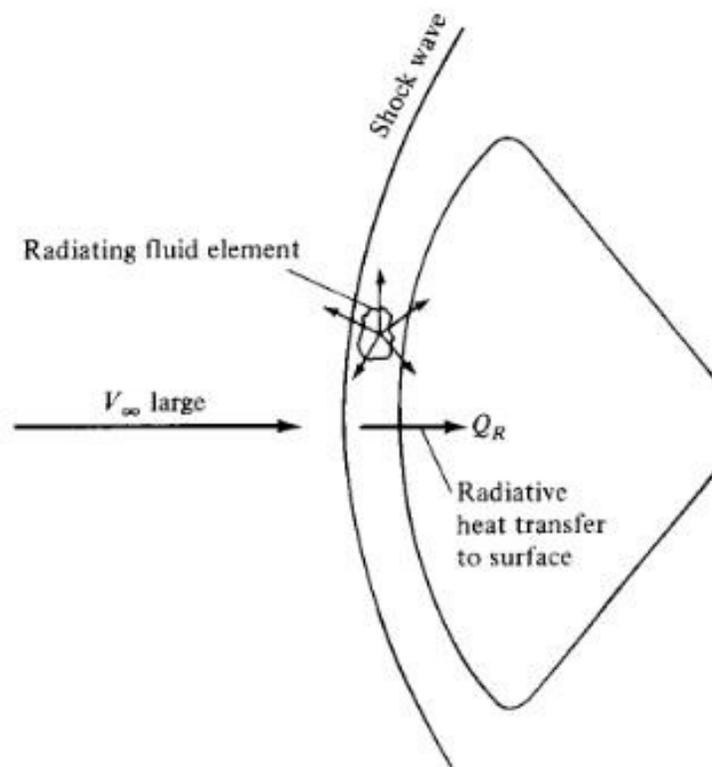
Figure 3. An entry vehicle experiencing radiative heating from the shock layer [2].

According to Anderson [2], radiative heat transfer rates are proportional to a power of
velocity on the order of V_∞^5 V_∞^{12}. This of course depends on variables such as air

density, free-stream velocity, and vehicle nose radius. The Galileo probe that entered

Jupiter's atmosphere in 1995, had an entry velocity of approximately 47.8 kilometers per

second, and was slowed to subsonic speeds in less than two minutes. The probe's 152 kg

thermal protection system or heat shield, lost approximately 80 kg of material during

Jovian entry. Moss and Simmonds [3] predicted the time varying contours for the shape

of the Galileo heat shield as shown below in figure 4.
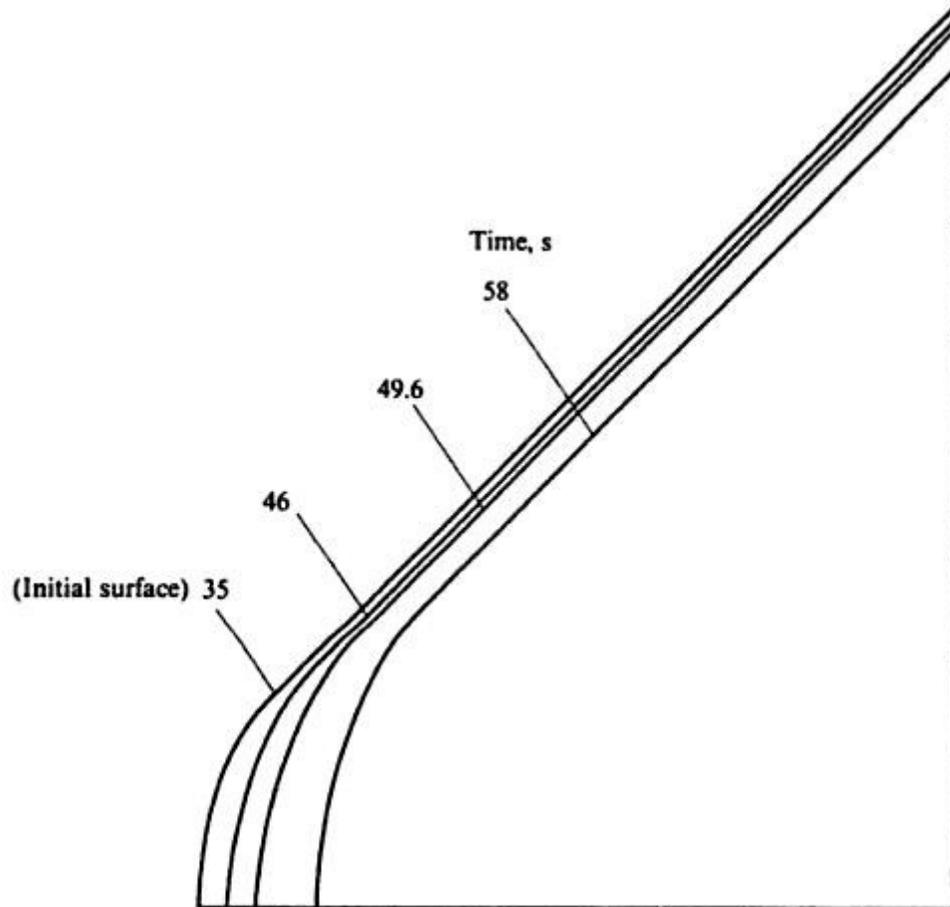
Figure 4. The predicted time varying contours of the Galileo probe [3].

The total heating experienced by the vehicle was comprised of approximately 95% radiative heating. It is also interesting to note that Galileo's heat shield at the beginning of its Jovian entry, made up almost half of its total mass. Approximately 53% of its thermal protection system was consumed. The rest was essentially very costly "dead-weight" which was never used. Weight which could have been put to better use in the form of additional scientific payload. Clearly, radiative heating will not only be a main source of consideration for engineers designing thermal protection systems for future interplanetary missions, but also increased knowledge and understanding of how to predict total radiative heating for aiding in properly sizing a vehicles thermal protection system is desired. The objective of this paper is to develop a simple and straight-forward technique that can accurately estimate radiative heating experienced by a hypervelocity reentry vehicle.

Chapter 3.   Radiative Heating

As mentioned earlier, when the air immediately behind a bow-shock is hot enough, it will radiate a considerable amount of energy in all directions. The temperature at which air will begin radiating a substantial amount of energy is approximately 10,000 K. When this happens, infinitesimal fluid elements in the flow-field actually loose energy because they are emitting radiation. The opposite can also happen where fluid elements will gain energy from the absorption of radiation that is emitted from other nearby fluid elements. When a shock layer is emitting and absorbing radiation, it has become a non-adiabatic flow-field. In such a flow-field, the total enthalpy is now a variable. This is a serious consequence of individual fluid elements emitting and absorbing radiation. Another consequence is that the dynamics of fluid flow and radiation are generally coupled. Simply put, radiative intensity is a function of air temperature and air density. Likewise, the flow-field properties are influenced by radiative intensity.

When studying radiative shock layers, there are generally two types of gases to consider. The first is called a transparent gas. In this case the gas will emit radiation, but not absorb it. It is assumed that all the radiation emitted from a radiative shock layer comprised of a transparent gas will escape to its surrounding environment. The second type of gas to consider is a self-absorbing gas that emits and absorbs radiation. In this case, some of the radiated energy will escape to the surrounding environment, while some will be absorbed and thus recaptured by the flow. The mathematical behavior of a

flow-field is heavily depended upon the radiative nature of the gas. For example, a flow-field comprised of a transparent gas is mathematically hyperbolic in terms of its behavior. Energy and flow information is not transmitted upstream of the shockwave. However, a flow-field comprised of a self-absorbing gas has a mathematically elliptic behavior. In this case radiation and flow-field information emitted from behind the bow-shock can be absorbed/received by fluid elements upstream of the bow-shock. A time-marching solution such as MacCormack's method would be required to compute such a flow-field.

Radiative energy that is transferred through a gas can be described by two quantities: radiative flux and radiative intensity. To help describe radiative intensity, consider a given amount of radiating gas shown below in figure 5.
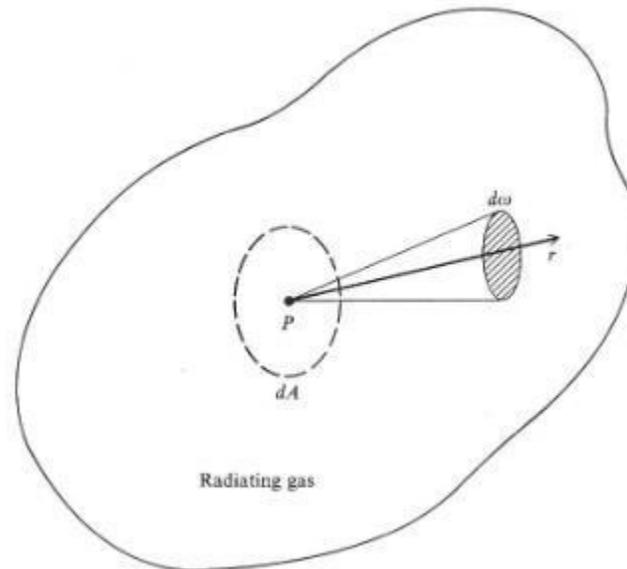
Figure 5. Radiative intensity for a given elemental volume of radiating gas [4].

First consider an infinitesimal area dA at point P, which is perpendicular to the direction of r. A solid angle is drawn about the r direction, and dE' is an infinitesimal amount of radiative energy in the frequency range between + that is emitted through dA during the infinitesimal time span dt from all directions contained in the solid angle . Specific radiative intensity is expressed by equation 3.1 below.

Simply put, radiative energy is transmitted in the direction of r across a unit area

perpendicular to r, per unit frequency, per unit solid angle. Radiative intensity must

always be referenced to a specific direction. The other term used to describe radiative

energy is called radiative flux. Flux can be used to describe numerous quantities such as

radiation, magnetism, momentum, etc. It is simply "something" per unit area per unit

time. Radiative flux is radiative energy per second per unit area. Radiative flux per unit

frequency is expressed below by equation 3.2.

$$= \int \cos \qquad (3.2)$$

Theta is defined in figure 6 below as the angle between the unit vector n, which is

perpendicular to the unit area dA, and the arbitrary direction L.

15
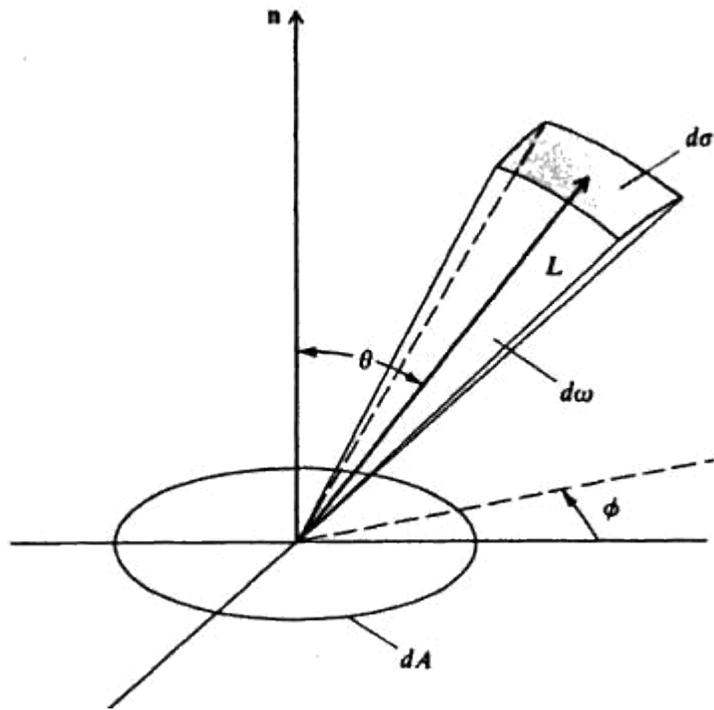
Figure 6. Illustrating the concept of radiative flux [4].

Radiative intensity is therefore expressed as = (, Ø). Anderson [4] defines the solid angle in terms of the area divided by the distance vector L^2.

$$= \quad \underline{\quad} = \frac{(\quad)(\quad Ø)}{\underline{\qquad\qquad}} \qquad\qquad (3.3)$$

Using this relationship, equation 3.2 can be expressed by the equation below.

$$= \int\int_{0\ 0}^{2} (.\emptyset)\emptyset \qquad\qquad (3.4)$$

16

When equation 3.4 is integrated over the entire electromagnetic spectrum, total

radiative flux can be expressed by equation 3.5.

$$q = \int_0^\infty \int_0^2 \int_0^\infty I(\nu, \omega) \, d\omega \qquad (3.5)$$

Equation 3.5 gives the total radiative flux for a body. It should be noted that radiative

intensity in equation 3.5 is also a function of the body's emissivity. To simplify the analysis

of total radiative flux, only blackbody radiative intensity will be considered. The radiative

intensity of a blackbody at frequency ν can be expressed by Planck's law below.

$$I = \frac{2h\nu^3}{c^2 \left(e^{h\nu/kT} - 1\right)} \qquad (3.6)$$

In equation 3.6 above, c is the speed of light, h is Planck's constant, k is the Boltzmann

constant, and T is temperature.

Using the definitions and expressions stated above, the radiative-transfer

equation can now be presented. Figure 7 below shows an elemental volume located

within a radiating gas, where intensity is incident on the element in the s direction.
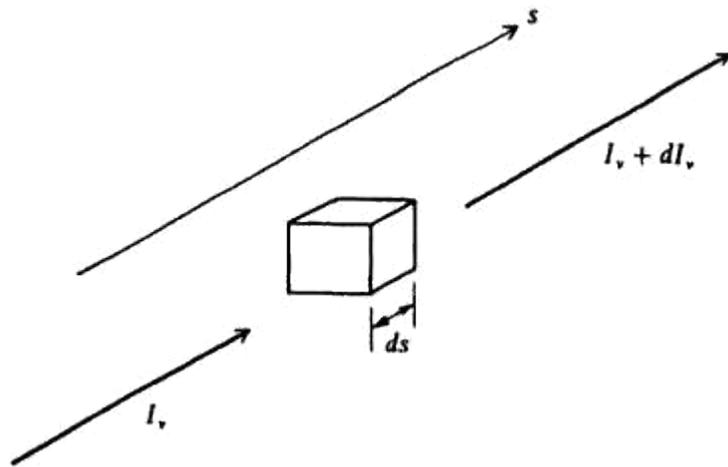
Figure 7. Illustrating the radiative-transfer equation [4].

Radiative intensity from within the element will be increased by emission and lowered by absorption. Energy emitted and energy absorbed can be expressed as equations 3.7 and 3.8 below.

$$=$$ (3.7)

$$=$$ (3.8)

The term is known as the emission coefficient, while the term is known as the absorption coefficient. The change in radiative intensity can then be expressed as equation 3.9 below.

$$=-$$ (3.9)

Equation 3.9 can also be expressed as equation 3.10 below.

$$\underline{\quad} = -\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad (3.10)$$

Anderson [4] defines the gradient of the radiative energy flux in terms of equation 3.10 as equation 3.11 below.

$$\nabla = \int - \quad\quad\quad \int \quad\quad\quad\quad (3.11)$$

The terms on the right side of equation 3.11 represent the local radiative emission and absorption. However, because the fluid transmits energy equally in all directions, equation 3.11 can be expressed as by equation 3.12 below.

$$\int = 4 \quad \int = 4 \quad\quad\quad\quad (3.12)$$

The term "J" in equation 3.12 is defined as the total radiative emission per second per unit volume. Equation 3.12 can now be rewritten as equation 3.13 below.

$$\nabla = 4 \; - \int \quad\quad\quad\quad (3.13)$$

Equation 3.13 is the radiative energy term usually included in the dynamic energy equations for hypersonic flows. If it were assumed that the radiating gas is a blackbody with intensity given by equation 3.6, intensity is constant and does not vary with distance "s" as shown in figure 7. Equation 3.10 can be expressed by equation 3.14 below.

19

$$= \qquad\qquad (3.14)$$

The term "J" in equation 3.13 is a function of frequency and can be integrated as shown below.

$$\int_0^\infty \qquad = \int \qquad\qquad (3.15)$$

Using the information above, a solution to the radiative-transfer equation where radiative flux is calculated will now be pursued. First, assume a finite volume of radiating transparent gas as sketched below in figure 8.
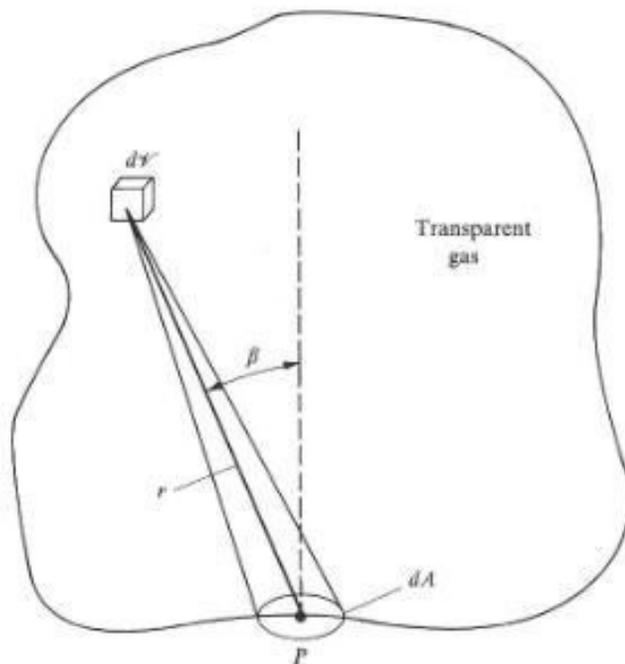
Figure 8. Illustrating the radiative flux at boundary point P [4].

20

The aim is to calculate the total radiative flux across the boundary at point P as a result of all the radiative gas inside the finite volume. Start by looking at the infinitesimal volume element dV, which is at a distance r from the infinitesimal area dA located at point P. The angle between the r vector and the unit vector perpendicular to dA is called Beta. The radiative energy that is transmitted from dV and crosses dA per second is expressed by equation 3.16.

$$\times \quad \frac{\cos}{^2} \tag{3.16}$$

The first term on the left of the multiplication sign represents the radiative energy per unit solid angle, which is multiplied by the solid angle intercepted by dA. After dividing by dA, the radiative flux at point P is found by integrating over the entire finite volume of the gas.

$$= \int \frac{\cos}{_2} = \frac{1}{_4} \int \frac{\cos}{_2} \tag{3.17}$$

The quantity E in equation 3.17 is defined as the total energy emitted by the gas in every direction per second per unit volume. The quantity E can be expressed by equation 3.18.

$$= 4 \tag{3.18}$$

Equation 3.17 can now be used to calculate the radiative flux at a given point on the surface of a hypervelocity reentry vehicle such as the stagnation point on Apollo Command Module. Before leaving this section on the theory behind radiative heating, an expression for calculating the radiative flux at a point along an infinite slab of transparent

radiating gas will be developed. The reason being is that an infinite slab has a very

simple geometry compared to blunt body shock-layers that generally have a

paraboloid shape. Figure 9 below shows an infinite slab of transparent radiating gas

with constant properties (density and temperature.)
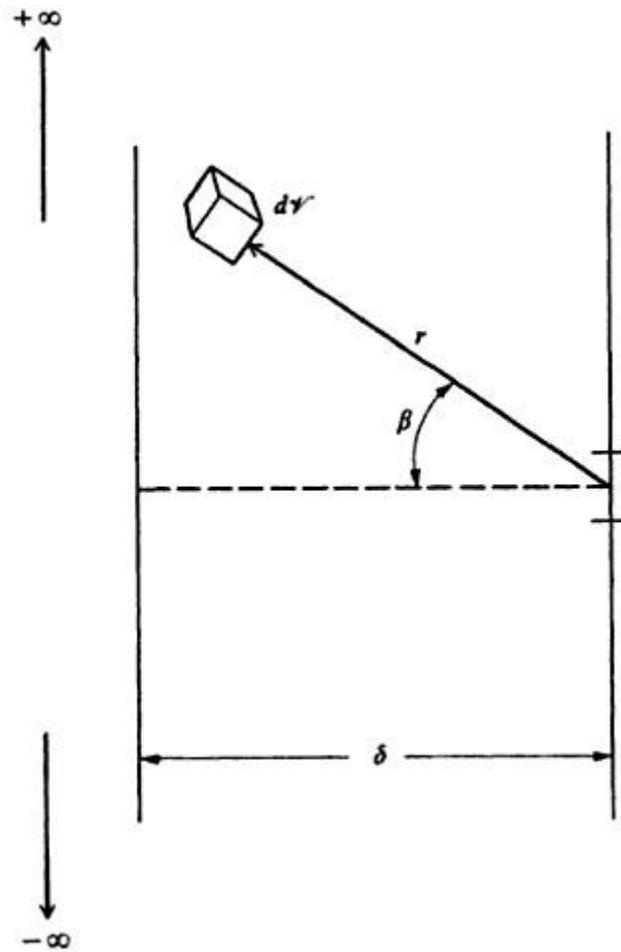
Figure 9. An infinite slab of transparent radiating gas [4].

Because the gas is assumed to have constant properties throughout the slab, E is constant throughout. The slab extends to plus and minus infinity along the vertical axis, as well as plus and minus infinity along the direction in and out of the page. The slab also has a given thickness . Looking at equation 3.17, dV can be expressed in terms of spherical coordinates by equation 3.19 below.

$$= {}^2 \sin \emptyset \qquad (3.19)$$

After substituting equation 3.19 into equation 3.17, the radiative flux at any point along the right boundary of the slab can be expressed by equation 3.20 below.

$$= \frac{}{4} \iiint_{000} \frac{2\,2\,\cos \overline{}}{} \frac{\cos \overline{}}{2} {}^2 \sin \emptyset \qquad (3.20)$$

Integrating equation 30 will yield equation 3.21 below.

$$(3.21)$$

$$\frac{}{4} \iint_{00} \sin \emptyset = \qquad \frac{\overline{}}{2} {}^{I_{\!}} \qquad \sin = {}^{/2} \qquad \frac{\overline{}}{2}$$

Equation 3.21 is a simple result for the radiative flux at any point on the surface/boundary of an infinite slab of transparent radiating gas. Approximating the radiative heating at the stagnation point on a hypervelocity blunt body can actually be done using the result of equation 3.21. Looking at figure 10 below, is the shock detachment distance and the stagnation point can be assumed to be a point along an infinite slab of transparent radiating gas.
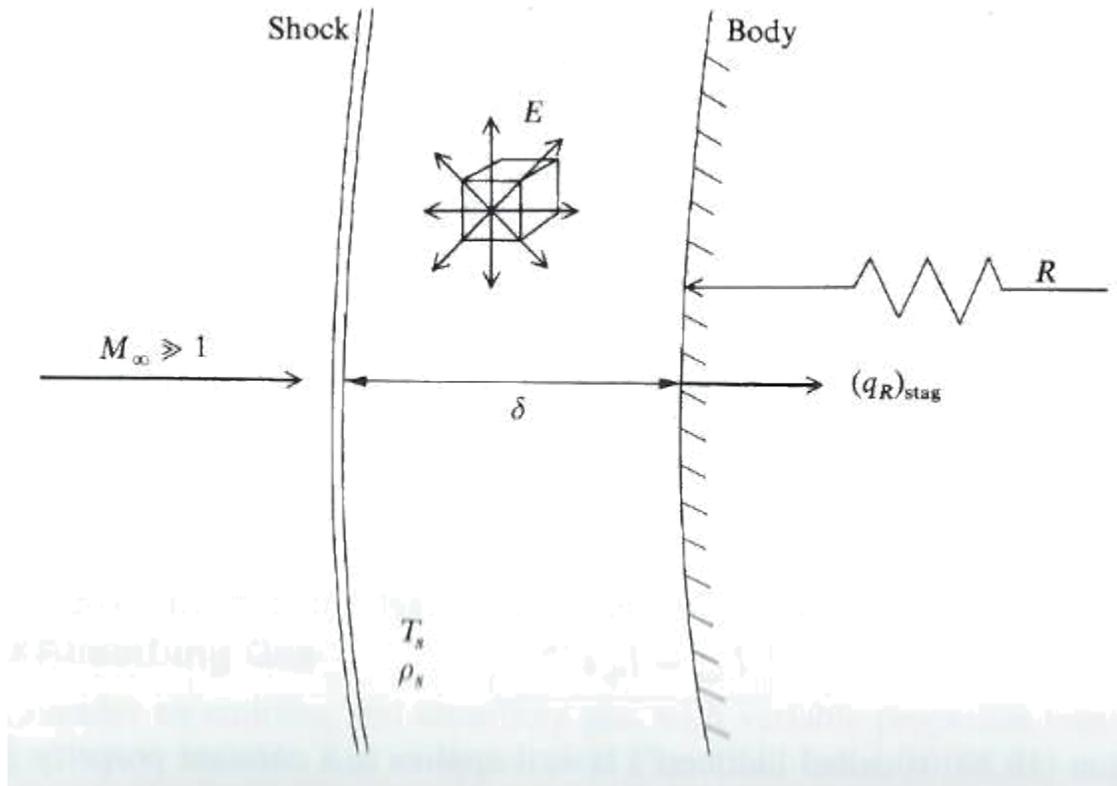
Figure 10. Approximating the stagnation radiative flux as that of an infinite slab [4].

Again, it is assumed that the gas properties behind the bow-shock are constant. That means E is constant since $=$ (,). is the gas temperature immediately behind the shockwave, and is the gas density immediately behind the shockwave. The shock detachment distance is approximated by equation 3.22 below.

$$= \underline{\qquad}$$

(3.22)

Using equations 3.21 and 3.22, the stagnation point radiative flux can be approximated by equation 3.23 below.

24

$$(\quad) \qquad = \_ \left(\quad \underline{\overset{\infty}{}} \quad \right) \qquad\qquad (3.23)$$

For a transparent radiating shock layer, equation 3.23 says that the radiative flux at the

stagnation point of a reentry vehicle is directly proportional to the nose radius R. By

reducing the nose radius of a vehicle, radiative heating will in turn be reduced.

Interestingly enough, this result is in direct contradiction to the result mentioned above

for convective heating. Convective heating on a reentry vehicle is minimized by

employing a blunt body shape, hence a large R.

Chapter 4.   Maslen's Method

S. H. Maslen was the chief of Aero-physics research at Martin Company, and an associate fellow member of AIAA. In October of 1963, he published a paper titled "Inciscid Hypersonic Flow Past Smooth Symmetric Bodies." It is an inverse method used for computing the entire flow-field in the shock-layer of a sufficiently smooth axisymmetric or planar body moving at hypersonic speeds. The method also involves the solution in a von Mises plane/transformation that uses the stream function and distance along the shockwave as its independent variables.

Shock-layers that flow over hypersonic bodies are generally thin. As the free-stream Mach number approaches infinity and ratio of specific heat capacity approaches 1, it can be shown that the shock angle approaches the turn angle . This means that the shock-layer becomes infinitely thin and dense. When such a limit is assumed, the shock shape, vehicle shape, and the streamline shapes in between are all the same shape. This assumption forms the basis of thin shock-layer theory. Maslen's Method has a simple and straight forward procedure and is frequently used in approximate analysis of inviscid hypersonic shock-layers. Its results are valid for a flow-field over slender as well as blunt-bodies. To begin the analysis of Maslen's Method, first consider the shock-layer geometry assumed by Maslen in figure 11 below.
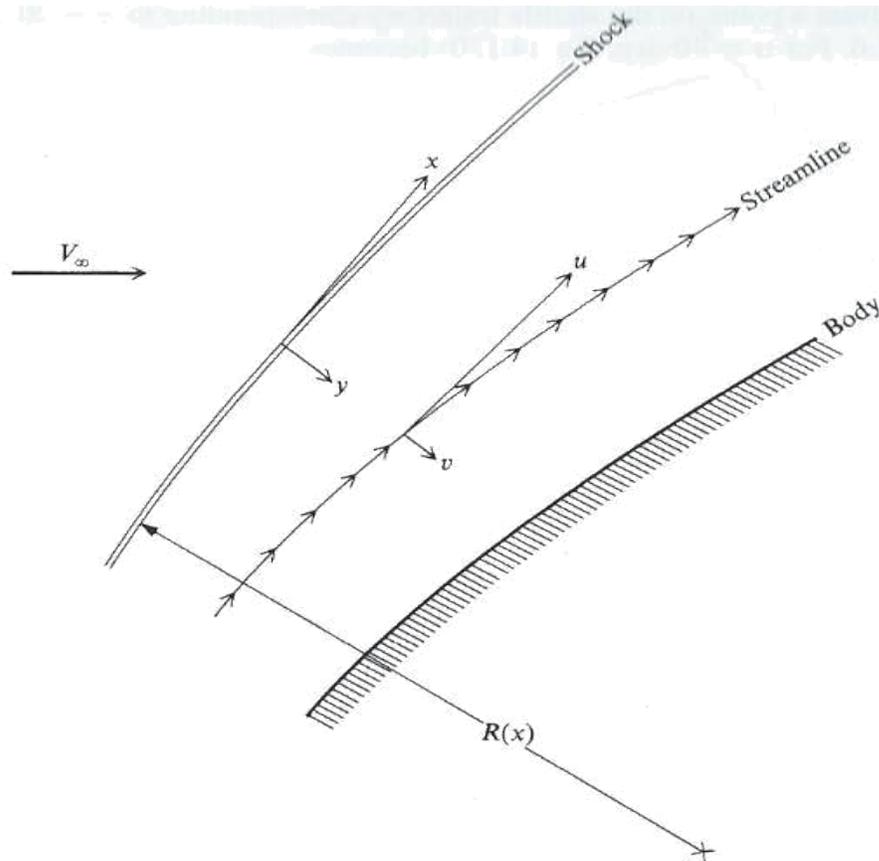
Figure 11. Thin shock-layer geometry used by Maslen [4].

In this system, x is the distance parallel to the shockwave divided by a characteristic

length, y is the distance perpendicular to the shockwave divided by a characteristic

length, u is the component of velocity in the x direction, and v is the component of

velocity in the y direction. While Maslen's Method applies to both 2-dimensional and

axisymmetric flow, an example of 2-D flow will be discussed.

Start by assuming the streamlines in figure 11 are basically parallel to the

shockwave, due to the shock-layer being very thin. The momentum equation

perpendicular to a streamline in a stream-based coordinate system is expressed by equation 4.1 below.

$$\frac{}{} {}^{2} = \frac{}{} \tag{4.1}$$

In equation 4.1, is defined as the shockwave radius of curvature. Next, a stream function is defined by the equation below.

$$= \overline{} \tag{4.2}$$

In order to replace y with ψ in equation 4.1, a von Mises transformation is performed, thereby making the independent variables (x,ψ). After performing the transformation, equation 4.1 is expressed by equation 4.3.

$$\overline{} = \overline{} \tag{4.3}$$

Next, it is assumed that all of the streamlines in the shock-layer are parallel to the shockwave, hence ≈ . Using this assumption, equation 4.3 is rewritten below.

$$\overline{} = \overline{} \tag{4.4}$$

Equation 4.4 can now be integrated from a point in the shock-layer where the value of the stream function is simply ψ, to a point just behind the shock-wave where = . This results in the following expression, which is the driving force behind Maslen's Method.

$$(,) = ( ) + \frac{()}{()} [-] \tag{4.5}$$

28

To produce a flow-field solution, the following algorithm is followed. First, assume a particular shockwave shape shown below in figure 12.
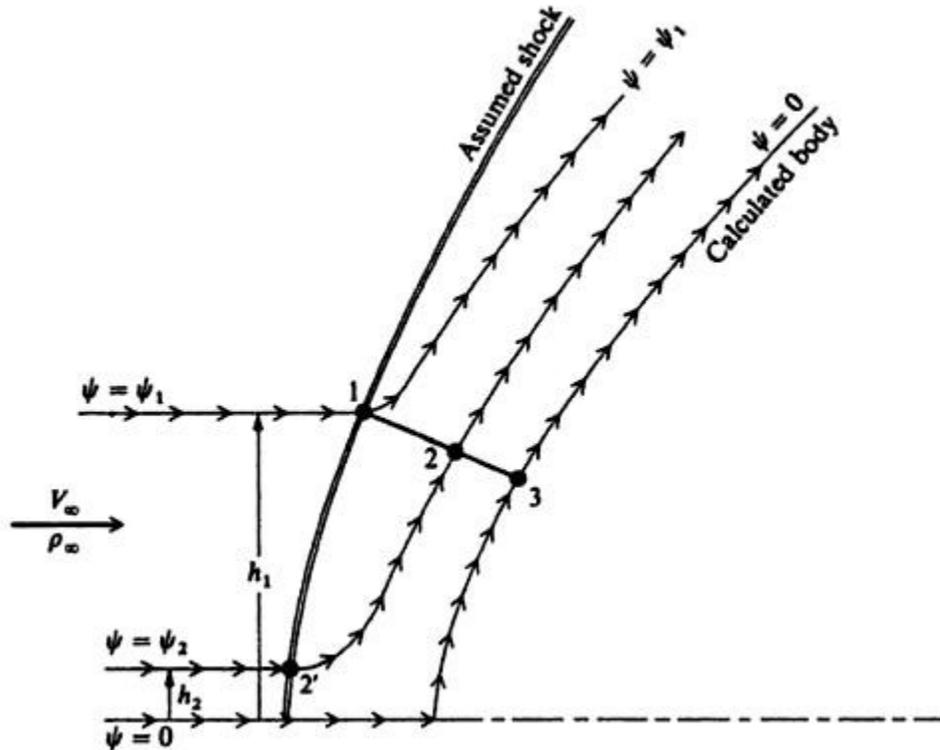
Figure 12. Maslen's assumed shock-shape and the calculated body [4].

The next step is to use the oblique shock relations since the flow quantities at point 1 (shown in figure 12) are known. The value of the stream function at point 1 is calculated from equation 4.6 below.

$$\psi_1 = \rho_\infty u_\infty h_1 \qquad (4.6)$$

Proceed by choosing a value of $_2$ such that $0 < _2 < _1$. This identifies point 2 along the y-axis in figure 12. The next step is to calculate the pressure at point 2 using equation 4.7.

$$= + \frac{1}{} [\quad -] \tag{4.7}$$

The next step is to calculate the entropy at point 2, knowing that the streamline at point 2, came through point 2', where $_2' = _2$ Point 2' on the shockwave is located by equation 4.8 below.

$$h_2 = \frac{2}{\circ\circ}$$ (4.8)

The variable h in equation 4.8 refers to the height above the axis of symmetry in figure 12, not to be confused with the symbol for enthalpy. Since the assumption that the flow along any streamline is isentropic, the statement of $2 = 2'$ can be made. Using thermodynamic equations of state, calculate the enthalpy at point 2 using equation 4.9 below.

$$2 \qquad \frac{2 + (- 1)^2}{}$$

$$- [1 + \frac{2}{+1}(^2 \quad - 1)]$$ (4.9)

is the component of the free stream Mach number normal to the shockwave. Using equations 4.10 through 4.13 below, the enthalpy at point to can be calculated using equation 4.14.

30

$$(4.10)$$

$$\frac{(\quad - \quad)+ (\quad \underline{2}\quad)}{}$$ (4.11)
(4.12)
(4.13)
(4.14)

$$2 = \infty$$

$$h \qquad = \quad + \quad \underline{\quad}\,_2$$ (4.15)

$$= \underline{\quad}\,_2$$

$$h_2 = \,_2(\quad + \quad)$$

Next, calculate the velocity at point 2 using equation 4.15 below.

The term h_0 refers to the total enthalpy. Because adiabatic flow is assumed, total enthalpy is constant. Since it is also assumed that $\gg$, the above expression can be rewritten as equations 4.16 and 4.17 below.

$$(4.16)$$

$$= (2(h \qquad \qquad) \qquad (4.17)$$

All of the flow quantities have now been calculated at point 2. Repeat each of the preceding steps for each point along the y-axis shown in figure 11 and figure 12 between the shock and the body. The body's surface is defined by $\psi=0$. The next step is to

31

calculate the physical coordinate y that represents each value of ψ, hence each point along the y-axis. This is done by integrating the stream function defined above.

$$= \int \overline{\qquad}$$

(4.18)

This integration can be performed since density and the x component of velocity have been previously calculated as a function of ψ. To locate the surface of the body, simply carry out the integration of equation 4.18 with the limits shown below.

$$= \int_0 \overline{\qquad}$$

(4.19)

The entire procedure is repeated for any number of points along a specified shockwave. This will not only generate the shock-layer flow-field, but also the body that supports the shock. Again, this algorithm is for a two-dimensional flow, which can easily be modified for an axisymmetric flow. Details of the procedure for axisymmetric flow are listed in [10].

An example of Maslen's Method will now be carried out. Refer to appendix A to see the Matlab code that carries out the whole process. First a shock shape of

mathematical form $= (2 \;)^{1/2}$ is assumed, where z is the axis of symmetry, and r is the axis normal to z. The radius of curvature is found by equation 4.20 below.

$$= \frac{[1 + (\qquad)]^{2\;3/2}}{\left|\;^2\;\right|}$$

(4.20)

By taking the first and second derivatives or r with respect to z, the shockwave radius of

curvature can be expressed as a function of z. The above mentioned Matlab program

carries out Maslen's Method to produce the supporting body of the prescribed shock

shape as shown in figure 13 below.
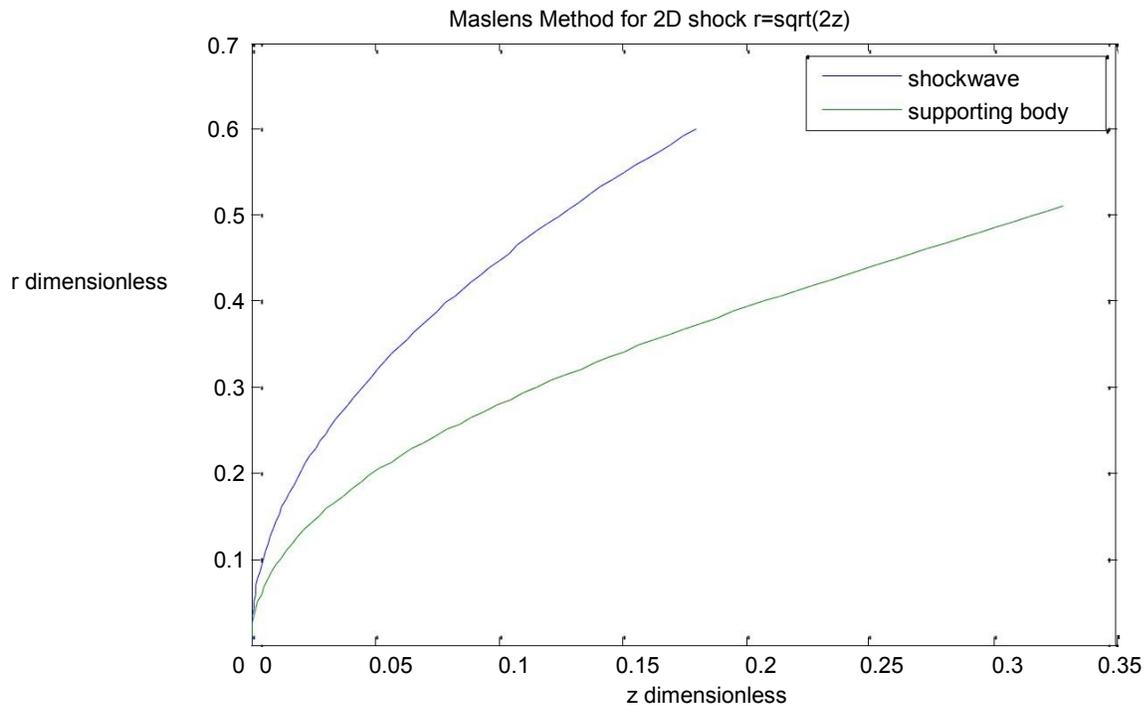
Figure 13. An example of Maslen's Method.

While the results shown in figure 13 are in agreement with the results found in [10], a

problem clearly exists in the stagnation region. The supporting blunt body appears to

touch the shock wave at the stagnation point. According to Maslen [10], there are several

difficulties with the above mentioned method in the stagnation region. First, the von

Mises transformation does not behave well on the axis of symmetry where both x and ψ are zero. Second, the curvature of the streamlines are opposite to that of the shockwave in this region so that the pressure should rise, rather than fall according to equation 4.5 when integrating from the shockwave to the surface of the body. Third, when the axis of symmetry is approached, then $\partial^2/\partial^2 \to \infty$. Maslen took a different approach than the above mentioned method for finding the supportive body and flow-field in the stagnation region. Details of this analysis can be found in [10]. Maslen's result for finding the supporting body, for axisymmetric hypersonic flow is given as equation 4.21 below.

$$(, 0) = \frac{-1}{(4)^2} \cdot \frac{1}{\{1 + ((-1)(+5)/4)\}} \tag{4.21}$$

In equation 4.21, $( )$ has already been found by equation 4.20, where z is a function of

X. Recall that x is the distance along the shockwave. Results for the prescribed shockwave $= (2)^{1/2}$ will now be presented below in figure 14.
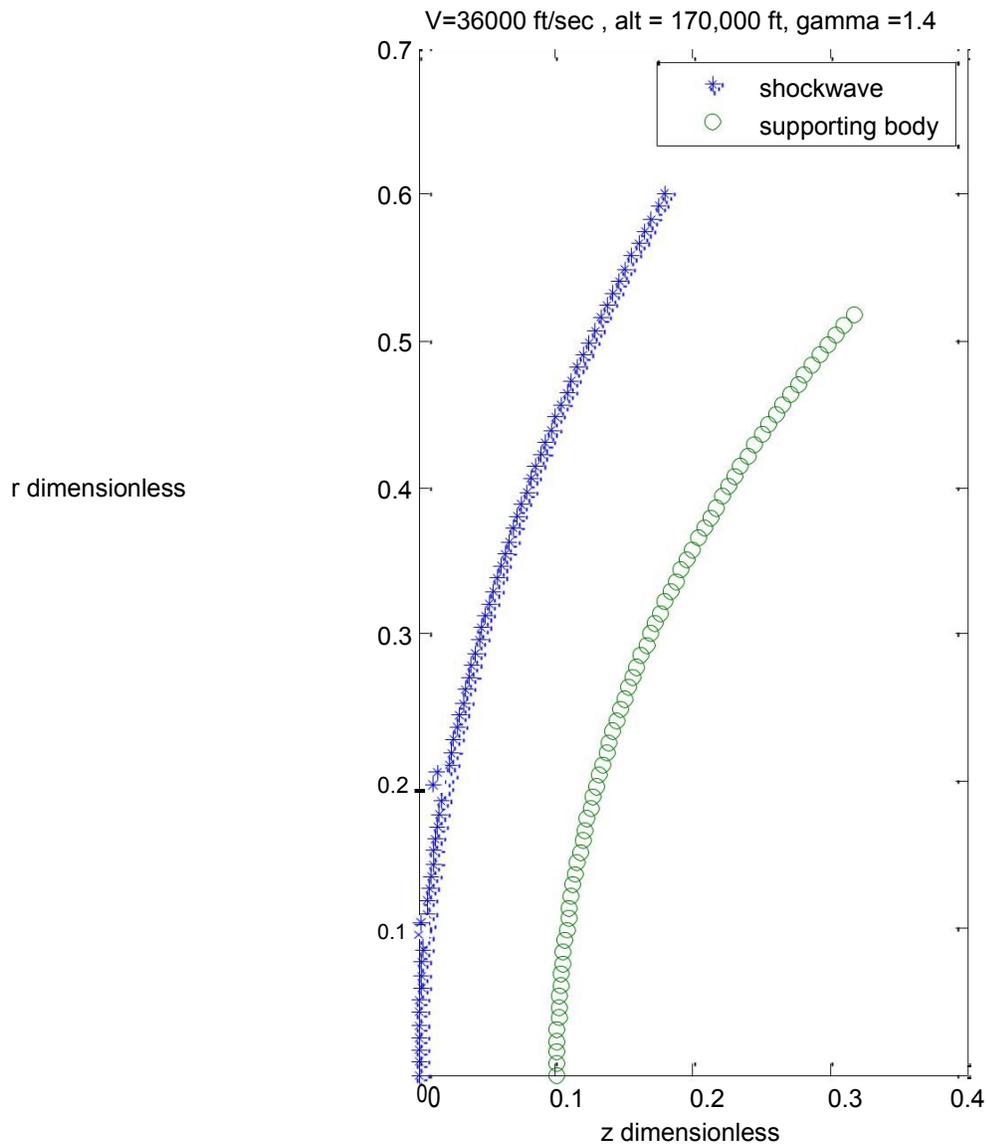
Figure 14. Maslen's Method for the stagnation region of a blunt body.

A Matlab program (see Appendix A) that performs Maslen's inverse method for a

supporting blunt body given an initial shock-shape is in place, which can be used for

estimating radiative flux at the stagnation point. However, before proceeding to calculating radiative heating, it must be pointed out that Maslen's approach is heavily depended upon the ratio of specific heat for air, or gamma. At speeds seen by the Apollo spacecraft on reentry, gamma is a variable, and the gas behind the bow shock is no longer calorically perfect, but chemically reactive. A detour must first be taken into the realm of inviscid high temperature equilibrium flows, and statistical thermodynamics. The aim being to establish a procedure that will accurately predict the flow conditions (pressure, temperature, density, gamma, etc…) immediately behind the bow-shock, where the flow is assumed to be in equilibrium. Recall from the previous discussion that radiative energy immediately behind a shockwave is a function of both air density and air temperature.

Chapter 5.   High-Temperature Equilibrium Air

There are five governing equations for an inviscid and adiabatic flow-field [4].

$$\text{Continuity:} \qquad \underline{\quad} + \underline{\frac{(\quad)}{\quad}} + \underline{\frac{(\quad)}{\quad}} + \underline{\frac{(\quad)}{\quad}} = 0 \qquad\qquad (5.1)$$

$$\text{x-momentum:} \qquad \underline{\quad} + \underline{\quad} + \underline{\quad} + \underline{\quad} = - \underline{\quad} \qquad\qquad (5.2)$$

$$\text{y-momentum:} \qquad \underline{\quad} + \underline{\quad} + \underline{\quad} + \underline{\quad} = - \underline{\quad} \qquad\qquad (5.3)$$

$$\text{z-momentum:} \qquad \underline{\quad} + \underline{\quad} + \underline{\quad} + \underline{\quad} = - \underline{\quad} \qquad\qquad (5.4)$$

$$\text{energy:} \qquad \underline{\quad} + \underline{\quad} + \underline{\quad} + \underline{\quad} = 0 \qquad\qquad (5.5)$$

The continuity equation was derived as a statement of global mass conservation. This equation is valid for both calorically perfect and chemically reacting flows. The x, y, and z momentum equations are a statement of Newton's second law of motion. These three equations are also independent of consequences that stem from chemical reactions. The energy equation was derived from the first and second laws of thermodynamics which hold for any type of gas, hence chemically reacting and non-reacting. Equation 5.5 can be rewritten with total enthalpy as the dependent variable.

$$\underline{\quad h_0 \quad} = \underline{\quad} \qquad\qquad (5.6)$$

The term D in equation 5.6 refers to the substantial derivative. Total enthalpy can be expressed as equation 5.7 below.

$$h = h + \frac{2}{2} \qquad\qquad (5.7)$$

High temperature effects do not change the form of equations 5.1-5.7. This is due to the

fact that energy changes caused by chemical reactions in a high speed flow are accounted

for by absolute enthalpy. Enthalpy or h contains the effective zero-point energies,

specifically the heats of formation in it. Therefore energy changes due to chemical

reactions are accounted for when h is taken as absolute enthalpy. The expression for

absolute enthalpy is shown below.

$$h = \sum (\; - \;) \qquad\qquad + \sum (\Delta \;)^{\circ} \qquad\qquad (5.8)$$

In equation 5.8, H is the enthalpy of the mixture (a gas comprised of numerous chemical species) per mole of mixture. $_0$ is the zero-point energy of species I, or simply the energy of the species at absolute zero. $_0$ is constant for a given chemical species. is the mole-mass ratio. The term $(\Delta \;)^{\circ}$ is called the heat of formation at absolute zero.

The first term on the right side of equation 5.8 is called sensible enthalpy. This can be

calculated using the results from statistical mechanics. The second term on the right side

of equation 5.8 is called the effective zero-point energy. This term is obtained from

measured and tabulated data. Therefore when h is taken as the form of equation 5.8

above, no heat addition terms are required for equation 5.6 in order to account for

chemical reactions!

A rather convenient way of inputting high-temperature equilibrium air properties into a flow-field calculation is through polynomial correlations of the tabulated and calculated data. The calculated and tabulated data of Tannehill and Mugge [8] are widely used today. Using the polynomial correlations of Srinivasan, Tannehill, and Weilmuenster [14], proceeding to calculate equilibrium normal and oblique shockwave flows can now be performed. Anderson [5] states the governing equations for a flow across a normal shockwave as follows.

$$\text{Continuity:} \qquad \rho_1 u_1 = \rho_2 u_2 \tag{5.9}$$

$$\text{Momentum:} \qquad p_1 + \rho_1 u_1^2 = p_2 + \rho_2 u_2^2 \tag{5.10}$$

$$\text{Energy:} \qquad h_1 + \frac{u_1^2}{2} = h_2 + \frac{u_2^2}{2} \tag{5.11}$$

Equations 5.9-5.11 hold for both reacting and non-reacting gases. The equilibrium thermodynamic properties for high-temperature gases will be taken from the polynomial correlations mentioned above. These high temperature properties can be expressed in the following "equations of state."

$$\rho_2 = \rho(p_2, h_2) \tag{5.12}$$

$$T_2 = T(p_2, h_2) \tag{5.13}$$

Anderson [4] points out that equations 5.9-5.13 will yield a series of closed-form algebraic relations for the pressure and temperature ratios, as well as downstream Mach number as a function of upstream Mach number. Unfortunately these simple relationships cannot be obtained for a gas that is chemically reacting. When chemical

reactions are occurring in a flow, equations 5.9-5.13 must be solved numerically. To begin such a numerical solution, equations 5.9-5.11 will be rearranged as follows.

$$= \, _2 \qquad \frac{1 \, 1}{_2}$$

After substituting equation 5.14 into equation 5.10 and solving

for expressed below.

$$_2 = \, _1 + \, _1 \, ^2 \, (1 - \, ^1) \qquad \qquad _1 \qquad \qquad \overline{} \\ _2$$

After substituting equation 5.14 into equation 5.11 and solving for

obtained below.

$$_2 \qquad \qquad \qquad 2 \qquad \qquad 2 \\ _= _+ \frac{}{_1} \, [1 - (\qquad \frac{}{_2} \qquad)] \\ _2 \qquad \qquad \, _{\tilde{}}$$

2, equation 5.15 is

^(5.14) h_2, equation 5.16 is

$$\tag{5.15}$$

$$\tag{5.16}$$

Equations 5.15 and 5.16 are expressed in terms of only the unknown density ratio

($_1/_2$). Since all the free-stream flow conditions shown below in figure 15 are known, a numerical iteration scheme can be set up for solving the flow condition downstream of the shock.

40

Known   To be calculated

u_1 u_2

p_1 p_2

h_1 h_2

ρ_1 ρ_2

T_1 T_2
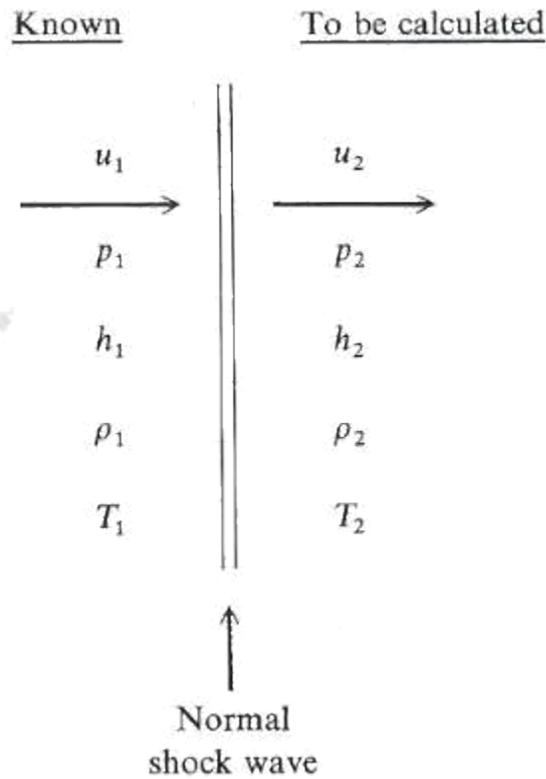
Normal
shock wave

Figure 15. Inviscid High Temperature Flow across a normal shockwave [4].

First, assume a value for (ρ_1/ρ_2). Anderson [4] recommends a value of 0.1 as a good starting point. Second, calculate $_2$ h_2 from equations 5.15 and 5.16. Third, calculate $_2$ from equation 5.12. Fourth, calculate a new (ρ_1/ρ_2) from the calculated value of $_2$ in the previous step. Finally, using the new (ρ_1/ρ_2), calculate a new

$_2$ h_2 using equations 5.15 and 5.16. Repeat these five steps until convergence occurs. According to Anderson [4], convergence usually takes less than five iterations. The sixth step is to take the correct values of $_2$, $_2$, h_2 and obtain $_2$ from equation 5.13. A correct value of $_2$ can then be obtained from equation 5.14. There is

41

fundamental difference between a chemically reactive gas and a calorically perfect gas. For a calorically perfect gas, the ratios h_2/h_1, ρ_2/ρ_1, and ρ_2/ρ_1 are functions of only the free-stream Mach number M_1. In a gas where chemical reactions are occurring, these

ratios are a function of M_1, M_1, M_1. It's rather interesting to consider that the free stream Mach number does not have a profound impact on the flow properties behind a normal shockwave in a high-temperature gas. This is in direct contrast to a calorically perfect gas. The most profound difference between chemically reacting air in equilibrium and a calorically perfect gas is the temperature ratio. This is due to the fact that in a calorically perfect gas, the kinetic energy of the flow ahead of the shockwave is primarily converted into translational and rotational molecular energy behind the shockwave. A thermally perfect gas will convert the kinetic energy of the flow ahead of the shockwave into all the various molecular modes of energy, as well as changing the zero-point energy of the chemical reaction products.

A procedure is now in place for calculating the equilibrium high-temperature flow conditions across a given shockwave that will then be used in the process for calculating the supporting body and shock-layer flow-field through Maslen's Method, as well as calculating the radiative energy that is emitted in the shock-layer. Matlab code used for carrying out the numerical procedure that calculates the equilibrium high-temperature gas properties is presented in appendix (C and D). As stated earlier, Maslen's Method for calculating the supportive body in the stagnation region of an axisymmetric flow, is highly dependent on the value of gamma, or the ratio of specific heat capacities of air. Values of gamma are obtained from [13].

42

Chapter 6.   Estimating Radiative Heat Flux

6.1   Transparent Shock Layer

To start the analysis for estimating the radiative heat flux at the stagnation point of a reentry vehicle similar to the Apollo Command Module, some initial assumptions will be made. The vehicle is traveling at 36,000 ft/sec at an altitude of 170,000 ft. The flow around the vehicle is considered to be a high-temperature chemically reactive gas in equilibrium. The shock-layer is comprised of a transparent radiating gas. It is also assumed that E, or total energy emitted, is constant along each radius "r" line drawn from the stagnation point on the surface of the body to the shockwave. A bow-shock shape of

$= (2)^{1/2}$ is assumed. This gives the shock radius of curvature as a function of z (and

hence of x) expressed as equation 6.1 below.

$$\frac{[1 + \frac{1}{2}]^{3/2}}{|-\frac{1}{2} \frac{1}{3/2}|}$$

Proceeding to carry out Maslen's Method for the stagnation region of an axisymmetric hypersonic flow using the Matlab program found in appendix (A) is carried out below. A ratio of specific heat capacity for air (gamma) behind a normal shockwave using the above flight conditions is obtained from [13]. Figure 16 below compares the supporting bodies for both the calorically perfect gas and the chemically reactive gas. The supporting body for the chemically reactive gas is then compared against the nose radius of the Apollo Command Module for accuracy.

V=36000 ft/sec , Alt = 170,000 ft , Nose radius = 4.69 m

Legend:
- shock wave
- calorically perfect
- chemically reacting
- ApolloCM

r meters

z meters
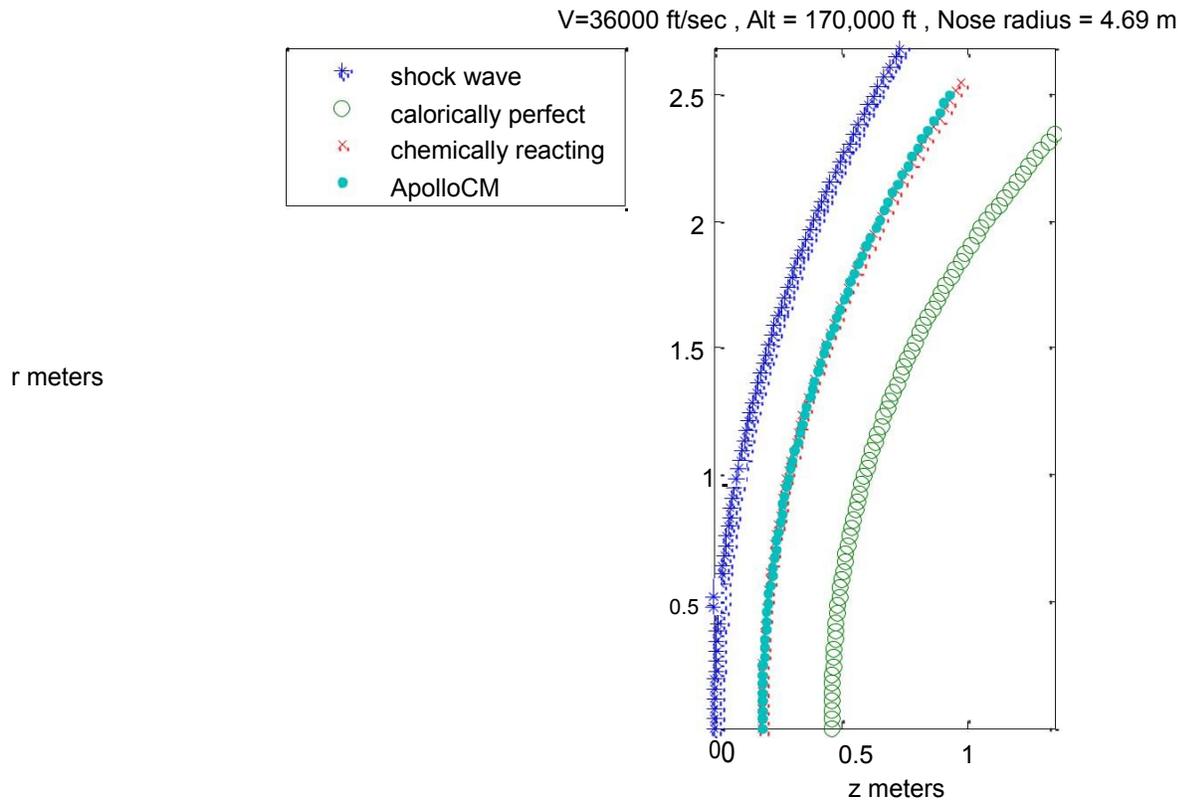
Figure 16. Maslen's Method for a calorically perfect gas and a chemically reactive gas.

Figure 16 shows a thinner shock-layer for a chemically reactive gas than for a calorically perfect gas. This is in agreement with theory presented by Anderson [4] shown below in figure 17. Shock-layers are generally thinner and denser for flows where the gas is chemically reactive.
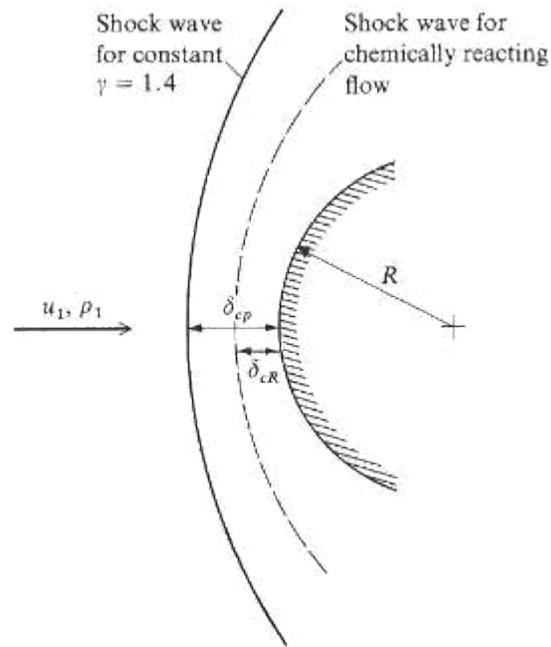
Figure 17. Comparison of shock detachment distances [4].

Later when calculating the radiative flux at the stagnation point, the distance "r" from the

stagnation point to each of the shockwave points will need to be known. These distances

are calculated and presented as a function of Beta angle in figure 18 below. Beta angle is

shown above in figure 9 as the angle between the axis of symmetry z, and the line

connecting the stagnation point to a given shockwave point.
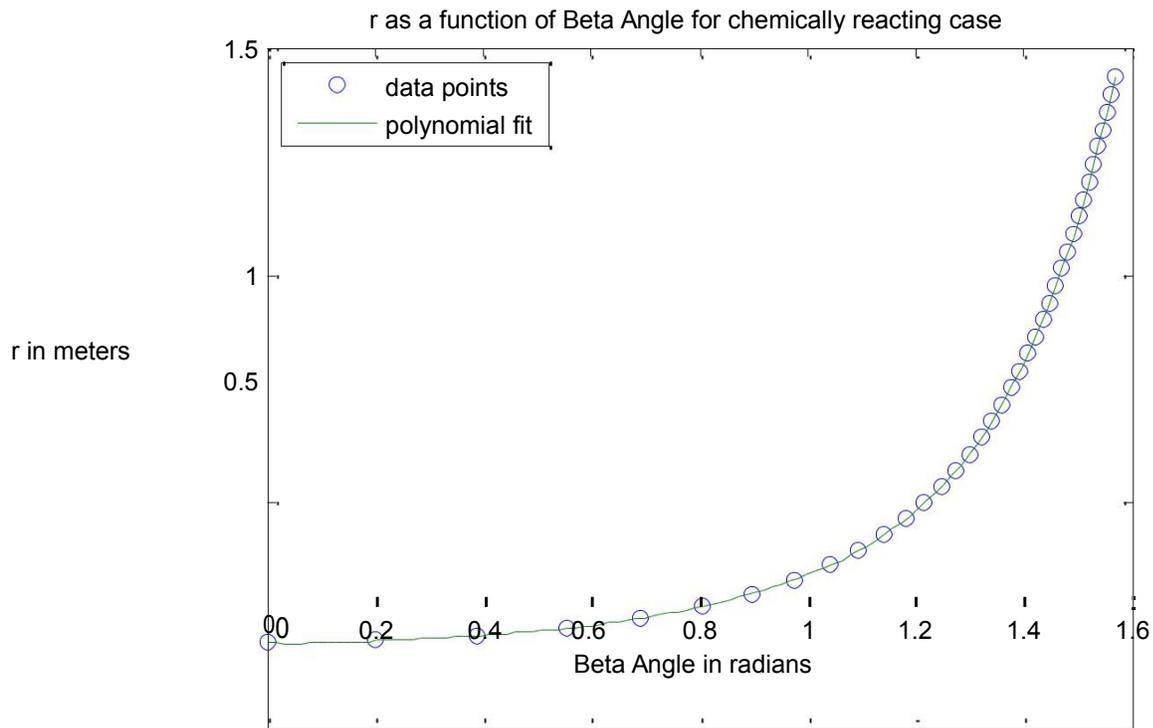
Figure 18. Stagnation-point/Shockwave distance as a function of Beta angle.

Next the flow conditions are calculated immediately behind the bow-shock using the techniques described above. The Matlab program used to perform these calculations is presented in appendix (C and D). The air temperature immediately behind the bow-shock is plotted as a function of Beta angle below in figure 19.
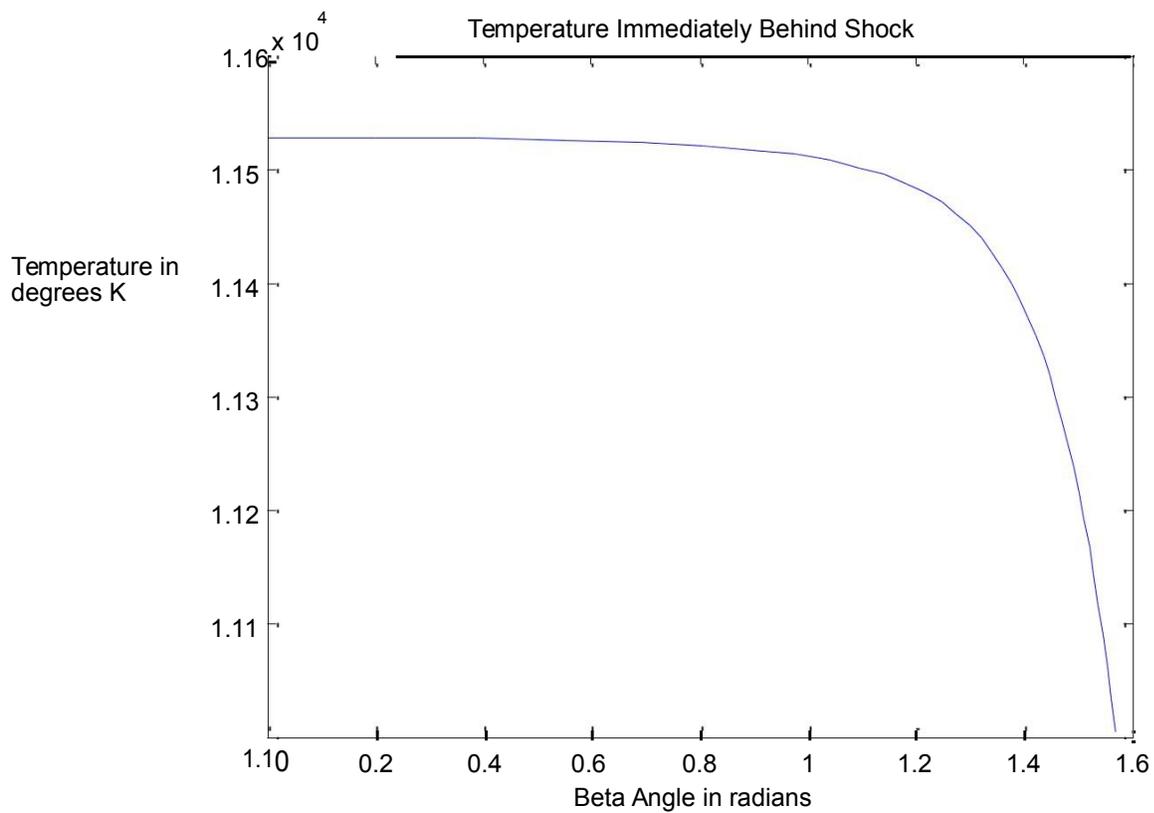
Figure 19. Air temperature immediately the bow-shock as a function of Beta angle.

Using the air temperature data calculated above, the radiative energy of air immediately behind the shockwave is calculated. Total radiative energy was defined above as =

4 , where J is the total radiative emission per second per unit volume. J is evaluated by equation 3.15 above. To evaluate this integral, the absorption coefficient of air as a function of radiative frequency must be obtained from data similar to what is shown below in figure 20.
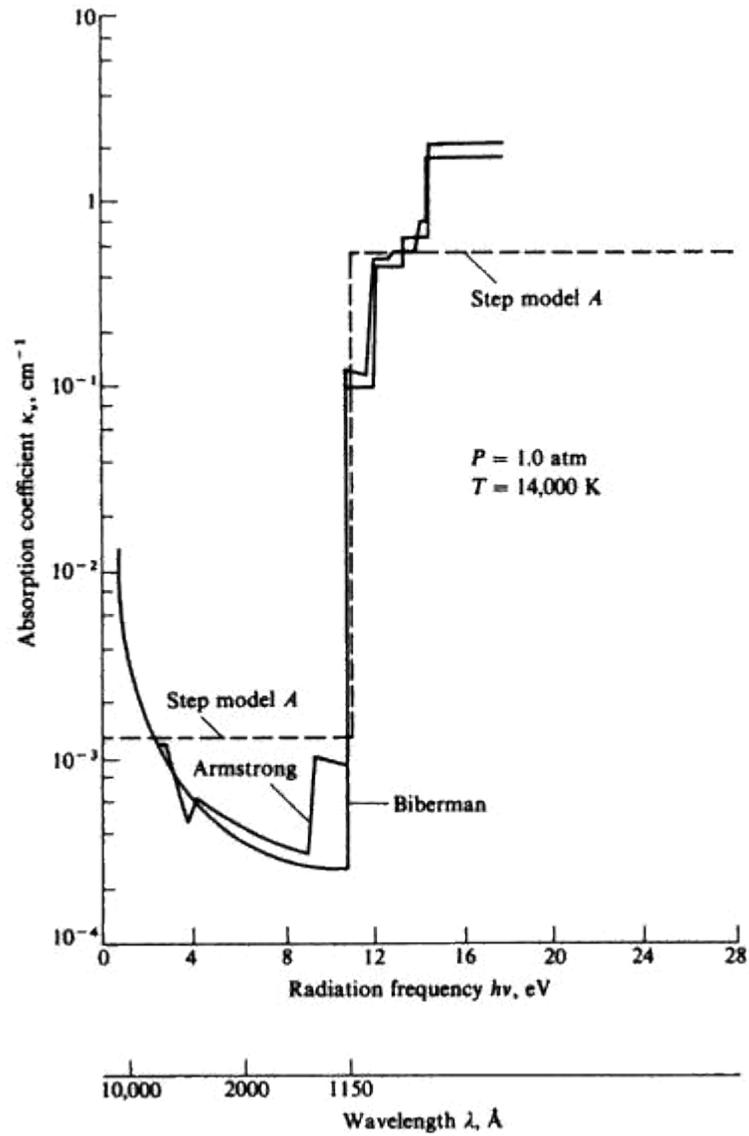
Figure 20. Absorption coefficient for air as a function of radiation frequency [4].

When evaluating the integral of equation 3.15, the limits of integration will be
determined by the frequency range of the data given. The Blackbody radiative intensity
is given as equation 3.6 above. Using the calculated temperatures as a function of Beta

angle in figure 19 above, equation 3.15 is evaluated to calculate J as a function of Beta angle. Since = 4 , total radiative energy is calculated as a function of Beta angle. This is shown below in figure 21.
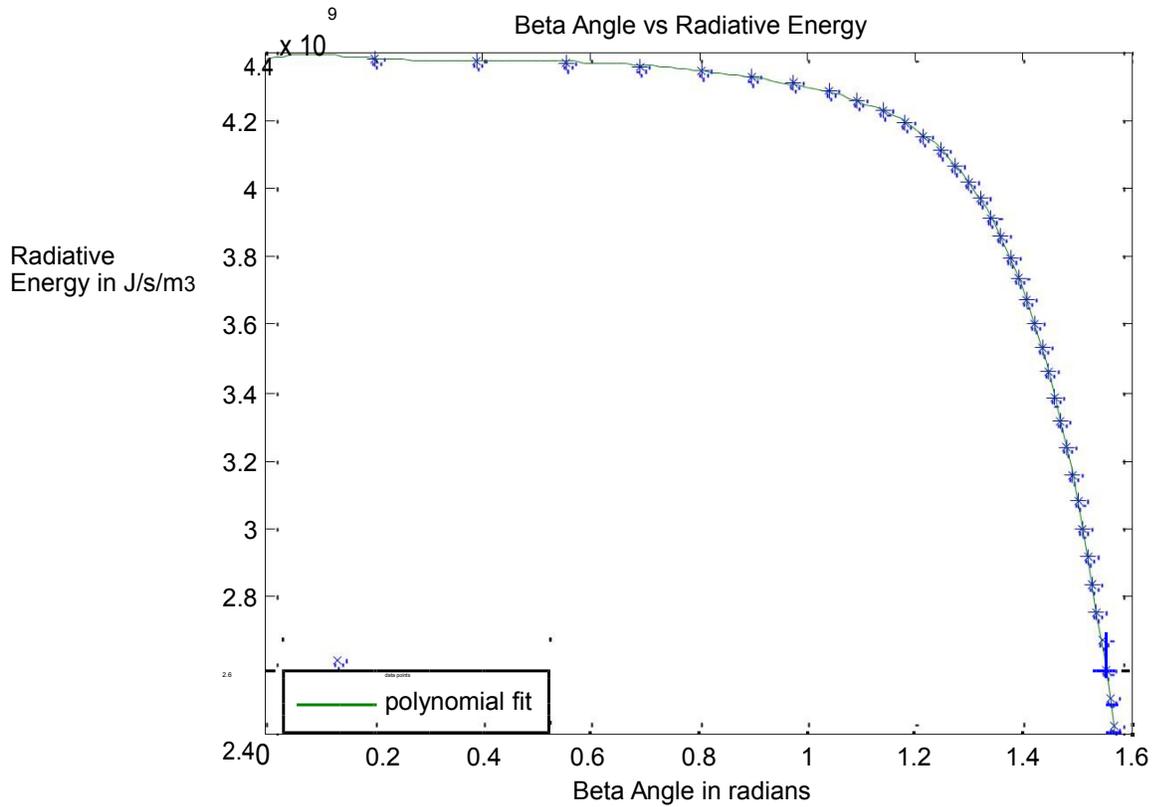
Figure 21. E as a function of Beta Angle.

Radiative flux at the stagnation point is now ready to be calculated using equation 3.20 above. However, through the analysis that has been conducted, E is no longer a constant as it was for the case of an infinite slab of radiating transparent gas. Likewise r is not a constant. The Matlab program has expressed both E and r as 6th order polynomial

49

functions of Beta angle. Equation 3.20 can now be expressed in the form shown below as equation 6.2.

$$\dot{q} = \frac{1}{4} \int_0^{2\pi} \int_0^{\pi} \int_0^{\frac{\pi}{2}} \cos() \sin() \, () \; \varnothing \tag{6.2}$$

Performing the first integration, equation 6.2 becomes a double integral shown below.

$$\dot{q} = \frac{1}{4} \int_0^{2\pi} \int_0^{\frac{\pi}{2}} \cos() \sin() \,()() \; \varnothing \tag{6.3}$$

Recall that dφ refers to integrating 2 radians about the axis of symmetry as shown in figure 9 above. Equation 6.3 can then be further integrated and expressed as the single integral shown below.

$$\dot{q} = \frac{1}{2} \int_0^{\frac{\pi}{2}} \cos() \sin() \,()() \tag{6.4}$$

Recall that radiation from the shock-layer is incident on the stagnation point only for Beta angles between 0 and $\pi/2$. Multiplying the 6th order polynomial functions for E and r and performing the final integration in Matlab, the radiative flux at the stagnation point is calculated below.

$$\dot{q} = 3.30 \times 10^{8} \; \frac{}{2}$$

50

As a sanity check, this value should be compared with the value given by equation 3.21 for the infinite slab of radiative transparent gas. Using the calculated value of E for the normal point on the bow-shock of 1.28×10^9 / ^{3}, and the value of "r" at a zero Beta angle, the calculated radiative flux is shown below.

$$ = \overline{}_{2} = 4.02 \times 10^8 \; \overline{}_{2} $$

It makes sense that the radiative flux for a point on an infinite slab of transparent radiative gas will be significantly more than the radiative flux at the stagnation point of the reentry vehicle. This is because the stagnation point of the vehicle is not receiving radiative emissions from an infinite slab. It is receiving radiation from a 3D "paraboloid" of radiating transparent gas centered at the stagnation point of the vehicle. Another benchmark for comparison is the data from Anderson [4] shown in figure 23 below. Running the entire procedure for a vehicle nose radius of 15 feet, a free-stream velocity of 36,000 ft/sec, and an altitude of 200,000 feet gives a calculated stagnation-point radiative heat flux of 1230 / ^{2}. For the corresponding flight conditions, figure 23 gives an approximate heat flux of 300 / ^{2}. This makes sense that the value given by figure 23 is less than half the value given by the procedure outlined above. The radiative heating takes into account both shock-layer cooling and non-grey self-absorption. Both of these phenomenon would significantly reduce the radiative heat flux seen by the stagnation point. Shock-layer cooling can be factored into the procedure above by accounting for the radiation loss parameter [15]. Figure 22 below shows

51

radiative cooling as a function of the radiation loss parameter. The radiation loss parameter Γ is defined below as equation 6.5.

$$\Gamma = \frac{\quad}{\quad} \tag{6.5}$$
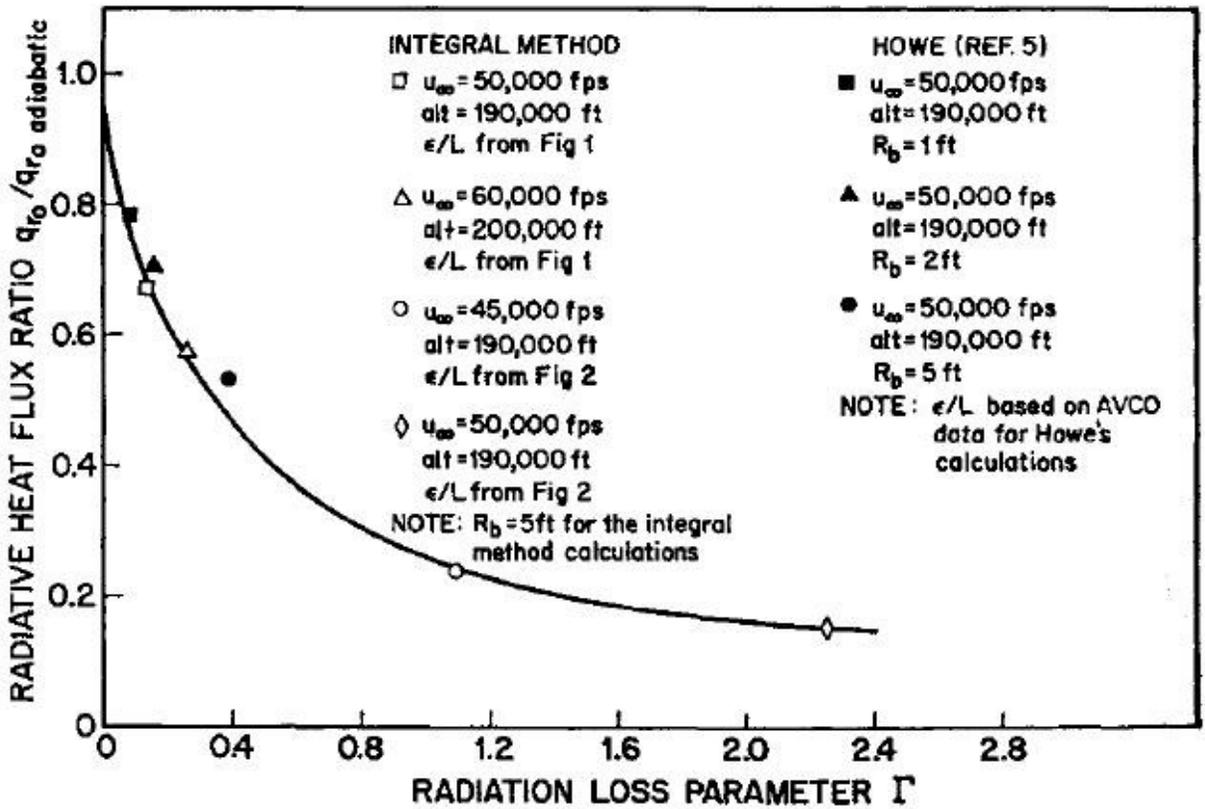
Figure 22. Radiative cooling as a function of the radiation loss parameter [15].

Applying the radiation loss parameter, a calculated stagnation-point radiative heat flux of

784 $/^2$ is obtained. This value is still expected to be significantly greater than the value given in figure 23 on account of non-grey self-absorption not being accounted

52

for. Emitting and absorbing shock-layer will now be explored in section 6.2 below in an attempt to account for non-grey self-absorption.
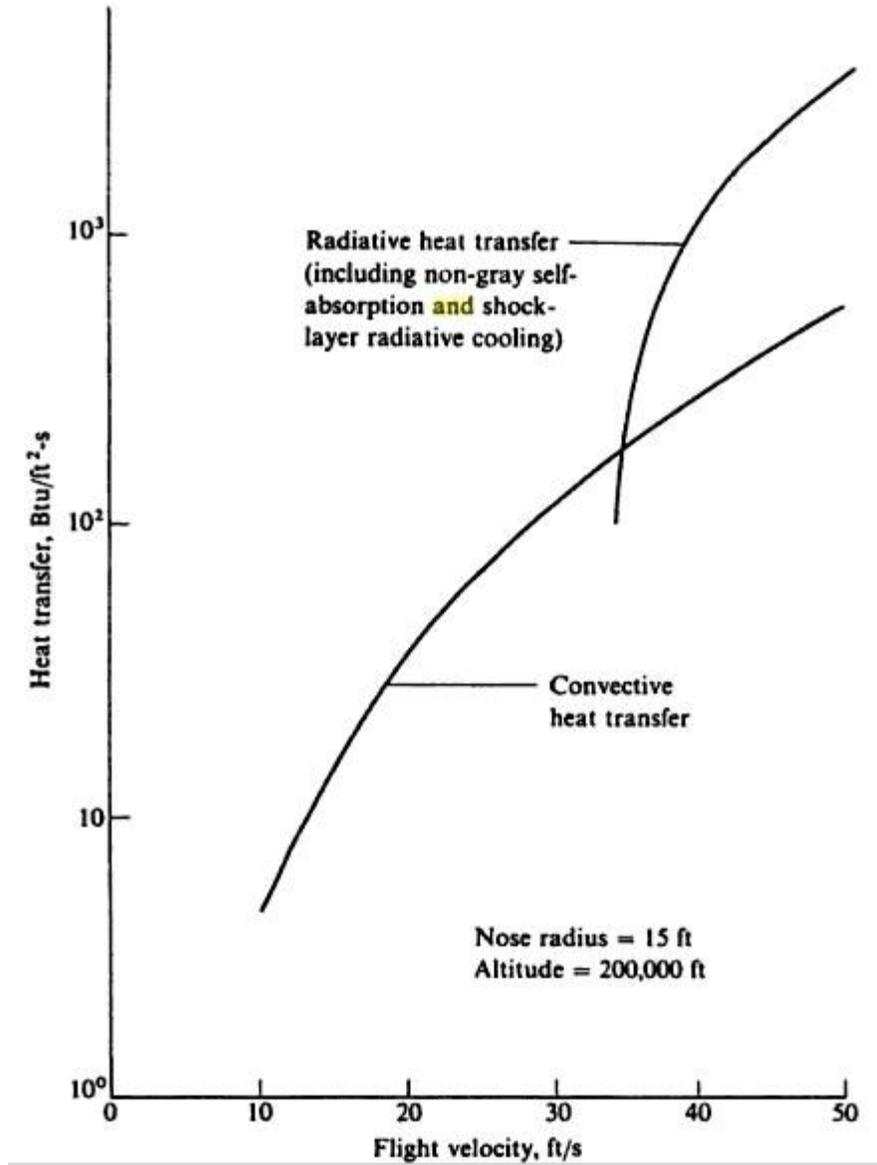
Figure 23. A comparison of radiative and convective stagnation point heat rates [4].

6.2   Emitting and Absorbing Shock Layer

To calculate the radiative heat flux at the stagnation-point of a reentry vehicle, the
assumptions made in reference [7] will be followed. The first assumption is that the gas
in the shock-layer will be in local thermodynamic and chemical equilibrium. Secondly,
only one-dimensional radiative energy transport will be considered. The third assumption
is that the surface of the vehicle is cold, non-emitting black surface. The fourth
assumption is that radiation emission and absorption upstream of the bow-shock will be
ignored. Using these assumptions, the radiative flux at the stagnation point can be
calculated using equation 6.6 below.

$$= 2 \iint ( ) \in_2 ( ) \tag{6.6}$$

In equation 6.5 above,   is given by equation 3.6, and t is the optical length to a given
point in the shock-layer. $\in$ ( ) is the exponential-integral of order n, shown below as equation 6.7.

$$1$$

$$\in ( ) = \int \tag{6.7}$$

The spectral optical thickness   of the shock-layer is calculated by equation 6.8 below.

$$= \int ( ) \tag{6.8}$$

In equation 6.8 above, is the shock-detachment distance given by Maslen's Method above. Unfortunately equation 6.6 cannot be evaluated as a closed-form solution, and must be evaluated numerically. To do this, the volume contained in the general region set by the integration limits in equation 6.6 must be calculated. This is done by summing the individual volumes of each general-hexahedron that lie under the function being integrated in equation 6.6. Reference [16] gives a straight forward procedure for calculating the volume of a general-hexahedron shown below in figure 24.
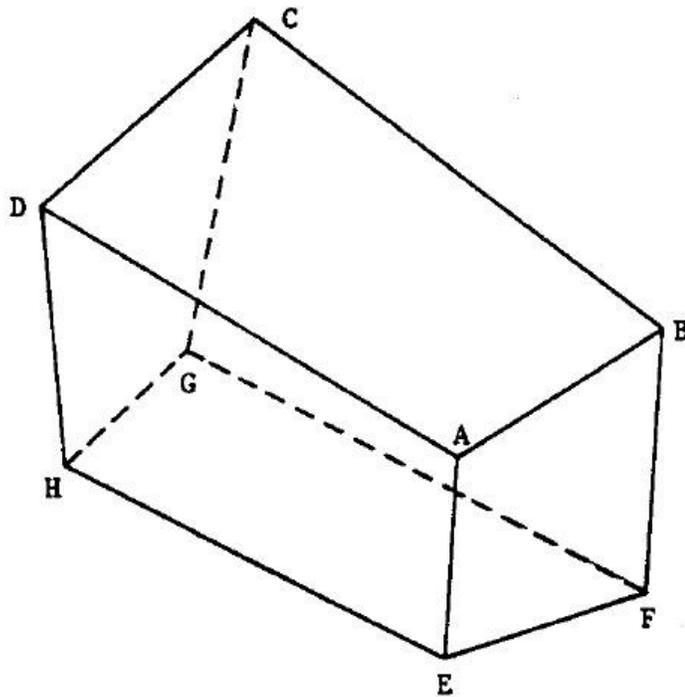
Figure 24. A general-hexahedron ABCDEFGH [16].

The volume of hexahedron shown in figure 24 is given by equation 6.9 below.

55

$$= (1/12) \cdot \{[\quad \times \quad] + [\quad \times \quad] + [\quad \times \quad]\} + (1/12)$$

$$\cdot \{[\quad \times \quad] + [\quad \times \quad]\} + (1/12) \cdot \{[\quad \times \quad] + [\quad \times \quad]\} + (1/12)$$

$$\cdot \{[\quad \times \quad] + [\quad \times \quad]\} \tag{6.9}$$

A procedure is now in place for evaluating equation 6.6 in the general region created by the optical length and frequency limits of integration. However, the radiation loss parameter calculated by equation 6.5 is applicable for a transparent shock-layer. Anderson [16] gives a procedure for obtaining Γ that can be applied to an emitting and

absorbing shock-layer. The Matlab code which implements all of the procedures listed above can be found in Appendices[ppp]. Using the above procedures which account for non-grey self-absorption and radiative shock-layer cooling, the radiative flux calculated for a vehicle with a nose radius of 15 feet, at an altitude of 200,000 feet, and a velocity of 36,000 ft/sec is 234 $/^2$. This is in much better agreement with the value of 300 $/^2$ obtained from figure 23.

Another benchmark for comparison are the results of Hoshizaki and Wilson [17]

shown below in table 1. The results of [17] were obtained from detailed analysis that

included coupling the gas-dynamic equations with radiative energy transport, and a

viscous boundary layer.

56

Table 1. Comparison of radiative flux

Flight Condition	Nose Radius (feet)	Analysis from section 6.2 (W/cm^2)	Results from [17] Hoshizaki and Wilson (W/cm^2)
$u_\infty = 50000$ ft/sec	2	1720	1700
Altitude = 200,000 ft.	4	2380	2180
	6	2730	2400

Chapter 7.   Conclusion

Data from Maslen's results are in excellent agreement with more precise methods of Van Dyke and Swigart. When incorporating high-temperature equilibrium effects into Maslen's method, the shock detachment distance at the stagnation point is reduced by

59.9% for the flight conditions shown in figure 16. Using the body given by Maslen's Method, a radiative heat flux at the stagnation point was calculated to be $3.30 \times$

10^8 $/$ ^{2} for the flight conditions shown in figure 16. This is approximately 17.9 % less than the radiative flux for a point in an infinite slab of transparent radiative gas with equal emissive energy and shock detachment distance. Results given in figure 23 provide a stagnation point radiative heat flux of approximately 300 $/$ ^{2} for a flight vehicle with a nose radius of 15 feet, at an altitude of 200,000 feet, and a flight velocity of 36,000 ft/sec. This is approximately 61.7% less than the calculated value for a transparent shock-layer that takes into account radiative cooling.

Results appear to agree more favorably with previously published data when an emitting and absorbing gas is modeled with shock-layer radiative cooling. A radiative flux of 234 $/$ ^{2} is calculated for the same conditions in figure 23 that give

approximately 300 $/$ ^{2}, which is approximately 22.0 % less. There is a 13.7 % difference between the results presented in table 1 above for a nose radius of 6 feet. A 9.2 % difference exists between results presented in table 1 for a nose radius of 4 feet.

Likewise, there is a 1.2 % difference between the results presented in table 1 for a nose radius of 2 feet.

58

It can be concluded that the techniques presented above give accurate predictions of stagnation-point radiative heating for purposes of engineering approximation.

References

[1] Young, Hugh D., Freedman, Roger A., University Physics, Addison-Wesley, 2007.

[2] J. D. Anderson, Introduction to Flight, 5th ed., New York: McGraw-Hill, 2005.

[3] Moss, J. N., and Simmonds, A. L., "Galileo Probe Forebody Flowfield Predictions During Jupiter Engry," AIAA Paper 82-0874, June 1982.

[4] J. D. Anderson, Hypersonic and High-Temperature Gas Dynamics, Reston: American Institute of Aeronautics and Astronautics, 2006.

[5] J. D. Anderson, Modern Compressible Flow: with Historical Perspective, 3rd ed., McGraw-Hill, New York, 2003.

[6] J. D. Anderson, Fundamentals of Aerodynamics, 5th ed., McGraw-Hill, New York, 2010.

[7] J. D. Anderson, "An Engineering Survey of Radiating Shock Layers," AIAA Journal, Vol. 7, No. 9, Sept. 1969, pp. 1665-1675.

[8] Tannehill, J. C., and Mugge, P. H., "Improved Curve Fits for the Thermodynamic Properties of Equilibrium Air Suitable for Numerical Computation Using Time-Dependent or Shock-Capturing Methods," NASA CR-2470, Oct. 1974.

[9] Anderson, John D., Jr., "Nongray Radiative Transfer Effects on the Radiating Stagnation Region Shock Layer and Stagnation Point Heat Transfer," U.S. Naval Ordinance Lab., NOLTR 67-104, White Oak, MD, July 1967.

[10] Maslen, S. H., "Inviscid Hypersonic Flow Past Smooth Symmetric Bodies," AIAA Journal, Vol. 2, No. 6, June 1964, pp. 1055-1061.

[11] Marrone, P. V., "Normal Shock Waves in Air: Equilibrium Composition and Flow Parameters for Velocities from 26,000 to 50,000 ft/s," Cornell Aeronautical Lab. (now CALSPAN), Rept. AG-1729-A-2, Buffalow, NY, March 1962.

[12] NASA CEA, URL: http://www.grc.nasa.gov/WWW/CEAWeb/ceaHome.htm

[13]   Hillje, Ernest R., "Entry Aerodynamics at Lunar Return Conditions Obtained from the Flight of Apollo 4 (AS-501)," NASA TN D-5399, 1969.

[14]   S. Srinivasan, J. C. Tennehill, K. J. Weilmuenster, "Simplified Curve Fits for the Thermodynamic Properties of Equilibrium Air" NASA-RP-1181, 1987.

[15]   K. H. Wilson, H. Hoshizaki, "Inviscid, Nonadiabatic Flow About Blunt Bodies", AIAA Journal, Vol. 3, No. 1, Jan. 1965, pp. 67-74.

[16]   D. E. Davies, D. Salmond, "Calculation of the Volume of a General Hexahedron for Flow Predictions", AIAA Journal, Vol. 23, No. 6, June. 1985, pp. 954-956.

[17]   K. H. Wilson, H. Hoshizaki, "Convective and Radiative Heat Transfer during Superorbital Entry", AIAA Journal, Vol. 5, No. 1, Jan. 1967, pp. 25-35.

[18]   Hillje, Ernest R., "Blunt Body Stagnation-Region Flow With Non-Grey Radiation Heat Transfer-A Singular Perturbation Solution" NASA TR R-295.

Appendix A

```matlab
%% Maslen's Method (for the downstream region)

%initial conditions
v_inf = 8000;           % m/sec
p_inf = 29.25;          % pascals
T_inf = 258.1;          % kelvin
rho_inf  = 3.9482e-04;  % kg/m^3
alt = 59000;            % m
gam = 1.4;              % ratio of specific heat air
R = 287;                % gas constant
cp = 1004.5;            % specific heat air const pressure
cv = 717.5;             % specific heat air const volume
M_inf  = v_inf/sqrt(gam*R*T_inf);   %free stream mach number


%assume a shock-wave shape

num =  72;                          %number of shock-wave points


x_shock  = linspace(0.4243,0,num).^2;   %square root
y_shock  = sqrt(2*x_shock);


  %calculate flow properties just behind the shock-wave @ all points

%stream function values for each point along the shock-wave
psi = rho_inf*v_inf*y_shock;
%radius of curvature at each point
R_s = (1+(1./(2*x_shock))).^1.5./abs(-1./(2*x_shock).^1.5);
R_s(1,num) = 1;
slope = 1./sqrt(2*x_shock);      %slope at each shock point
Beta = atand(slope);      %wave angle at each point
Theta = atand((2*cotd(Beta).*M_inf^2.*sind(Beta).^2-1)./(M_inf^2*...
    (gam+cosd(2*Beta))+2));   %the flow turn angle
%free stream mach number normal to the shock-wave at each point
M_inf_n = M_inf*sind(Beta);
%orthogonal mach number behind the shock-wave
M_n = sqrt((1+((gam-1)/2)*M_inf_n.^2)./(gam*M_inf_n.^2-((gam-1)/2)));
M = M_n./sind(Beta-Theta);      %mach number behind the shock-wave
%static pressure immediately behind the shock-wave
P = p_inf*(1+((2*gam*(M_inf_n.^2-1))/(gam+1)));
%static density immediately behind the shock-wave
Rho = rho_inf*(gam+1)*M_inf_n.^2./(2+(gam-1)*M_inf_n.^2);
%static temperature immediately beind the shock-wave
T=T_inf*(1+((2*gam*(M_inf_n.^21))/(gam+1))).*((gam+1)*M_inf_n.^2./(2+..
.(gam-1)*M_inf_n.^2)).^-1;
%magnigude of velocity immediately behind the shock-wave
V = sqrt(gam*R*T).*M;
u = V.*cosd(Beta-Theta);      % u velocity, or x component of velocity
```

```matlab
h0 = cp*T_inf+(v_inf^2/2);    %total enthalpy of the flow


%entropy change across the shock-wave at each point
Sshock_Sinf = cp*log((1+(2*gam*(M_inf_n(1,2:num).^2-1)/(gam+1))).*...
    ((2+(gam-1)*M_inf_n(1,2:num).^2)./((gam+1)*M_inf_n(1,2:num).^2)))-...
    R*log((1+(2*gam*(M_inf_n(1,2:num).^2-1)/(gam+1)))));




Pnew = zeros(1,num-1);
Tnew = zeros(1,num-1);
Rho_new = zeros(1,num-1);
hnew = zeros(1,num-1);
unew = zeros(1,num-1);


 for n = 1:1:num-1
    if n == 1
        Pnew(1,n) = P(1,n) + (u(1,n)/R_s(1,1))*(psi(1,n+1)-psi(1,n));
        Tnew(1,n) =
T_inf*exp((Sshock_Sinf(1,n)+R*log(Pnew(1,n)/p_inf))/cp);
        Rho_new(1,n) = Pnew(1,n)/(R*Tnew(1,n));
        hnew(1,n) = Tnew(1,n)*(cv+R);
        unew(1,n) = sqrt(2*(h0-hnew(1,n)));
    else
        Pnew(1,n) = Pnew(1,n-1) + (unew(1,n-1)/R_s(1,1))*...
            (psi(1,n+1)-psi(1,n));
        Tnew(1,n) =
T_inf*exp((Sshock_Sinf(1,n)+R*log(Pnew(1,n)/p_inf))/cp);
        Rho_new(1,n) = Pnew(1,n)/(R*Tnew(1,n));
        hnew(1,n) = Tnew(1,n)*(cv+R);
        unew(1,n) = sqrt(2*(h0-hnew(1,n)));
end
end

P_dist = [P(1,1) Pnew];
T_dist = [T(1,1) Tnew];
Rho_dist = [Rho(1,1) Rho _new];
u_dist = [u(1,1) unew];
M_dist = [V(1,1)/sqrt(gam*R*T(1,1)) unew./sqrt(gam*R*Tnew)];

funct_dist = 1./(Rho_dist.*u_dist);
poly = polyfit(psi,funct_dist,5);



y = zeros(1,num-1);

for j = 1:1:num-1
    psi_temp = linspace(psi(1,j+1),psi(1,1),100*j);
    funct_trapz = poly(1,1)*psi_temp.^5 + poly(1,2)*psi_temp.^4 +
poly(1,3)...
        *psi_temp.^3 + poly(1,4)*psi_temp.^2 + poly(1,5)*...
```

```matlab
        psi_temp + poly(1,6)*ones(1,100*j);
    y(1,j) = trapz(psi_temp,funct_trapz);
end


X_flow = x_shock(1,1) + y*cosd(90-Beta(1,1));
Y_flow = y_shock(1,1) - y*sind(90-Beta(1,1));

figure(1)
plot(x_shock,y_shock,'*',X_flow,Y_flow,'o')

% loop through for the whole body

X_flow = zeros(num-1,num-1);
Y_flow = zeros(num-1,num-1);


for pts = 1:1:num-1;
    x_temp = linspace(x_shock(1,pts)-1e-10,x_shock(1,num),num).^2;
    y_temp = sqrt(2*x_temp);
    psi_tempT = rho_inf*v_inf*y_temp;
    slope_temp = 1./sqrt(2*x_temp);
    Beta_temp = atand(slope_temp);
    Theta_temp = atand((2*cotd(Beta_temp).*M_inf^2.*sind(Beta_temp).^2-
1)./...
        (M_inf^2*(gam+cosd(2*Beta_temp))+2));
    Mn_inf_temp = M_inf*sind(Beta_temp);
    Mn_temp = sqrt((1+((gam-1)/2)*Mn_inf_temp.^2)./(gam*Mn_inf_temp.^2-
...
        ((gam-1)/2)));
    M_temp = Mn_temp./sind(Beta_temp-Theta_temp);
    Ptemp = p_inf*(1+((2*gam*(Mn_inf_temp.^2-1))/(gam+1)));
    Rho_temp = rho_inf*(gam+1)*Mn_inf_temp.^2./(2+(gam-
1)*Mn_inf_temp.^2);
    Ttemp = T_inf*(1+((2*gam*(Mn_inf_temp.^2-
1))/(gam+1))).*((gam+1)*...
        Mn_inf_temp.^2./(2+(gam-1)*Mn_inf_temp.^2)).^-1;
    Vtemp = sqrt(gam*R*Ttemp).*M_temp;
    utemp = Vtemp.*cosd(Beta_temp-Theta_temp);
    h0_temp = cp*T_inf+(v_inf^2/2);

    Sshock_Sinf_temp = cp*log((1+(2*gam*(Mn_inf_temp(1,2:num).^2-
1)/(gam+1))).*...
        ((2+(gam-
1)*Mn_inf_temp(1,2:num).^2)./((gam+1)*Mn_inf_temp(1,2:num).^2)))-...
        R*log((1+(2*gam*(Mn_inf_temp(1,2:num).^2-1)/(gam+1))));

    PnewT = zeros(1,num-1);
    TnewT = zeros(1,num-1);
    Rho_newT = zeros(1,num-1);
    hnewT = zeros(1,num-1);
    unewT = zeros(1,num-1);

    for nT = 1:1:num-1
```

64

```matlab
        if nT == 1
            PnewT(1,nT) = Ptemp(1,nT) + (utemp(1,nT)/R_s(1,pts))*...
                (psi_tempT(1,nT+1)-psi_tempT(1,nT)); TnewT(1,nT)
            = T_inf*exp((Sshock_Sinf_temp(1,nT)+...
                R*log(PnewT(1,nT)/p_inf))/cp);
            Rho_newT(1,nT) = PnewT(1,nT)/(R*TnewT(1,nT));
            hnewT(1,nT) = TnewT(1,nT)*(cv+R);
            unewT(1,nT) = sqrt(2*(h0_temp-hnewT(1,nT)));
        else
            PnewT(1,nT) = PnewT(1,nT-1) + (unewT(1,nT-
1)/R_s(1,pts))*...
                (psi_tempT(1,nT+1)-psi_tempT(1,nT));
            TnewT(1,nT) = T_inf*exp((Sshock_Sinf_temp(1,nT)+R*...
                log(PnewT(1,nT)/p_inf))/cp);
            Rho_newT(1,nT) = PnewT(1,nT)/(R*TnewT(1,nT));
            hnewT(1,nT) = TnewT(1,nT)*(cv+R);
            unewT(1,nT) = sqrt(2*(h0_temp-hnewT(1,nT)));
        end
    end

    P_dist_temp = [P(1,1) PnewT];
    T_dist_temp = [T(1,1) TnewT];
    Rho_dist_temp = [Rho(1,1) Rho_ newT];
    u_dist_temp = [u(1,1) unewT];

    funct_dist_temp = 1./(Rho_dist_temp.*u_dist_temp);
    poly_temp = polyfit(psi_tempT,funct_dist_temp,5);

    y_int = zeros(1,num -1);
    for jT = 1:1:num-1
        psi_tempT1 = linspace(psi_tempT(1,jT+1),psi_tempT(1,1),100*jT);
        funct_trapz = poly_temp(1,1)*psi_tempT1.^5 + poly_temp(1,2)*...
            psi_tempT1.^4 + poly_temp(1,3)*psi_tempT1.^3 +
poly_temp(1,4)*...
            psi_tempT1.^2 + poly_temp(1,5)*psi_tempT1 +
poly_temp(1,6)*...
            ones(1,100*jT);
        y_int(1,jT) = trapz(psi_tempT1,funct_trapz);
    end

    X_flow(pts,:) = x_shock(1,pts) + y_int*cosd(90-Beta(1,pts));
    Y_flow(pts,:) = y_shock(1,pts) - y_int*sind(90-Beta(1,pts));
end

figure(2)
plot(x_shock,y_shock,X_flow(:,num-1),Y_flow(:,num-1)),...
    title('Maslens Method for 2D shock r=sqrt(2z)'),...
    xlabel('z dimensionless'),ylabel('r dimensionless'),...
    legend('shockwave','supporting body')
```

Appendix B

```
% Maslens Method for the Stagnation Region
clear

%assume a shock-wave shape
CL = 4.573;%4.6939;%1.829;%%0.915;
num = 72;                              %number of shock-wave points

x_shock = linspace(0.4243,0,num).^(1/0.468);   %square root
y_shock = (2*x_shock).^0.468;
yap = linspace(2.5,0,num);


%calculate flow properties just behind the shock-wave @ all points
%initial conditions
v_inf = 10976;        % m/sec
p_inf = 20.1;         % pascals
T_inf = 243.9;        % kelvin
rho_inf = 2.867e-04; % kg/m^3
alt = 60976;          % m
gam = 1.4;            % ratio of specific heat air
R = 287;             % gas constant
cp = 1004.5;         % specific heat air const pressure
cv = 717.5;          % specific heat air const volume
M_inf = v_inf/sqrt(gam*R*T_inf);   %free stream mach number

%stream function values for each point along the shock-wave
psi = rho_inf*v_inf*y_shock;

%radius of curvature at each point
R_s = (1+(1./(2*x_shock))).^1.5./abs(-1./(2*x_shock).^1.5);
R_s(1,num) = 1;

slope = 1./sqrt(2*x_shock);    %slope at each shock point
Beta = atand(slope);      %wave angle at each point
Theta = atand((2*cotd(Beta).*M_inf^2.*sind(Beta).^2-1)./...
    (M_inf^2*(gam+cosd(2*Beta))+2));  %the flow turn angle

%free stream mach number normal to the shock-wave at each point
M_inf_n = M_inf*sind(Beta);

%orthogonal mach number behind the shock-wave
M_n = sqrt((1+((gam-1)/2)*M_inf_n.^2)./(gam*M_inf_n.^2-((gam-1)/2)));


M = M_n./sind(Beta-Theta);     %mach number behind the shock-wave
```

```matlab
%static pressure immediately behind the shock-wave
P = p_inf*(1+((2*gam*(M_inf_n.^2-1))/(gam+1)));
%static density immediately behind the shock-wave
Rho = rho_inf*(gam+1)*M_inf_n.^2./(2+(gam-1)*M_inf_n.^2);
%static temperature immediately beind the shock-wave
T = T_inf*(1+((2*gam*(M_inf_n.^2-
1))/(gam+1))).*((gam+1)*M_inf_n.^2./...
    (2+(gam-1)*M_inf_n.^2)).^-1;
%magnigude of velocity immediately behind the shock-wave
V = sqrt(gam*R*T).*M;
u = V.*cosd(Beta-Theta);     % u velocity, or x component of velocity
h0 = cp*T_inf+(v_inf^2/2);   %total enthalpy of the flow

% here we are comparing a chalorically perfect gas with gamma equal to
1.4
% and a chemically reactive gas with a gamma of 1.32 obtained from NASA
CEA
% program where we can clearly see a thiner shock layer for the
chemically
% reactive situation!

% ratio of specific heats given by NASA CEA program & Apollo 4 flight
data
gam1 = 1.12;

X_wall = zeros(1,num);
X_wall1 = zeros(1,num);
Y_wall = zeros(1,num);
Y_wall1 = zeros(1,num);
ywall = zeros(1,num);
ywall1 = zeros(1,num);

for w = 1:1:num
    ywall(1,w) = R_s(1,w)*((gam-1)/sqrt(4*gam))*(1/(1+sqrt((gam-
1)*(gam+5)...
        /(4*gam))));
    ywall1(1,w) = R_s(1,w)*((gam1-1)/sqrt(4*gam1))*(1/(1+sqrt((gam1-
1)*...
        (gam1+5)/(4*gam1))));
    X_wall(1,w) = x_shock(1,w) + ywall(1,w)*cosd(90-Beta(1,w));
    X_wall1(1,w) = x_shock(1,w) + ywall1(1,w)*cosd(90-Beta(1,w));
    Y_wall(1,w) = y_shock(1,w) - ywall(1,w)*sind(90-Beta(1,w));
    Y_wall1(1,w) = y_shock(1,w) - ywall1(1,w)*sind(90-Beta(1,w));
end

sthta = asind(yap/CL);
xbod = -CL*cosd(sthta)+CL+X_wall1(1,num)*CL;
x_body = xbod;

figure(1)
plot(x_shock*CL,y_shock*CL,'*',X_wall*CL,Y_wall*CL,'o',...
```

```matlab
        X_wall1*CL,Y_wall1*CL,'x',x_body,yap,'.'),...
        legend('shock wave','calorically perfect','chemically reacting',...
        'ApolloCM','location','NorthWestOutside'), ...
        xlabel('z meters'),ylabel('r meters'),...
        title('V=36000 ft/sec , Alt = 170,000 ft , Nose radius = 4.69 m')
axis equal tight

dist = sqrt((X_wall - x_shock).^2 + (Y_wall - y_shock).^2);
```

Appendix C

```matlab
%% Distance from stagnation point "r" to shockwave as function of Beta

% we only want the distances of points on the shockwave that can emit
% radiative energy in the normal component of the stagnation point
surface

beta = zeros(1,num);
r=zeros(1,num);
count=0;
count1=0;
for p = num:-1:1
    count=count+1;
    betatemp = atan((y_shock(1,p)*CL/(X_wall1(1,num)*CL-
x_shock(1,p)*CL)));
    if betatemp >= 0
        r(1,count) = sqrt((X_wall1(1,num)*CL-x_shock(1,p)*CL)^2+...
            (y_shock(1,p)*CL-Y_wall1(1,num)*CL)^2);
        beta(1,count) = betatemp; count1=count1+1;
    else
        r(1,count) = 0;
        beta(1,count) = 0;
    end
end
RR = r(1,1:count1);
beta_r = beta(1,1:count1);

%%%%%%%% establish r as a function of beta %%%%%%%%%%%%
order = 6;
coef = polyfit(beta_r,RR,order);    % r polynomial fit coefficients
x = linspace(0,beta_r(1,count1),100);
y =
coef(1,1)*x.^6+coef(1,2)*x.^5+coef(1,3)*x.^4+coef(1,4)*x.^3+coef(1,5)*.
..
    x.^2+coef(1,6)*x+coef(1,7); plot(beta_r,RR,'o',x,y),xlabel('Beta
Angle in radians'),ylabel('r in meters'),...
    title('r as a function of Beta Angle for chemically reacting
case'),...
    legend('data points','polynomial fit','location','NorthWest')
```

Appendix D

```
 %% Density, Pressure, Enthalpy, and u2 as a function of Beta angle

% free stream conditions at 170,000 ft and free stream velocity 36000
% ft/sec converted to metric units

cv = 717.5;
R = 287;
 rho1 = rho_inf;
p1  = p_inf;
T1  = T_inf;
u1  = v_inf;
h1  = T1*(cv+R);

utemp = u1*sind(Beta);
u = zeros(1,count1);
count2=0;
for e = num:-1:(num-count1+1)
    count2=count2+1;
    u(1,count2) = utemp(1,e);
end

u;

% calculate the pressure, density, u2, and enthalpy as a function of
% Beta

h = zeros(1,count1);
p_final = zeros(1,count1);
rho_ratio = zeros(1,count1);
u_s = zeros(1,count1);

for q = 1:1:count1
%assume density ratio  should be 0.065853785

rho_ratio _temp = 0.1;
err = 1;

while err > 5e-06
rho2 = rho1/rho_ratio_temp;  %assumed

p2 = p1 + rho1*u(1,q)^2*(1-rho_ratio_temp);
h2 = h1 + (u(1,q)^2/2)*(1-rho_ratio_temp^2);

X = log10(p2/1.01325e05);
Y = log10(rho2/1.2925);
```

```matlab
Z = X - Y;


if Y >= -7.0 && Y <= -4.5
    if Z <= 0.1
        c = [1.3986e0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0];
        sgn = 1;
    elseif Z > 0.1 && Z <= 0.85
        c = [2.53908e02 1.01491e02 -3.87199e02 -1.54304e02
7.28532e00...
            9. 86233e01 -8.04378e00 4.63763e01 -1.82577e-03
2.18994e01...
            -2.52423e02 -1.01445e02 3.87210e02 1.54298e02 -7.2773e00...
            -9.87576e01 8.04277e00 -4.63883e01 2.28399e-03 -
2.19438e01...
            -11e00 2.0e00 11e00 -2.0e00];
        sgn = -1;
    elseif Z > 0.85 && Z <= 1.30
        c = [-1.05745e01 -1.93693e00 3.07202e01 3.35578e00 -7.79965e-
02...
            -2.60637e01 6.68790e-02 -1.42391e00 -9.86882e-04
7.23223e00...
            -1.86342e01 2.41997e-02 3.20880e01 -7.46914e-01 3.75161e-
02...
            -1.69985e01 -4.10125e-02 5.39041e-01 5.74637e-04
2.56253e00...
            2.768567e02 2.152383e01 -2.164837e02 -1.394837e01];
        sgn = 1;
    elseif Z > 1.30 && Z <= 1.95
        c = [6.17584e-01 -2.40690e-01 1.95904e00 3.41644e-01 -1.01073e-
02...
            -1.68951e00 6.77631e-03 -1.10932e-01 -1.15922e-04 4.26058e-
01...
            -1.34222e01 -5.43713e-01 1.81528e01 3.95928e-01 -7.41105e-
03...
            -7.97425e00 1.67768e-03 -5.80593e-02 -3.32714e-06
1.12448e00...
            8.677803e01 -8.370349e00 -4.074084e01 7.407405e00];
        sgn = 1;
    else
        c = [-8.32595e00 -3.50219e-01 1.36455e01 3.59350e-01 -3.70109e-
03...
            -6.49007e00 3.30836e-03 -8.38594e-02 1.10018e-04
1.02443e00...
            -3.08441e01 -1.49510e00 3.00585e01 9.19650e-01 -3.60024e-
02...
            -9.33522e00 1.02522e-02 -1.35228e-01 -4.68760e-04 8.92634e-
01...
            8.800047e01 -1.679356e01 -3.333353e01 8.465574e00];
        sgn = 1;
    end
elseif Y > -4.5 && Y <= -0.5
    if Z <= 0.10
        c = [1.399e00 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0];
```

```matlab
        sgn = 1;
    elseif Z > 0.10 && Z <= 0.95
        c = [- 1.33083e02 -9.98707e00 3.94734e02
2.35810e01 1.43957e00...
            -3.84712e02 -1.43175e00 -1.36367e01 1.77068e-05
1.24325e02...
            1. 34486e02 9.99122e00 -3.94719e02 -2.35853e01 -
1.43799e00...
            3.84616e02 1.43039e00 1.36318e01 1.44367e-04 -1.24348e02...
            -2.141444e01 1.381584e00 2.039473e01 -1.315789e00];
        sgn = -1;
    elseif Z > 0.95 && Z <= 1.50
        c = [-7.36684e00 -1.13247e00 2.47879e01 1.99625e00 -4.91630e-
02...
            -2.3299e01 4.16673e-02 -8.59418e-01 -6.58149e-04
7.19016e00...
            -2.42647e00 5.57912e-01 -2.03055e00 -1.22031e00 3.74866e-
02...
            7. 75414e00 -3.39278e-02 6.08488e-01 5.21042e-04
- 3.68326e00...
            8.077385e01 -1.273807e01 -6.547623e01 1.190475e01];
        sgn = 1;
    elseif Z > 1.50 && Z <= 2.00
        c = [4.31520e-01 -2.83857e-01 2.27791e00 3.99159e-01 -1.29444e-
02...
            -1.84314e00 8.78724e-03 -1.28136e-01 -1.60583e-04 4.45362e-
01...
            -1.03883e01 -3.58718e-01 1.35068e01 1.87268e-01 -4.28184e-
03...
            -5.63894e00 -9.52016e-04 -1.45625e-03 -4.10506e-05
7.39915e-01...
            2.949221e02 1.36866e01 -1.559335e02 -3.78776e00];
        sgn = 1;
    else
        c = [-3.77766e00 -5.53738e-01 6.60834e00 4.87181e-01 -2.11045e-
02...
            -2.94754e00 9.67277e-03 -1.02365e-01 -2.1942e-04 4.3962e-
01...
            4.05813e01 3.25692e00 -4.79583e01 -2.53660e00 9.06436e-
02...
            1. 8904e01 -3.47578e-02 4.94114e-01 1.00077e-03
- 2.48554e00...
            5.34718e02 7.495657e01 -2.219822e02 -3.017229e01];
        sgn = 1;
    end
else
    if Z <= 0.10
        c = [1.4017e00 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0];
        sgn = 1;
    elseif Z > 0.10 && Z <= 1.05
        c = [-9.67488e01 2.05296e-01 2.69927e02 -1.92887e00 3.78392e-
01...
            -2.46711e02 -3.24965e-01 1.54416e00 -3.61036e-03
7.48760e01...
```

```matlab
                9.81502e01 -2.05448e-01 -2.69913e02 1.93052e00 -3.78527e-
01...
                2.46630e02 3.24832e-01 -1.54646e00 3.66182e-03 -
7.4898e01...
                -2.659865e01 1.564631e00 2.312926e01 -1.360543e00];
            sgn = -1;
        elseif Z > 1.05 && Z <= 1.60
            c = [-2.67593e-01 -1.87457e-01 5.07693e00 2.72286e-01 1.04541e-
02...
                -5.0852e00 -1.42211e-02 -7.81935e-02 6.38962e-04
1.58711e00...
                2.87969e00 3.9009e-01 -8.06179e00 -5.5125e-01 -1.01903e-
02...
                7. 29592e00 1.35906e-02 1.83861e-01 -8.97772e-04 -
2.15153e00...
                1.828573e02 -3.428596e01 -1.51786e02 2.976212e01];
            sgn = 1;
        else
            c = [9.21537e-01 -2.39670e-01 1.30714e00 3.4299e-01 -2.18847e-
02...
                -1.20916e00 1.36691e-02 -1.10206e-01 -4.90274e-04
3.087920e-01...
                -6.77089e00 -6.90476e-02 8.18168e00 -9.52708e-02 2.98487e-
02...
                -3.07662e00 -1.78706e-02 6.60408e-02 6.28419e-04 3.38590e-
01...
                1.5916669e02 3.976192e01 -7.966199e01 -1.66667e01];
            sgn = 1;
        end
end


gam_bar = c(1,1)+c(1,2)*Y+c(1,3)*Z+c(1,4)*Y*Z+c(1,5)*Y^2+c(1,6)*Z^2+...

  c(1,7)*Y^2*Z+c(1,8)*Y*Z^2+c(1,9)*Y^3+c(1,10)*Z^3+(c(1,11)+c(1,12)*Y+ ...
    c(1,13)*Z+c(1,14)*Y*Z+c(1,15)*Y^2+c(1,16)*Z^2+c(1,17)*Z*Y^2+...

c(1,18)*Y*Z^2+c(1,19)*Y^3+c(1,20)*Z^3)/(1+sgn*exp(c(1,21)+c(1,22)*Y+...
    c(1,23)*Z+c(1,24)*Y*Z));


h2_new = (p2/rho2)*(gam_bar/(gam_bar-1));


D = (h2-h2_new)*1e-10;
rho_ratio_temp = rho_ratio_temp+D;
err = abs(h2-h2_new);
end
p_final(1,q) = p2;
rho_ratio(1,q)=rho_ratio_temp;
h(1,q) = h2_new;
end

rho2_final = rho1./rho_ratio;
p_final;
```

```matlab
% calculate u2 velocity component normal to the shockwave
u2 = u1*rho1./rho2_final;
```

Appendix E

```matlab
% Calculate temperature immediately behind the shock-wave as a
function of
% Beta angle

Temp_final = zeros(1,count1);

for v = 1:count1

Y1 = log10(rho2_final(1,v)/1.2925);
X1 = log10(p_final(1,v)/1.01325e05);
Z1 = X1 - Y1;

if Y1 >= -7.0 && Y1 <= -4.5
    if Z1 > 0.25 && Z1 <= 0.95
        d = [1.23718e-01 1.08623e-02 2.24239e-01 -8.24608e-02 -
1.17615e-03...
            1.18397e00 -1.87566e-03 6.4852e-02 -1.19155e-04 -5.52634e-
01...
            0 0 0 0 0 0 0 0 0 0 0 0 0 0 0];
    elseif Z1 > 0.95 && Z1 <= 1.40
        d = [-8.12952e00 -8.28637e-01 2.26904e01 1.41132e00 -2.98633e-
02...
            -1.91806e01 2.70066e-02 -5.78875e-01 -2.28103e-04
5.62580e00...
            -3.99845e00 2.26369e-01 2.52870e00 -7.28448e-01 1.09769e-
02...
            2. 99238e00 -1.83819e-02 3.91440e-01 -1.51380e-04
- 2.04463e00...
            -3.887015e01 -2.908228e01 4.070557e01 2.682347e01];
    elseif Z1 > 1.40 && Z1 <= 1.95
        d = [-1.98573e01 -1.67225e00 3.76159e01 2.10964e00 -3.40174e-
02...
            -2.22215e01 2.31712e-02 -6.44596e-01 -9.80275e-05
4.40486e00...
            -5.36809e00 2.41201e-01 -1.25881e00 -8.62744e-01 -3.79774e-
03...
            5. 58609e00 -7.81335e-03 3.78963e-01 -3.80005e-04
- 1.81566e00...
            2.08e01 -2.56e01 1.0e00 1.80e01];
    else
        d = [-2.33271e01 -1.89958e00 3.21440e01 1.68622e00 -4.42123e-
02...
            -1.38645e01 2.82629e-02 -3.40976e-01 6.63272e-04
2.04466e00...
            8.35474e00 1.71347e00 -1.60715e01 -1.63139e00 4.14641e-
02...
            8. 70275e00 -2.30068e-02 3.60966e-01 1.53246e-05
- 1.46166e00...
            1.115884e02 -6.452606e00 -5.337863e01 2.026986e00];
```

```matlab
        end
elseif Y1 > -4.5 && Y1 <= -0.5
    if Z1 > 0.25 && Z1 <= 0.95
        d = [2.03910e-02 7.67310e-03 8.48581e-01 -2.93086e-02 8.40269e-
04...
            2.67251e-01 -1.47701e-03 2.37262e-02 3.13687e-05 -1.41973e-
01...
            0 0 0 0 0 0 0 0 0 0 0 0 0 0 0];
    elseif Z1 > 0.95 && Z1 <= 1.45
        d = [-5.12404e00 -2.8474e-01 1.54532e01 4.52475e-01 -1.22881e-
02...
            -1.35181e01 8.56845e-03 -1.68725e-01 -3.25256e-04
4.18451e00...
            7.52564e00 8.35238e-01 -1.95558e01 -1.23393e00 3.34510e-
02...
            1. 71779e01 -2.34269e-02 4.54628e-01 4.81788e-04 -
5.09936e00...
            6.148442e01 -1.828123e01 -5.468755e01 1.562500e01];
    elseif Z1 > 1.45 && Z1 <= 2.05
        d = [-1.23779e01 -1.14728e00 2.41382e01 1.38957e00 -3.63693e-
02...
            - 1.42844e01 2.24265e-02 -4.06553e-01 -3.23888e-04
2.8762e00...
            4.40782e00 1.33046e00 -1.15405e01 -1.59892e00 5.30580e-
02...
            8. 57309e00 -3.10376e-02 4.71274e-01 4.77650e-04 -
1.96233e00...
            1.4075e02 -6.499992e00 -7.75e01 5.0e00];
    else
        d = [-1.27244e01 -1.66684e00 1.72708e01 1.45307e00 -3.64515e-
02...
            -6.97208e00 1.90463e-02 -3.04323e-01 4.80787e-04 9.67524e-
01...
            7.71330e00 5.0834e-01 -9.8211e00 -4.49138e-01 -9.41787e-
04...
            4.16530e00 -2.40293e-03 9.63923e-02 -8.28450e-04 -5.88807e-
01...
            -1.092654e03 -3.05312e02 4.656243e02 1.312498e02];
    end
else
    if Z1 > 0.25 && Z1 <= 1.00
        d = [-1.54141e-03 6.58337e-04 9.82201e-01 -3.85028e-03
1.23111e-04...
            3.77441e-02 -4.08210e-04 4.56963e-03 2.13592e-05 -2.35172e-
02...
            0 0 0 0 0 0 0 0 0 0 0 0 0 0 0];
    elseif Z1 > 1.00 && Z1 <= 1.45
        d = [8.06492e-01 9.91293e-02 -1.70742e00 -2.28264e-01 5.03500e-
03...
            3. 02351e00 -6.13927e-03 1.31574e-01 1.69824e-04
- 1.12755e00...
            -1.17930e-01 -2.12207e-01 1.36524e00 4.05886e-01 -1.88260e-
02...
```

```matlab
                -2.10926e00 1.65486e-02 -1.89881e-01 -5.1140e-04 8.79806e-
01...
                1.959604e02 -4.269391e01 -1.734931e02 3.762898e01];
        else
            d = [-1.66249e00 -8.91113e-02 4.11648e00 8.78093e-02 -3.09742e-
03...
                -1.84445e00 1.99879e-03 -7.50324e-03 6.85472e-05 3.05784e-
01...
                1.11555e01 1.3210e00 -1.71236e01 -1.2919e00 6.28124e-02...
                8. 63804e00 -3.07949e-02 3.07809e-01 1.57743e-03 -
1.42634e00...
                1.330611e02 8.979635e00 -7.265298e01 -2.449009e00];
        end
end

Temp_final(1,v) =
273.15*10^(d(1,1)+d(1,2)*Y1+d(1,3)*Z1+d(1,4)*Y1*Z1+...

d(1,5)*Y1^2+d(1,6)*Z1^2+d(1,7)*Z1*Y1^2+d(1,8)*Y1*Z1^2+d(1,9)*Y1^3+ ...
    d(1,10)*Z1^3+(d(1,11)+d(1,12)*Y1+d(1,13)*Z1+d(1,14)*Y1*Z1+...
    d(1,15)*Y1^2+d(1,16)*Z1^2+d(1,17)*Z1*Y1^2+d(1,18)*Y1*Z1^2+...
    d(1,19)*Y1^3+d(1,20)*Z1^3)/(1+exp(d(1,21)+d(1,22)*Y1+d(1,23)*Z1+...
    d(1,24)*Y1*Z1)));

end

plot(beta_r,Temp_final),xlabel('Beta Angle in radians'),...
    ylabel('Temperature in degrees K'),...
    title('Temperature Immediately Behind Shock')
```

Appendix F

```matlab
%% Radiative Energy as a function of Beta angle

% absorption coefficient as a function of radiation frequency using the
% Bibermann Model (figure 18.11 Anderson Hypersonic Book)

h_eV = 4.135667e-15; % Plank's constant in eV*s h
= 6.62606957e-34; % Plank's constant in J*s
kb = 1.3806488e-23; % Boltzmann constant in J*K^-1
C = 2.99792458e08; % speed of light in m/s
Ts = Temp_final; % Temp behind the shock wave as a function of Beta
angle
J = zeros(1,count1);

for v = 1:count1

if rho2_final(1,v)/1.225 > 1e-06 && rho2_ final(1,v)/1.225 <= 1e-
    02 if Ts(1,v) > 7000 && Ts(1,v) <= 9000
        abscoef = [5.001e01 5.001e01 8.629 8.629 2.786 2.786 3.457e-
01...
            3.457e-01 3.404e-03 3.404e-03 4.285e-04 4.285e-04];
    elseif Ts(1,v) > 9000 && Ts(1,v) <= 11000
        abscoef = [4.606e01 4.606e01 1.067e01 1.067e01 5.012 5.012...
            8.511e-01 8.511e-01 2.443e-02 2.443e-02 4.571e-03 4.571e-
03];
    elseif Ts(1,v) > 11000 && Ts(1,v) <= 13000
        abscoef = [3.319e01 3.319e01 1.004e01 1.004e01 5.433 5.433
1.132...
            1.132 5.47e-02 5.47e-02 1.69e-02 1.69e-02];
    elseif Ts(1,v) > 13000 && Ts(1,v) <= 15000
        abscoef = [1.577e01 1.577e01 5.902 5.902 3.35 3.35 8.091e-01...
            8.091e-01 6.152e-02 6.152e-02 2.786e-02 2.786e-02];
    elseif Ts(1,v) > 15000 && Ts(1,v) <= 17000
        abscoef = [5.297 5.297 2.291 2.291 1.355 1.355 3.681e-01
3.681e-01...
            3.945e-02 3.945e-02 2.564e-02 2.564e-02];
    end
elseif rho2_final(1,v)/1.225 > 1e-02 && rho2_final(1,v)/1.225 <= 1e-
    01 if Ts(1,v) > 7000 && Ts(1,v) <= 9000
        abscoef = [4.678e02 4.678e02 8.418e01 8.418e01 2.583e01
2.583e01...
            3.214 3.214 2.57e-02 2.57e-02 8.318e-03 8.318e-03];
    elseif Ts(1,v) > 9000 && Ts(1,v) <= 11000
        abscoef = [4.88e02 4.88e02 1.123e02 1.123e02 5.304e01
5.304e01...
            8.913 8.913 1.242e-01 1.242e-01 4.943e-02 4.943e-02];
```

```matlab
    elseif Ts(1,v) > 11000 && Ts(1,v) <= 13000
        abscoef = [4.389e02 4.389e02 1.294e02 1.294e02 7.241e01
7.241e01...
            1.475e01 1.475e01 3.236e-01 3.236e- 01 2.415e-01 2.415e-
    01]; elseif Ts(1,v) > 13000 && Ts(1,v) <= 15000
        abscoef = [3.37e02 3.37e02 1.212e02 1.212e02 7.295e01
7.295e01...
            1.702e01 1.702e01 5.483e-01 5.483e- 01 7.194e-01 7.194e-
    01]; elseif Ts(1,v) > 15000 && Ts(1,v) <= 17000
        abscoef = [2.107e02 2.107e02 8.833e01 8.833e01 5.48e01
5.48e01...
            1.417e01 1.417e01 6.546e-01 6.546e-01 1.002 1.002];
    end
end
lambda_angstrom = [400 852 852 911 911 1020 1020 1130 1130 1800 1800
10000];
lambda_metric = lambda_ angstrom*1e-10; %wavelength in
meters fq = C./lambda_metric;

sl = size(fq);
sa = size(abscoef);
frequency = zeros(1,sl(1,2));
absorptioncoeff = zeros(1,sa(1,2));
coun = 0;

for gg = 1:1:sl(1,2)
    frequency(1,gg) = fq(1,sl(1,2)-coun);
    absorptioncoeff(1,gg) = abscoef(1,sl(1,2)-coun);
    coun = coun+1;
end

B_v = 2*h*frequency.^3./(C^2*(exp(h*frequency/(kb*Ts(1,v)))-1));
int1 = absorptioncoeff.*B_v;
A1 = trapz(frequency,int1);
J(1,v) = A1;

end

E = 4*pi*J;
coefs = polyfit(beta_r,E,order);  % E polynomial fit coefficients
xe=linspace(0,beta_r(1,count1),100);
ye=coefs(1,1)*xe.^6+coefs(1,2)*xe.^5+coefs(1,3)*xe.^4+coefs(1,4)*xe.^3+
...
    coefs(1,5)*xe.^2+coefs(1,6)*xe+coefs(1,7);
plot(beta_r,E,'*',xe,ye),xlabel('Beta Angle in radians'),...
    ylabel('Radiative Energy in J/s/m^3'),...
    title('Beta Angle vs Radiative Energy'),...
    legend('data points','polynomial fit','location','SouthWest')

f_b = conv(coefs,coef);
```

Appendix G

```matlab
%  Integrating equation 18.14 from Anderson Hypersonic Book for
q_radiative

%where B is the variable for Beta Angle which we desire to integrate
from
%pi/2 to zero
F = @(B)sin(B).*cos(B).*(f_b(1,1)*B.^12 + f_b(1,2)*B.^11 +
f_b(1,3)*B.^10 +...
    f_b(1,4)*B.^9 + f_b(1,5)*B.^8 + f_b(1,6)*B.^7 + f_b(1,7)*B.^6 + ...
    f_b(1,8)*B.^5 + f_b(1,9)*B.^4 + f_b(1,10)*B.^3 + f_b(1,11)*B.^2 +
...
    f_b(1,12)*B + f_b(1,13));


Q_inf_slab = E(1,1)*X_wall1(1,num)*CL/2


Q_metric = quad(F,0,pi/2)/2  % j/s/m^2
Q_english = Q_metric*0.000088055   %btu/s/ft^2


GAMMA = E(1,1)*r(1,1)/(0.5*rho1*u1^3)


%from figure 5 in anderson engineering survey of radiative shock layers
GAMMAx = [0 1/60 0.06 0.125 0.218 0.35 0.4 0.6 0.8 1.2 1.6 2 2.4 6];
f_of_GAMMA = [1 0.9 0.8 0.7 0.6 0.48 0.45 0.36 0.3 0.23 0.19 0.17 0.15
0.11];


correction_factor = pchip(GAMMAx,f_of_GAMMA,GAMMA)


Q_metric_corrected = Q_metric*correction_factor
Q_english_corrected = Q_english*correction_factor
```

Appendix H

```
%  Integrating equation 18.14 from Anderson Hypersonic Book for
q_radiative

%where B is the variable for Beta Angle which we desire to integrate
from
%pi/2 to zero
F = @(B)sin(B).*cos(B).*(f_b(1,1)*B.^12 + f_b(1,2)*B.^11 +
f_b(1,3)*B.^10 +...
    f_b(1,4)*B.^9 + f_b(1,5)*B.^8 + f_b(1,6)*B.^7 + f_b(1,7)*B.^6 + ...
    f_b(1,8)*B.^5 + f_b(1,9)*B.^4 + f_b(1,10)*B.^3 + f_b(1,11)*B.^2 +
...
    f_b(1,12)*B + f_b(1,13));


Q_inf_slab = E(1,1)*X_wall1(1,num)*CL/2


Q_metric = quad(F,0,pi/2)/2   % j/s/m^2
Q_english = Q_metric*0.000088055    %btu/s/ft^2


GAMMA = E(1,1)*r(1,1)/(0.5*rho1*u1^3)


%from figure 5 in anderson engineering survey of radiative shock layers
GAMMAx = [0 1/60 0.06 0.125 0.218 0.35 0.4 0.6 0.8 1.2 1.6 2 2.4 6];
f_of_GAMMA = [1 0.9 0.8 0.7 0.6 0.48 0.45 0.36 0.3 0.23 0.19 0.17 0.15
0.11];


correction_factor = pchip(GAMMAx,f_of_GAMMA,GAMMA)


Q_metric_corrected = Q_metric*correction_factor
Q_english_corrected = Q_english*correction_factor
```

Appendix I

```matlab
%% Estimating Q_rad for emitting and absorbing infinite slab of air
% Using Improved Absorption Coefficient Data for Air from NASA TR R-299
T_shock = Ts(1,1);

if rho2_final(1,1)/1.225 > 1e-04 && rho2_ final(1,1)/1.225 <= 1e-
    02 if T_shock > 7000 && T_shock <= 9000
        abscoef = [5.001e01 5.001e01 8.629 8.629 2.786 2.786 3.457e-
01...
            3.457e-01 3.404e-03 3.404e-03 4.285e-04 4.285e-04];
    elseif T_shock > 9000 && T_shock <= 11000
        abscoef = [4.606e01 4.606e01 1.067e01 1.067e01 5.012 5.012...
            8.511e-01 8.511e-01 2.443e-02 2.443e-02 4.571e-03 4.571e-
03];
    elseif T_shock > 11000 && T_shock <= 13000
        abscoef = [3.319e01 3.319e01 1.004e01 1.004e01 5.433 5.433
1.132...
            1.132 5.47e-02 5.47e-02 1.69e-02 1.69e-02];
    elseif T_shock > 13000 && T_shock <= 15000
        abscoef = [1.577e01 1.577e01 5.902 5.902 3.35 3.35 8.091e-01...
            8.091e-01 6.152e-02 6.152e-02 2.786e-02 2.786e-02];
    elseif T_shock > 15000 && T_shock <= 17000
        abscoef = [5.297 5.297 2.291 2.291 1.355 1.355 3.681e-01
3.681e-01...
            3.945e-02 3.945e-02 2.564e-02 2.564e-02];
    end
elseif rho2_final(1,1)/1.225 > 1e-02 && rho2_final(1,1)/1.225 <= 1e-
    01 if T_shock > 7000 && T_shock <= 9000
        abscoef = [4.678e02 4.678e02 8.418e01 8.418e01 2.583e01
2.583e01...
            3.214 3.214 2.57e-02 2.57e-02 8.318e-03 8.318e-03];
    elseif T_shock > 9000 && T_shock <= 11000
        abscoef = [4.88e02 4.88e02 1.123e02 1.123e02 5.304e01
5.304e01...
            8.913 8.913 1.242e-01 1.242e-01 4.943e-02 4.943e-02];
    elseif T_shock > 11000 && T_shock <= 13000
        abscoef = [4.389e02 4.389e02 1.294e02 1.294e02 7.241e01
7.241e01...
            1.475e01 1.475e01 3.236e-01 3.236e- 01 2.415e-01 2.415e-
01]; elseif T_shock > 13000 && T_shock <= 15000
        abscoef = [3.37e02 3.37e02 1.212e02 1.212e02 7.295e01
7.295e01...
            1.702e01 1.702e01 5.483e-01 5.483e- 01 7.194e-01 7.194e-
01]; elseif T_shock > 15000 && T_shock <= 17000
        abscoef = [2.107e02 2.107e02 8.833e01 8.833e01 5.48e01
5.48e01...
```

```matlab
                1.417e01 1.417e01 6.546e-01 6.546e-01 1.002 1.002];
    end
end
lambda_angstrom = [400 852 852 911 911 1020 1020 1130 1130 1800 1800
10000];
lambda_metric = lambda_ angstrom*1e-10;
fq = C./lambda_metric;

sl = size(fq);
sa = size(abscoef);
frequency = zeros(1,sl(1,2));
absorptioncoeff = zeros(1,sa(1,2));
coun = 0;

for gg = 1:1:sl(1,2)
    frequency(1,gg) = fq(1,sl(1,2)-coun);
    absorptioncoeff(1,gg) = abscoef(1,sl(1,2)-coun);
    coun = coun+1;
end

size_d _delta = 100; % number of points between stagnation-point &
shock
nu1 = frequency;
fun1 = absorptioncoeff;
sz = size(nu1);
e1 = zeros(size_d_delta,sz(1,2));
deltas = X_wall1(1,num)*CL*fun1;
atin = zeros(100,sz(1,2));

for k = 1:1:sz(1,2)
    NU = nu1(1,k);
    abcoef = fun1(1,k);
    cntt = 0;
    n = linspace(0,X_ wall1(1,num)*CL*abcoef,size_d_delta);
    sizee = size(n);
    for th = 1:1:sizee(1,2)
        cntt = cntt+1;
        nn = linspace(1e-10,1,100);
        e2 = exp(-n(1,th)./nn);
        e3 = trapz(nn,e2); % radiative attenuation term
        atin(cntt,k) = e3;
        %e3 = 1;
        e1(cntt,k) = 2*pi*(2*h*NU^3/(C^2*(exp(h*NU/(kb*Ts(1,1)))-
1)))*e3;
    end
end

% position of each e1 point based on d_frequency and d_delta

% y's
y_s = zeros(size_d_delta,sz(1,2));
```

```matlab
for ys = 1:1:sz(1,2)
    y_s(:,ys) = linspace(0,deltas(1,ys),size_d_delta);
end


% x's
x_s = zeros(size_d_delta,sz(1,2));


for xs = 1:1:size_d_delta
    x_s(xs,:) = nu1;
end


% proceed with calculating the volume of each discritized cell using the
% technique presented by AIAA paper "Calculation of the Volume of a General
% Hexahedron for Flow Predictions" by Davies and Salmond (1985)
V_rad = zeros(size_d_delta-1,sz(1,2)-1);


for dd = 1:1:(size_d_delta-1)
    for df = 1:1:(sz(1,2)-1)
        AA = [x_s(dd,df)  y_s(dd,df)  e1(dd,df)];
        BB = [x_s(dd+1,df)  y_s(dd+1,df)  e1(dd+1,df)];
        CC = [x_s(dd+1,df+1)  y_s(dd+1,df+1)  e1(dd+1,df+1)];
        DD = [x_s(dd,df+1)  y_s(dd,df+1)  e1(dd,df+1)];
        EE = [x_s(dd,df)  y_s(dd,df)  0];
        FF = [x_s(dd+1,df)  y_s(dd+1,df)  0];
        GG = [x_s(dd+1,df+1)  y_s(dd+1,df+1)  0];
        HH = [x_s(dd,df+1)  y_s(dd,df+1)  0];
        cross1 = cross((BB-DD),(CC-AA));
        cross2 = cross((EE-BB),(FF-AA));
        cross3 = cross((DD-EE),(HH-AA));
        cross4 = cross1;
        cross5 = cross((CC-GG),(CC-FF));
        cross6 = cross2;
        cross7 = cross((FF-GG),(FF-HH));
        cross8 = cross3;
        cross9 = cross((HH-GG),(HH-CC));
        V_rad(dd,df) = (1/12)*dot((AA-GG),(cross1+cross2+cross3))+...
            (1/12)*dot((BB-GG),(cross4+cross5))+...
            (1/12)*dot((EE-GG),(cross6+cross7))+...
            (1/12)*dot((DD-GG),(cross8+cross9));
    end
end


Vrad = abs(V_rad);
sumVrad = sum(Vrad);
q_rad_metric = sum(sumVrad')  % J/s/m^2
q_rad_english = q_rad_metric*0.000088055  % BTU/s/ft^2
```

Appendix J

```
%% Andersons Method for GAMMA effective
%Using Improved Absorption Coefficient Data TR R-299
Kabsorb2 = abscoef(1,9:sl(1,2));


Kabsorb1 = abscoef(1,1:8);


T_shock = Ts(1,1);
rho_shock = rho2_final(1,1);
epslon = rho_ratio(1,1);
lam2 = lambda_metric(1,9:sl(1,2)); % long wavelengths, lower
frequencies
lam1 = lambda_metric(1,1:8); % short wavelengths, higher frequencies
B_lam1 = 2*h*C^2./((exp(h*C./(lam1*kb*T_shock))-1).*lam1.^5); B_lam2
= 2*h*C^2./((exp(h*C./(lam2*kb*T_shock))-1).*lam2.^5);


num1_area = trapz(lam1,(B_lam1.*Kabsorb1));
den1_area = trapz(lam1,B_lam1);
num2_area = trapz(lam2,(B_lam2.*Kabsorb2));
den2_area = trapz(lam2,B_lam2);


kappa_1s = num1_area/den1_area;
kappa_2s = num2_area/den2_area;


% estimated shock detachment distance in meters
delta_AB = CL*epslon/(1+sqrt(2*epslon));


tau_1s = kappa_1s*delta_AB;  % shock-layer optical thickness


nn = linspace(1e-10,1,100);
e2 = exp(-tau_1s./nn);
e3 = trapz(nn,e2);  % radiative attenuation term


GAMMA_eff = 4*pi*(exp(-tau_1s)*kappa_1s*den1_area +
kappa_2s*den2_area)*...
    delta_AB/(rho1*u1^3/2);
correction_factor1 = spline(GAMMAx,f _of_GAMMA,GAMMA_eff)
Q_r = 2*pi*den1_area*e3*tau_1s +...
    2*pi*kappa_2s*den2_area*delta_AB*correction_factor1
Q_r_english = Q_r*0.000088055
```

Appendix K

```matlab
%  Final Stagnation Radiative Flux (including shock-layer cooling
factor)

Q_RADIATION_FINAL_METRIC = q_rad_metric*correction_factor1
Q_RADIATION_FINAL_ENGLISH = q_rad_english*correction_factor1

engr = zeros(11,1);
engr(1:5,1) = sumVrad(1,1:5);
engr(6:11,1) = sumVrad(1,6:11)*correction_factor1;
sum(engr)*0.000088055;
```