SOUNDING ROCKET REDESIGN AND OPTIMIZATION FOR PAYLOAD EXPANSION AND IN FLIGHT TELEMETRY TRANSMITTAL

by

MATTHEW ADAM HUFFMAN
B.S.A.E. University of Central Florida 2003

A thesis submitted in partial fulfillment of the requirements for
the degree of Master of Science
from the Mechanical, Materials and Aerospace Engineering Department
in the College of Engineering and Computer Science
at the University of Central Florida
Orlando, Florida

Spring Term
2005

ABSTRACT

Due to renewed interest in the sub orbital rocket program of the Florida Space Authority and a surplus of Super Loki Sounding Rockets, an effort to improve the usefulness of this surplus is herein undertaken. Currently, the capacity of the payload compartment in the upper stage of the Super Loki system is very limited. A redesign of the upper stage will allow larger and more versatile payloads to be carried, assuming the appropriate design parameters are met. It has therefore been undertaken to create a design procedure that is comprehensive in scope in order to affect this redesign.

This procedure includes five major components. These are the separation of the upper and lower stages, the stability of the vehicle, the altitude and velocity of the rocket, the mechanical loading and finally the aerodynamic heating. Semi-empirical methods were used whenever possible to allow comparison with experimental data. This procedure revealed that larger diameter upper stages might be used up to a reasonable maximum of four inches. The four-inch modification is found to be stable as were the smaller modifications considered. The altitude and velocity of the rocket were found via a simple Eulerian time stepping scheme resulting in an estimate of approximately 148,000ft for the four-inch dart. The mechanical loading analysis allowed for the material selection for the rocket components. Reinforced steel fins and carbon fiber tubing, for the payload section, are adequate to meet expected mechanical loads, those being, 16000psi for the fin section due to launcher forces, 22800psi for compressive plus torsion forces on the composite section and 18000psi for the ejection stresses. An ablative coating is considered necessary to counteract the 760^{o}F temperatures along the composite tube.

ACKNOWLEDGEMENTS

To my wife Wendi, who carried us through the years and put up with many days and nights alone. Thank you for your patience, understanding and love.

To my parents, blood and law, for showing me, by example, how to both understand and care for the world and pursue goals that might help to make the world a better place. Not to mention the tuition.

To my advisor, Dr. Larry Chew, who taught me what it means to be an Engineer and always inspired me to reach for higher and higher standards.

To Mr. Robert Eppig, who's guidance, knowledge and uncommon common sense always brought new light to any problem.

I would also like to thank the faculty and staff of the MMAE Department for their flexibility, dedication and professionalism throughout my time at UCF.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

NOMENCLATURE

A	Axial Force
a	Speed of sound
Ae	Exit Area (Nozzle)
AR	Aspect Ratio
Ca	Coefficient of axial force
Cf	Coefficient of Friction
cg	Center of gravity
Cm	Coefficient of moment
cmac	Mean aerodynamic chord
Cn	Coefficient of normal force
Cp	Coefficient of pressure
cp	Center of pressure
D	Drag
G	Heat absorption capacity
H	Height above sea level, thickness
h	Height above sea level, thickness
I	Moment of inertia
i	unit vector, enthalpy
J	Mechanical equivalent of heat
L	Lift
M	Mach number
m	mass flow
N	Normal Force
P	Pressure
Pr	Prandtl number
q	Dynamic pressure, heat transfer
R	Radius of Curvature, Universal Gas constant
Re	Reynolds number
r	Radial coordinate, radius, recovery factor
S	Area
s	Shock wave detachment distance
T	Temperature
u	x component of velocity
V	Vehicle velocity
v	y component of velocity
w	z component of velocity
x	Longitudinal axis
y	Transverse axis in vertical plane
z	Transverse axis in horizontal plane
α	Angle of Attack
β	Mach factor, velocity gradient
δ	Boundary layer thickness
γ	Ratio of specific heat
ρ	Density
τ	Shear stress
ϕ	Radial angle
θ	Thickness of ablator, tangent angle of body surface
ψ	Cross flow component
π	3.14159 radians
ω	Angular velocity
ε	Coefficient of turbulent viscosity, emissivity, boat tail angle
μ	Molecular viscosity, Mach angle
ν	Kinematic viscosity
λ	Slenderness ratio, velocity gradien (non-dimensional)
σ	Stefan Boltzmann Constant, Stress

Subscripts:

x,y,z	Cartesian coordinate axes
1,2,3	Principal axes
∞	Free stream conditions, outer side of boundary layer
n	normal
s	tangential
i	inner radius
o	outer radius
ref	reference quantity
0	Stagnation quantitu
b	base
w	wetted area
D	diffusive term
t	thermal conductivity term

CHAPTER 1: INTRODUCTION

1.1 Motivation

The Florida Space Authority has in its possession a surplus of about 200 Super Loki Sounding rockets. These rockets were developed in the 1960's to carry weather sonde payloads to high altitudes to send back atmospheric data. Due to the advent of weather satellites, these payloads are no longer necessary. Super Lokis are now primarily used for educational purposes. There is a shortage, however, of the largest of three inert (no propellant) upper stages, or darts. Two of these darts (designated PWN-11 and -12) are designed only to carry metallic charutes (parachutes) and/or very small meteorological payloads and consequently have very small payload capacities. Their inner diameters range from 0.87 inch to 1.75-inch i.d. on the largest dart, the PWN-10 and therefore cannot support very versatile payloads. It is desired, therefore, that a larger dart be developed based on the same booster so that more versatile payloads may be developed thereby increasing the educational usefulness of these rockets.

1.2 Design Criteria

In order to begin the design process, a description of the parameters involved is necessary, and is given in this chapter. The Super Loki goes through a wide range of extremes throughout its flight. The acceleration during lift off is in the vicinity of 60g's. For the first 0.1 seconds (approximate) the rocket is in contact with its launcher, which spins the rocket up to an angular velocity of around 7 revolutions per second. This is accomplished with spiral guide rails in which the dart fins are guided.

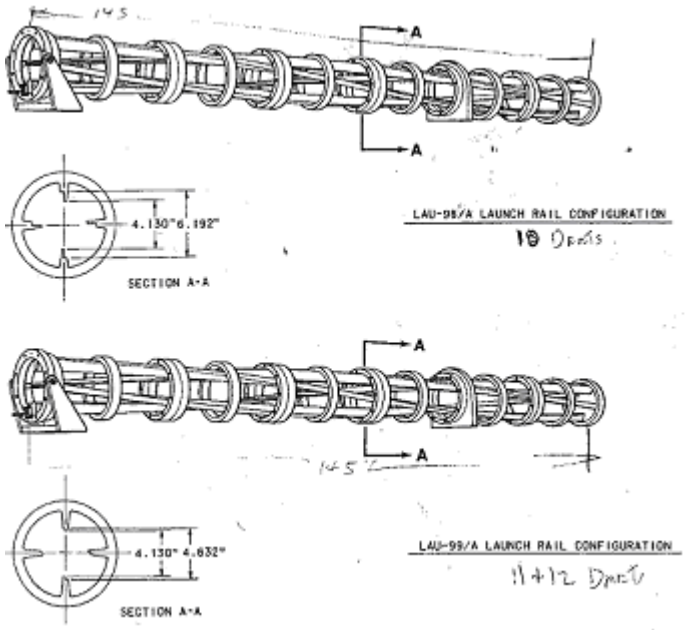

Figure 1: Super Loki Launcher

The maximum velocity obtained, for the current configuration, at motor burnout, is approximately 1629m/s (5342ft/s), which occurs at around 2000m (6562ft). The booster, or first stage, then separates from the dart due to a differential in aerodynamic drag. This initial boost sends the upper stage to an apogee of around 75000m (270000ft). Large temperature, pressure and density fluctuations occur during this approximately 120-second flight. Finally, approximately 250g's are experienced by the payload and dart during ejection. The values given in this section are derived from [3].

In order to achieve a successful design, the physical parameters regarding the mechanical and thermal loads, as well as the flight stability criteria, given the information above, must be accurately quantified. An overview of the requirements to meet the various criteria follows.

1.3 Stage Separation

In the current system the upper and lower stages separate, after motor burnout, due to a difference in aerodynamic drag between the dart and the booster. In other words the drag on the motor is greater, due to its larger diameter and length, and therefore it is decelerated more rapidly than the dart. The two pieces therefore separate. If the proposed dart is too large in diameter, fin area, or a combination of factors resulting in too large a drag, it may not separate properly from the motor.

1.4 Stability

There is little data available on the Super Loki in regards to stability, as well as aerodynamic heating and structural loading, however a great deal of information exists regarding rockets and ballistics in general. Additionally, the method developed in this paper, applied to the Super Loki with the existing dart, should at the very least provide a stable result, as the Super Loki is a well-tested stable platform including the additional stability imparted by the launcher.

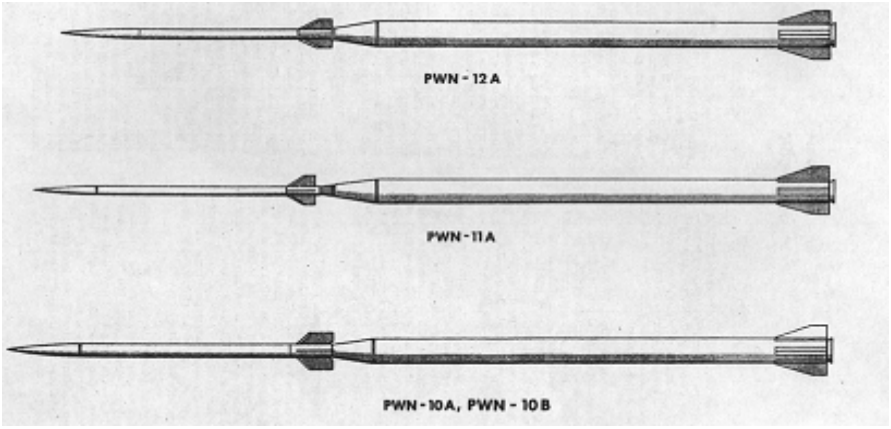

Figure 2: Super Loki Configurations

This method will then be applied to the larger capacity dart, once a check on all of the applicable assumptions has been made, and geometric requirements for stability determined. These geometric requirements consist of maximum diameter, nose cone shape, fin shape and size, fineness ratio (ratio of length to diameter), etc.

1.5 Trajectory

The trajectory of the rocket must be calculated, at least approximately, in order to get an idea of the rockets maximum altitude and velocity. This will be accomplished by using a Eulerian time stepping procedure, assuming no drag over a discrete time interval, and using thrust data from [3]. The average thrust of 17885N (4021 lbs) will be assumed constant over the burn time period, as the actual thrust curve for the boosters is unavailable. The velocity, thus calculated, will be modified by calculating the drag at this speed and then subtracting the result from the no drag value. This procedure is carried out until the end of the burn period. After this time, only the forces of drag and gravity are considered. The aerodynamic characteristics of the booster-dart combination are used during burn and the dart alone during coast.

1.6 Mechanical Loading

1.6.1 Launcher-Rocket Interaction

The Super Loki is launched in a fixed launcher with helix shaped, or spiral, guide rails that spin the rocket as it traverses the launcher. It is estimated that the rocket leaves the launcher in approximately 0.1 second. Given the moment of inertia of the rocket, the impulse-momentum

theorem may be used to find the forces put on each component of the dart as will be discussed later. The fins on the upper stage are guided by the launcher rails and must withstand the forces put on them without significant, plastic deformation. The cylindrical portion of the dart body, as well as the nose cone and all other components, must also withstand the high level of torque experienced during launch.

Additionally, some project payloads that are flown on the proposed dart may require data transmission as the rocket coasts up to apogee. The current dart is made of steel, and typical transmitters cannot transmit through this casing. As part of this analysis, the feasibility of using alternative materials, such as composites, will be investigated, considering material strength and thermal capabilities as regards to mechanical loads and aerodynamic heating respectively.

1.6.2 Ejection Forces

Along with the launcher-rocket interaction forces, the force of payload ejection is a significant consideration if composite materials are to be used. According to accelerometer data given in [16], 250g's are experienced by the payload, and therefore the rest of the dart, in the current configuration. If a material with lower strength than that currently used for the cylindrical portion of the body is used, it must be assured that this component will not rupture before the nose cone and payload are expelled.

1.7 Aerodynamic Heating

As given above, at motor burnout, the Super Loki rocket, current configuration, reaches a speed of 1629m/s (5342ft/s). This is considered to be the maximum speed and it occurs at approximately 2000m (6562ft). This speed at such a low altitude, and therefore relatively high density, results in a great deal of aerodynamic resistance and friction. Heating is therefore expected to be significant enough to warrant a thorough investigation. Although no data is available from the manufacturer, and very little from other sources, the original dart was made of steel and covered with ablative coating. This suggests that the manufacturer expected a high level of heating as well. Heating is not only important to the rocket structure but also, and perhaps more so, to the payload within.

CHAPTER 2: FLOW ABOUT BODIES OF REVOLUTION

2.1 Equations of Motion

Missile aerodynamics allow for simplifications to the general equations of flow about arbitrary bodies. The flow about bodies of revolution is either axially symmetric or plane symmetric if there are no protuberances present on the body. Following closely the development in [2] of the equations of flow about bodies of revolution, this simplification is derived below. Given the specific nature of this analysis, to that of bodies of revolution, an explanation of these governing equations was considered warranted.

2.1.1 The Cartesian Coordinate System

$$\frac{d\overline{V}}{dt} = -\frac{1}{\rho} \cdot grad(p) + \frac{1}{\rho} \cdot \left(\frac{\partial \overline{\tau}_x}{\partial x} + \frac{\partial \overline{\tau}_y}{\partial y} + \frac{\partial \overline{\tau}_z}{\partial z} \right) \tag{1}$$

Equation (1) is Newton's second law applied to a fluid element. ρ is the fluid density, $grad(p)$ is the pressure gradient and is equal to $\frac{\partial p}{\partial x} \cdot i_1 + \frac{\partial p}{\partial y} \cdot i_2 + \frac{\partial p}{\partial z} \cdot i_3$ where the i values are the three orthogonal unit vectors. $\partial \overline{\tau}_x$, $\partial \overline{\tau}_y$, $\partial \overline{\tau}_z$ are the shear force vectors and are equal to $p_x \cdot i_1 + p_y \cdot i_2 + p_z \cdot i_3$ respectively.

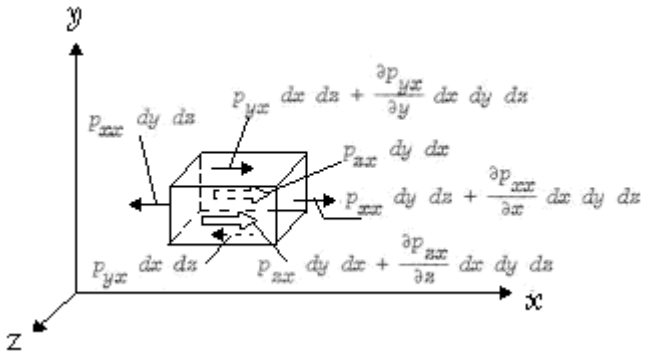

Figure 3: Forces on a volume of fluid

This equation describes the motion of a weightless, nonconductive, compressible, viscous gas. In some situations such as obtaining a first approximation to a flow field, the viscous forces may be neglected. The viscous forces may then be accounted for using the boundary layer method in the manner of Prandtl as in [13]. The inviscid flow may be described by Euler's equation:

$$\frac{d\overline{V}}{dt} = -\frac{1}{\rho} \cdot grad(p) \qquad (2)$$

This equation applies to rotational and irrotational flow.

2.1.2 Cylindrical Coordinate System

For convenience, in the application of this equation to bodies of revolution (axially symmetric bodies), we transform to cylindrical coordinates. The pressure gradient becomes:

$\frac{\partial p}{\partial x}\cdot i_1+\frac{\partial p}{\partial r}\cdot i_2+\frac{1}{r}\frac{\partial p}{\partial \phi}\cdot i_3$. Designating the velocity components in this coordinate system, see figure 4, as V_x, V_r, and V_ϕ, the equation of motion can be expanded to:

$$\begin{aligned}
&\frac{\partial V_x}{\partial t}+V_x\cdot\frac{\partial V_x}{\partial x}+V_r\cdot\frac{\partial V_x}{\partial r}+\frac{V_\phi}{r}\cdot\frac{\partial V_x}{\partial \phi}=-\frac{1}{\rho}\cdot\frac{\partial p}{\partial x},\\
&\frac{\partial V_r}{\partial t}+V_x\cdot\frac{\partial V_r}{\partial x}+V_r\cdot\frac{\partial V_r}{\partial r}+\frac{V_\phi}{r}\cdot\frac{\partial V_r}{\partial \phi}-\frac{V_\phi^{\,2}}{r}=-\frac{1}{\rho}\cdot\frac{\partial p}{\partial r},\\
&\frac{\partial V_\phi}{\partial t}+V_x\cdot\frac{\partial V_\phi}{\partial x}+V_r\cdot\frac{\partial V_\phi}{\partial r}+\frac{V_\phi}{r}\cdot\frac{\partial V_\phi}{\partial \phi}+\frac{V_r\cdot V_\phi}{r}=-\frac{1}{\rho\cdot r}\cdot\frac{\partial p}{\partial \phi}
\end{aligned}\qquad (3)$$

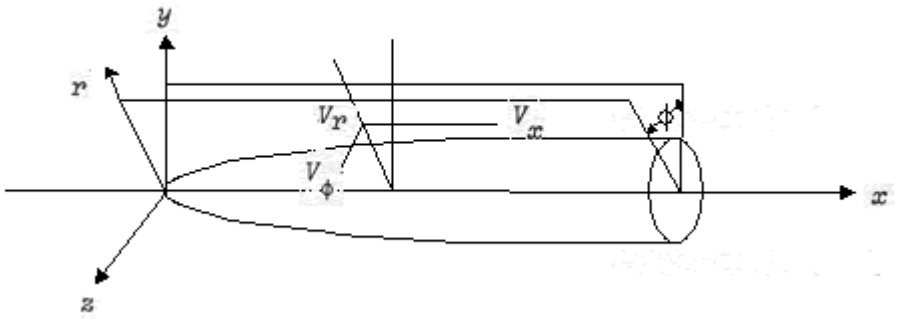

Figure 4: Velocity components cylindrical coordinate system

These equations are useful for unsteady, rotational flow around bodies of revolution of arbitrary shape at an angle of attack. In the case of steady flow, the partial derivatives with respect to time disappear and the equations of motion become:

$$\begin{aligned}
&V_x\cdot\frac{\partial V_x}{\partial x}+V_r\frac{\partial V_x}{\partial r}=-\frac{1}{\rho}\cdot\frac{\partial p}{\partial x},\\
&V_x\cdot\frac{\partial V_r}{\partial x}+V_r\frac{\partial V_r}{\partial r}=-\frac{1}{\rho}\cdot\frac{\partial p}{\partial r}
\end{aligned}\qquad (4)$$

The development of the equations above sets a foundation for the analysis of the flow about a body of revolution. For the specific cases of flow around a cone, boundary layer, or inviscid flow in the blunt nose region behind a detached shock wave, spherical and normal (or natural) coordinates systems are better suited. It is therefore desirable to develop, in turn, the equations of motion in these coordinate systems.

2.1.3 Spherical Coordinate System

$$
\begin{aligned}
&V_r \frac{\partial V_r}{\partial r} + \frac{V_\Theta}{r} \cdot \frac{\partial V_r}{\partial \Theta} + \frac{\partial V_\Psi}{r \cdot \sin(\Theta)} \cdot \frac{\partial V_r}{\partial \Psi} - \frac{\partial V_\Theta{}^2 + V_\Psi{}^2}{r} = -\frac{1}{\rho} \cdot \frac{\partial p}{\partial r}, \\
&V_r \frac{\partial V_\Theta}{\partial r} + \frac{V_\Theta}{r} \cdot \frac{\partial V_\Theta}{\partial \Theta} + \frac{\partial V_\Psi}{r \cdot \sin(\Theta)} \cdot \frac{\partial V_\Theta}{\partial \Psi} + \frac{V_r \cdot V_\Theta}{r} - \frac{V_\Psi{}^2 \cdot \cot(\Theta)}{r} = -\frac{1}{r \cdot \rho} \cdot \frac{\partial p}{\partial \Theta}, \\
&V_r \frac{\partial V_\Psi}{\partial r} + \frac{V_\Theta}{r} \cdot \frac{\partial V_\Psi}{\partial \Theta} + \frac{\partial V_\Psi}{r \cdot \sin(\Theta)} \cdot \frac{\partial V_\Psi}{\partial \Psi} + \frac{V_r \cdot V_\Psi}{r} - \frac{V_\Theta \cdot V_\Psi \cdot \cot(\Theta)}{r} = -\frac{1}{r \cdot \rho \cdot \sin(\Theta)} \cdot \frac{\partial p}{\partial \Psi}
\end{aligned}
\tag{5}
$$

In steady, symmetric flow, the flow parameters do not depend on Ψ, see figure 5 and therefore become:

$$
\begin{aligned}
&V_r \cdot \frac{\partial V_r}{\partial r} \frac{V_\Theta}{r} \cdot \frac{\partial V_r}{\partial \Theta} - \frac{V_\Theta{}^2}{r} = -\frac{1}{\rho} \cdot \frac{\partial p}{\partial r}, \\
&V_r \cdot \frac{\partial V_\Theta}{\partial r} + \frac{V_\Theta}{r} \frac{\partial V_\Theta}{\partial \Theta} + \frac{V_r \cdot V_\Theta}{r} = -\frac{1}{\rho} \cdot \frac{\partial p}{\partial \Theta}
\end{aligned}
\tag{6}
$$

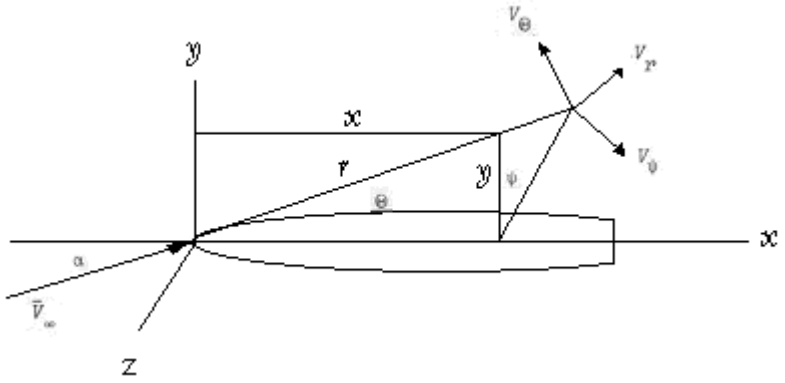

Figure 5: Velocity components in spherical coordinate system

2.1.4 Natural Coordinate System

In this coordinate system, one axis is directed tangent to the arc s of a streamline, and the other is normal n or perpendicular to the streamline at a given point. The pressure gradient is, in this case, given by: $grad\ p = \frac{\partial p}{\partial s} \cdot i_1 + \frac{\partial p}{\partial n} \cdot i_2$ and the acceleration is: $\frac{d\overline{V}}{dt} = w_s \cdot i_1 + w_n \cdot i_2$. The tangential and normal components of acceleration are given by: $w_s = V_s \cdot \frac{\partial V_s}{\partial s}$ and $w_n = V_s^{\,2} \cdot \frac{\partial \delta}{\partial s}$ where δ is the flow deflection angle. Substituting into the acceleration (motion) equation, the components of this equation become:

$V_s \cdot \frac{\partial V_s}{\partial s} = -\frac{1}{\rho} \cdot \frac{\partial p}{\partial s}$ (tangential) and $V_s^{\,2} \cdot \frac{\partial \delta}{\partial s} = -\frac{1}{\rho} \cdot \frac{\partial p}{\partial n}$ (normal)

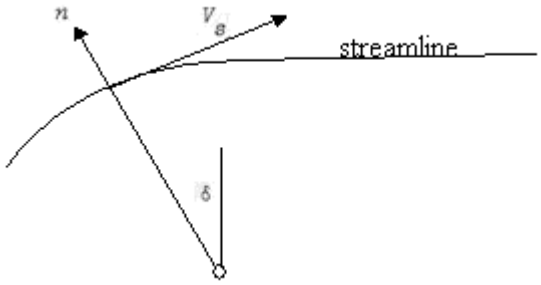

Figure 6: Vector components in natural coordinate system

As mentioned above, the normal coordinate system, see figure 6, is best suited for equations in the region of the boundary layer, and the region between a detached shock wave and the blunt nose of the body. It can also be seen in this development that symmetry has been assumed and that the equations above are intended to calculate the flow in a two dimensional cross section of the flow. The parameters at an arbitrary point in the flow M , as illustrated in figure 7, can be calculated based on its distance from the surface, with the following formulas for the velocity components, in x and y:

$$\begin{aligned} &\frac{V_x}{1+y/R}\cdot\frac{\partial V_x}{\partial x}+V_y\cdot\frac{\partial V_x}{\partial y}+\frac{V_x\cdot V_y}{R+y}=-\frac{1}{1+y/R}\cdot\frac{1}{\rho}\cdot\frac{\partial p}{\partial x}, \\ &\frac{V_x}{1+y/R}\cdot\frac{\partial V_y}{\partial x}+V_y\cdot\frac{\partial V_y}{\partial y}-\frac{V_x^{\,2}}{R+y}=-\frac{1}{\rho}\cdot\frac{\partial p}{\partial y} \end{aligned} \qquad (7)$$

Where R is the radius of curvature of the body.

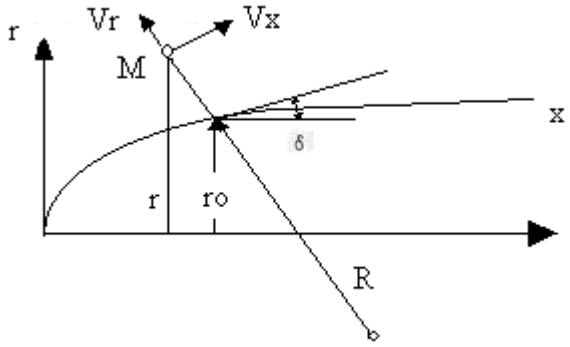

Figure 7: Velocity components in body curvilinear coordinate system

2.2 Boundary Layer Equations

The primary assumption in the development of the boundary layer equations is that the boundary layer thickness remains small compared to the linear dimensions of the body. If this assumption is valid, than the pressure at different points on the cross sectional area of the boundary layer may be assumed to be the same as that at the outer edge of the boundary layer, see figure 8. The forces acting on the stream-wise end surfaces of the fluid element in the boundary layer are independent of viscosity and will be determined by the pressure just outside of the boundary layer.

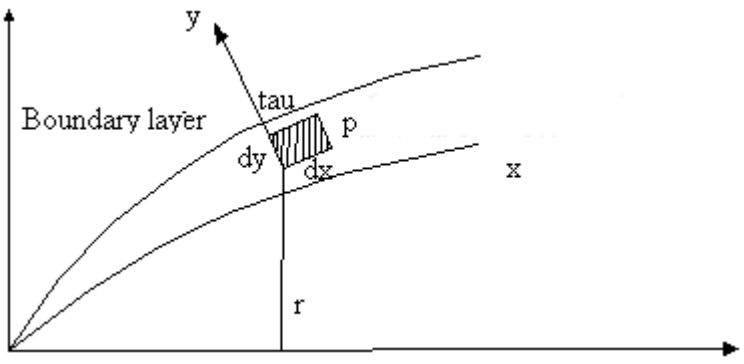

Figure 8: Forces in the boundary layer

Additionally, due to the thin nature of the boundary layer, coriolis effects are considered negligible.

The equation of motion in the boundary layer is then:

$$V_x \cdot \frac{\partial V_x}{\partial x} + V_y \cdot \frac{\partial V_x}{\partial y} = -\frac{1}{\rho} \cdot \frac{\partial p}{\partial x} + \frac{1}{r \cdot \rho} \cdot \frac{\partial (r \cdot \tau)}{\partial y} \tag{8}$$

This equation is valid for laminar and turbulent boundary layers. In the turbulent boundary layer V_x, V_y, ρ and p are assumed to be averaged in this equation and the friction stress comes from: $\tau = (\mu + \varepsilon) \cdot \frac{\partial V_x}{\partial y}$ where ε is the coefficient of turbulent viscosity and μ is the coefficient of molecular viscosity in the laminar boundary layer. In the laminar boundary layer ε is zero and the friction is given by: $\tau = \mu \cdot \frac{\partial V_x}{\partial y}$ Consequently, the equation for the laminar boundary layer becomes:

$$V_x \cdot \frac{\partial V_x}{\partial x} + V_y \cdot \frac{\partial V_x}{\partial y} = -\frac{1}{\rho} \cdot \frac{\partial p}{\partial x} + \frac{1}{r \cdot \rho} \cdot \frac{\partial}{\partial y} \cdot \left(\mu \cdot r \cdot \frac{\partial V_x}{\partial y} \right) \tag{9}$$

In calculating the velocity distribution in the boundary layer, approximate expressions of the velocity profile, for incompressible flow over a flat plate, may be used, following the Von Karman Method as in [13]. For laminar flow:

$$\frac{V_x}{V_\delta} = \frac{3}{2} \cdot \frac{y}{\delta} - \frac{1}{2} \cdot \frac{y^3}{\delta^3} + ...$$

And for turbulent flow: (10)

$$\frac{V_x}{V_\delta} = \left(\frac{y}{\delta} \right)^{1/7}$$

These expressions do not take into account compressibility or heat transfer and may only be used for low speed flow, and as a first guess in high-speed flow. The effects due to high-speed flow must be superimposed on these simplified results. In equations (10), V_x is the flow velocity at an arbitrary point in the boundary layer, and V_δ is the flow just outside of the layer. The latter velocity component may be determined by first solving the inviscid flow equations. Finally the boundary layer thickness is calculated:

$$\frac{\delta}{L} = A \cdot \bar{x}^m \cdot \mathrm{Re}^n \tag{11}$$

Where L is the length of the plate, $\bar{x}$ is the dimensionless coordinate, $\bar{x} = x/L$, and Re is the Reynolds number, $\mathrm{Re} = V_\delta \cdot L/\nu$. ν is the kinematic viscosity of the fluid.

2.3 Continuity

In addition to the equations of motion and boundary layer equations, the equations of continuity and energy must be considered. Energy will be considered in the next section. The continuity equation, for a fixed volume of gas, is given by:

$$\frac{dp}{dt} + \rho \cdot div\overline{V} = 0 \tag{12}$$

As was seen in the previous section, the cylindrical and spherical coordinate systems will be more useful in the analysis of a body of revolution than the Cartesian coordinates. The divergence of the velocity in cylindrical coordinates is:

$div\,\overline{V} = \frac{\partial V_x}{\partial x} + \frac{\partial V_r}{\partial r} + \frac{1}{r} \cdot \frac{\partial V_\phi}{\partial \phi} + \frac{V_r}{r}$. The continuity equation in this coordinate system is therefore:

$$r \cdot \frac{\partial \rho}{\partial t} + \frac{\partial(\rho \cdot r \cdot V_x)}{\partial x} + \frac{\partial(\rho \cdot r \cdot V_r)}{\partial r} + \frac{1}{r} \cdot \frac{\partial(\rho \cdot r \cdot V_\phi)}{\partial \phi} = 0 \tag{13}$$

In the case of a symmetric body of revolution the fourth term in the above equation is dropped.

In the spherical coordinate system:

$$div\,\overline{V} = \frac{1}{r^2} \cdot \frac{\partial(r^2 \cdot V_r)}{\partial r} + \frac{1}{r \cdot \sin(\Theta)} \cdot \frac{\partial(V_\Theta \cdot \sin(\Theta)}{\partial \Theta} + \frac{1}{r \cdot \sin(\Theta)} \frac{\partial V_\Psi}{\partial \Psi}$$

and the continuity equation is: (14)

$$\frac{d\rho}{dt} + \rho \cdot \left(\frac{1}{r^2} \cdot \frac{\partial(r^2 \cdot V_r)}{\partial r} + \frac{1}{r \cdot \sin(\Theta)} \cdot \frac{\partial(V_\Theta \cdot \sin(\Theta)}{\partial \Theta} + \frac{1}{r \cdot \sin(\Theta)} \frac{\partial V_\Psi}{\partial \Psi} \right)$$

For steady flow:

$$\frac{1}{r^2} \cdot \frac{\partial(\rho \cdot r^2 \cdot V_r)}{\partial r} + \frac{1}{r \cdot \sin(\Theta)} \cdot \frac{\partial(\rho \cdot V_\Theta \cdot \sin(\Theta)}{\partial \Theta} + \frac{1}{r \cdot \sin(\Theta)} \frac{\partial(\rho \cdot V_\Psi)}{\partial \Psi} = 0$$

For steady flow at zero angle of attack, the third term above can be dropped:

$$\frac{1}{r^2} \cdot \frac{\partial(\rho \cdot r^2 \cdot V_r)}{\partial r} + \frac{1}{r \cdot \sin(\Theta)} \cdot \frac{\partial(\rho \cdot V_\Theta \cdot \sin(\Theta)}{\partial \Theta} = 0 \tag{15}$$

In the normal (natural) coordinate system for steady flow:

$$\frac{\partial \delta}{\partial n} + \frac{\partial \log(\rho)}{\partial s} + \frac{\partial \log(V_s)}{\partial s} + \frac{\sin(\delta)}{r} = 0 \tag{16}$$

For a point M in the flow in the normal system:

$$\frac{\partial(\rho \cdot r \cdot V_x)}{\partial x} + \frac{\partial[\rho \cdot r \cdot V_y \cdot (1 + y/R)]}{\partial y} = 0 \tag{17}$$

In examining the boundary layer the term y/R can be neglected giving the equation for continuity in the boundary layer:

$$\frac{\partial(\rho \cdot r \cdot V_x)}{\partial x} + \frac{\partial(\rho \cdot r \cdot V_y)}{\partial y} = 0 \tag{18}$$

2.4 Energy Equation

The energy equation is the mathematical representation of the conservation of energy principal. This principal states, “the change in kinetic energy of the moving particles is equal to the sum of the work done by external forces and the energy flux caused by external transfer of heat” [2]. The energy equation for gas enclosed in an arbitrary volume:

$$\frac{d}{dt}\left(\int_{\tau} \rho \cdot \left(\frac{V^2}{2} + u \right) \cdot d\tau \right) = \int_{S} \overline{p}_n \overline{V} \cdot dS + \int_{S} q_T \cdot dS + \int_{\tau} \varepsilon_{RAD} \cdot d\tau + \int_{S} q_D \cdot dS \tag{19}$$

Where q_T is the heat rate caused by thermal conductivity and q_D that of diffusion thru a unit surface. ε_{RAD} is the radiative heat flux of a unit mass per unit time, u is the internal energy of a unit mass of gas and $\overline{p}_n$ is the surface force vector.

CHAPTER 3: STAGE SEPARATION

3.1 Solution Development

The forces on a rigid body in a fluid flow consist of pressure and shear forces (due to viscosity) as well as the thrust force applied by the motor. These forces can be integrated and resolved into a force vector. The force vector on a body in flight is usually referenced to a coordinate system, with its origin at the center of gravity, that coincides with the instantaneous angle of attack of the vehicle see figure 9.

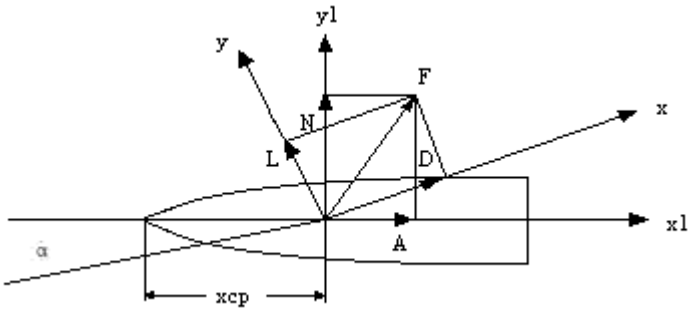

Figure 9: Force on a body of revolution

3.1.1 lift and drag

The force acting along the velocity vector is called the drag and that acting normal to the drag is called the lift. If the force is referenced to the body fixed coordinate system the forces are termed axial and normal respectively. This relation is formulated below in dimensionless coefficient form:

$$C_D = C_A \cdot \cos(\alpha) + C_N \cdot \sin(\alpha) \tag{20}$$

$$C_L = C_N \cdot \cos(\alpha) - C_A \cdot \sin(\alpha) \quad (21)$$

The coefficient forms of lift C_L and drag C_D, as well as axial C_A and normal C_N forces are found by dividing the force in question by the dynamic pressure $q = \frac{\rho_\infty \cdot V_\infty^{\,2}}{2}$ and the area, S, that the forces are applied to e.g. $C_D = \frac{D}{q \cdot S}$ where D is the force due to drag.

Another important parameter is the pitching moment coefficient C_M. The pitching moment coefficient is similar to the coefficients above but, in order to maintain non-dimensionality, must also be divided by a characteristic length i.e. $C_M = \frac{M}{q \cdot S \cdot l}$ This length is generally taken as the static margin, the distance between the center of pressure and the center of gravity.

From [21], it is possible to determine these coefficients for different parts of the dart, i.e. the nose, cylindrical portion of the body, the fins etc. and superimpose the results to get the totals. Additionally, interference between fin and body may be added to this result. These calculations can be compared to experimental results [21] on bodies of similar shape in order to quantify the accuracy of the results.

3.1.2 Normal and Axial

It is generally desirable, in missile aerodynamics, to calculate the axial and normal coefficients first as opposed to the lift and drag coefficients due to the typically small angles of attack. A development of these coefficients is given below:

The axial force coefficient can be defined as being composed of three parts:

$$C_A = C_{Ap} + C_{AB} + C_{Af} \tag{22}$$

Where C_{Ap}, C_{AB} and C_{Af} are the axial force coefficients due to pressure, base drag and friction respectively.

3.1.3 Axial due to pressure

In order to calculate these coefficients, the pressure and shear distribution over the whole body must be known. The axial force due to pressure on an elemental area of the body surface is given by:

$$dA_p = (p - p_\infty) r \cdot ds \cdot d\phi \cdot \sin(\delta) \tag{23}$$

Where *p* is the pressure on the area, p_∞ is the free-stream pressure, and $r \cdot ds \cdot d\phi \cdot \sin(\delta)$ is the area.

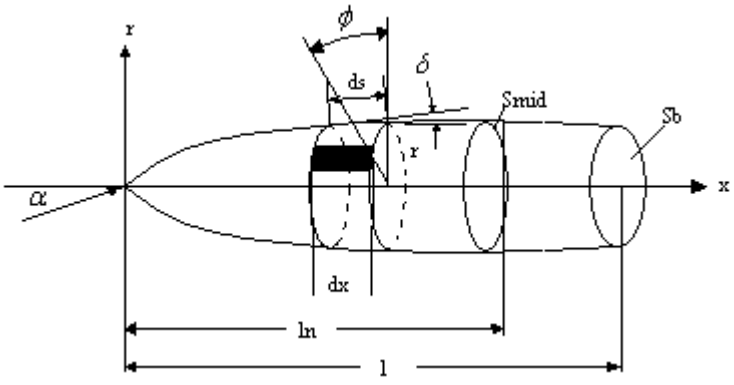

Figure 10: Variables for determining aerodynamic characteristics

Since $dx = ds \cdot \cos(\delta)$ this equation, upon integration, becomes:

$$A_p = 2 \cdot \int_0^l r \cdot \tan(\delta) dx \cdot \int_0^{\pi} (p - p_\infty) \cdot d\phi \tag{24}$$

Where l is the length of the body. The axial force due to pressure is related to the coefficient C_{Ap} by: $A_p = C_{Ap} \cdot q \cdot S_{\max}$ where $S_{\max}$ is the maximum cross sectional area of the body and is given by $S_{\max} = \pi \cdot r_{\max}^{\ 2}$. $r_{\max}$ is the radius of the maximum cross-sectional area of the body.

Now we have:

$$C_{Ap} = \frac{4 \cdot \lambda_l}{\pi} \cdot \int_0^1 \bar{r} \cdot \tan(\delta) \cdot d\tilde{x} \cdot \int_0^{\pi} C_P \cdot d\phi \tag{25}$$

Where $\bar{r} = \frac{r}{r_{mid}}$, $\tilde{x} = \frac{x}{l}$, $\lambda_l = \frac{l}{2 \cdot r_{mid}}$ (fineness ratio of the body), and $C_P = \frac{p - p_\infty}{q}$ the pressure coefficient, which is a function of $x(r)$ and ϕ.

In the case of zero angle of attack:

$$C_{Ap} = 4 \cdot \lambda_l \cdot \int_0^1 C_P \cdot \bar{r} \cdot \tan(\delta) \cdot d\tilde{x} \tag{26}$$

C_P, in this case, is only a function of either x or r.

In order to obtain the pressure distribution over the body for these conditions, the method laid out in [10] was used. This method involves the use of a line of sources and boundary conditions such that the velocity at the surface of the body is zero. The pressure distribution may then be integrated to find the magnitude and center of pressure of the body alone, thereby allowing the fins to be sized accordingly.

3.1.4 Axial due to base pressure

The base drag coefficient, C_{AB} is determined based on the pressure acting on the base with an area S_B.

The pressure on this area is $p_B - p_\infty$.

$$\therefore \qquad C_{AB} = \frac{(p_B - p_\infty) \cdot S_B}{q \cdot S_{mid}} \tag{27}$$

or $C_{AB} = -C_{pB} \cdot \bar{S}_B$ where $\bar{S}_B = \frac{S_B}{S_{mid}}$ and C_{pB} is the base pressure coefficient. The base pressure depends on whether or not the engine is operating and on the exit area of the rocket nozzle.

When the engine is burning and for Mach numbers greater than 1:

$$C_{pB} = \left(\frac{1 - Ae}{S_{\mathrm{Re}\,f}} \right) \cdot \frac{.25}{M} \tag{28}$$

For Mach numbers less than 1:

$$C_{pB} = \left(\frac{1 - Ae}{S_{\mathrm{Re}\,f}} \right) \cdot \left(.12 + .13 \cdot M^2 \right) \tag{29}$$

When the engine is not operating, the exit area is considered to be equal to zero resulting in base pressure values for Mach greater than 1:

$$C_{pB} = \left(\frac{1}{S_{\mathrm{Re}\,f}} \right) \cdot \frac{.25}{M} \tag{30}$$

And for Mach less than 1:

$$C_{pB} = \left(\frac{1}{S_{\mathrm{Re}\,f}} \right) \cdot \left(.12 + .13 \cdot M^2 \right) \tag{31}$$

3.1.5 Axial due to friction

The elemental frictional force acting on an area of body surface S_w is given by:

$$dA_f = \tau_0 \cdot dS \cdot \cos(\delta) = \tau_0 \cdot r \cdot \cos(\delta) \cdot ds \cdot d\phi \tag{32}$$

As before $dx = ds \cdot \cos(\delta)$ and:

$$A_f = \int_{S_w} \tau_0 \cdot \cos(\delta) \cdot dS = 2 \cdot \int_0^l r \cdot dx \cdot \int_0^\pi \tau_0 \cdot d\phi \tag{33}$$

The local skin friction coefficient is defined as: $c_{fx} = \frac{\tau_0}{q}$, the calculation of which will be given in section 3.1.3. The frictional axial force coefficient may then be given by:

$$C_{Ar} = \frac{A_f}{q \cdot S_{mid}} = \frac{2}{S_{mid}} \cdot \int_{S_w} c_{fx} \cdot \cos(\delta) \cdot dS = \frac{4 \cdot \lambda_l}{\pi} \cdot \int_0^1 \bar{r} \cdot d\tilde{x} \cdot \int_0^\pi c_{fx} \cdot d\phi \tag{34}$$

At zero angle of attack (symmetric flow):

$$C_{Af} = 4 \cdot \lambda_l \cdot \int_0^1 c_{fx} \cdot \bar{r} \cdot d\tilde{x} \tag{35}$$

3.1.3 Skin Friction Coefficient

3.1.3.1 Incompressible flow

The skin friction coefficient may be determined, for incompressible flow by the following semi-empirical method:

First the frictional stress at the wall, for laminar flow, is given by:

$$\tau_w = \mu \left(\frac{\partial V_x}{\partial y} \right)_{y=0} \tag{36}$$

For turbulent flow:

$$\tau_w = 0.0225 \cdot \rho_\delta \cdot V_\delta^2 \left(\frac{\nu}{V_\delta \cdot \delta} \right)^{1/4} \tag{37}$$

Consequently:

$$\tau_w = B \cdot \rho_\delta \cdot V_\delta^2 \cdot \mathrm{Re}_x^{\,n} \tag{38}$$

Re_x is the local Reynolds number and is given by, $\mathrm{Re}_x = V_\delta \cdot x / \nu$.

Thus the local incompressible skin friction coefficient is:

$$c_{finc} = \frac{\tau_w}{q} = C \cdot \mathrm{Re}_x^{\,n} \tag{39}$$

Where $q = \rho_\delta \cdot \frac{V_\delta^2}{2}$.

The average value of the local skin friction coefficient is given by:

$$c_f = \int_0^1 c_{finc} \cdot d\bar{x} = D \cdot \mathrm{Re}^n \tag{40}$$

The values for A,B,C,D, m and n are given in the following table from [2]:

Table 1: Boundary layer constants

	A	B	C	D	m	n
Laminar	4.65	0.332	0.664	1.32	0.5	-0.5
Turbulent	0.37	0.0289	0.0578	.0.72	0.8	-0.2

The value for the turbulent skin friction coefficient is reliable as long as the Reynolds number does not exceed approximately 10^6. For values from about, $10^6 \le \mathrm{Re} \le 10^9$ the following equation gives better results, according to [13]:

$$c_{tinc} = 0.427 \cdot (\log(\mathrm{Re}) - 0.407)^{-2.64} \tag{41}$$

3.1.3.2 Effects of changes in the atmosphere

The Reynolds number determines the influence of the atmosphere on the skin friction coefficient. The speed of sound and the kinematic viscosity vary with altitude according to the standard atmosphere tables [9]. The value of the Reynolds number may be determined, in relation to the Sea Level value by use of [9] and by the following relationships:

$$\mathrm{Re} = \mathrm{Re}_{SL} \cdot f_1 \cdot (H) \cdot f_2 \cdot (H) \cdot f_3 \cdot (H) \tag{42}$$

Where $\mathrm{Re}_{SL} = M_\infty \cdot a_{SL}\, l / \nu_{SL}$, $f_1 \cdot (H) = \rho / \rho_{SL}$, $f_2 \cdot (H) = a / a_{SL}$, $f_3 \cdot (H) = \mu_{SL} / \mu$.

3.1.3.3 Effects of compressibility

In order to take into account the effects of compressibility, based on experimental data, the relation proposed in [14] gives, for turbulent flow:

$$c_{ft} = \frac{0.0637 \cdot \exp(-0.172 \cdot M_\infty)}{\mathrm{Re}^{.182}} \tag{43}$$

The laminar value may be determined, in relation to the incompressible value, by:

$$c_{fl} = c_{flinc} \cdot \left(1 + 0.03 \cdot M_\infty^2\right)^{-1/3} \tag{44}$$

Again, these relationships are approximate and based on the flow over a flat plate. Sufficient accuracy may be achieved, however, by considering smaller and smaller values of *L*, plate length, until a convergent value within acceptable error limits is found.

High temperatures influence the skin friction coefficient as well. Considering the fact that, very close to the surface of the body, the velocity is very small, the gas in this region is similar in character to an incompressible gas. If it is assumed that this portion of the boundary layer has a basic influence on the friction and heat transfer in the boundary layer, the compressible equations may be modified to account for heat transfer. It has been shown empirically, according to [2], that the use of effective parameters, such as effective temperature, enthalpy, density and viscosity, give satisfactory results in calculating the effects of heat transfer on the boundary layer parameters, e.g. the skin friction coefficient.

By first calculating the potential, frictionless, rotationless flow over the body, the flow parameters at the external edge of the boundary layer may be determined. Then, by calculating the skin temperature, by use of the energy equation, the effective temperature may be determined by the following:

$$T^* = T_e \cdot \left(1 + 0.032 \cdot M_e{}^2 + 0.58 \cdot \left(\frac{T_w}{T_e} - 1 \right) \right) \tag{45}$$

Where T_e and M_e are the temperature and Mach number at the external edge of the boundary layer, and T_w is the temperature at the wall. Once the effective temperature, T^*, is known, the effective viscosity and density may be used to calculate the effective Reynolds number, $\mathrm{Re}^* = \frac{\rho^* \cdot V_e \cdot x}{\mu^*}$ which in turn may be used to modify the compressible skin friction coefficient to include the effects of high temperature boundary layers, as follows:

$$c_{flam} = \frac{0.664}{\sqrt{\mathrm{Re}_x^*}} \tag{46}$$

$$c_{fturb} = \frac{0.0592}{\sqrt{\mathrm{Re}_x^*}} \tag{47}$$

Where c_{flam} is the laminar coefficient and c_{fturb} is the turbulent coefficient.

3.1.3.4 Empirical Equations English Units

The body wave and friction axial components can be simplified for English units:

$$Cawave = \left(1.59 - \frac{1.83}{M^2}\right) \cdot \left(\tan^{-1} \cdot \left(\frac{.5}{\ln/d}\right)\right)^{1.69} \tag{48}$$

$$Cafriction = .053 \cdot \left(\frac{lb}{d}\right) \cdot \left(\frac{M}{q \cdot lb}\right)^{0.2} \tag{49}$$

The fin contributions are:

$$Cdfinwave = nf \cdot \left(0.0133 \cdot \left(\frac{M}{q \cdot cmac}\right)^{0.2} \cdot \left(\frac{2 \cdot Sw}{Sref}\right)\right) \tag{50}$$

$$Cdfin\_friction = nf \cdot \left(\frac{2}{\gamma \cdot M_{Ale}{}^2}\right) \cdot \left(\left(\frac{\left((\gamma+1) \cdot M_{Ale}{}^2\right)}{2}\right)^{\gamma/\gamma+1} \cdot \left(\frac{\gamma+1}{2 \cdot \gamma \cdot M_{Ale}{}^2 - (\gamma-1)}\right)^{1/\gamma-1} - 1\right) \tag{51}$$

The forms of equations 48-51 were given in [18]. Where $\ln$ is the length of the nose and lb is the length of the body and d is the diameter of the body component, nf is the number of fins in a finest, $cmac$ is the mean aerodynamic chord, M_{Ale} is the Mach number perpendicular to the

leading edge of the fins. These equations were broken down for each component to find the individual drag components of the body.

3.2 Original Configuration Stage separation

The largest dart currently in use is known to separate from the booster at motor burnout [8]. The friction drag on the booster will be greater than on the dart due to the larger surface area and larger fins. The wave drag on the dart fins and the booster fins are expected to be of similar magnitude. If these assumptions are true, than the only other contributing drag components come from the nose cone, the conical transition on the booster and the base drag, see figure 2, once the motor has burned out. The booster wave drag should be similar to the nose drag, although it is expected to be larger with the 2-inch dart. The component of greatest interest here therefore is base drag, and it is desired to illustrate the effect that this value has on total drag when the engine is not in operation. For the 2-inch dart, it is assumed that all drag components on the booster are either equal to or greater than the dart drag components. The nose wave drag and booster base drag coefficients are shown below versus Mach number in order to illustrate the magnitude of base drag contribution to separation.

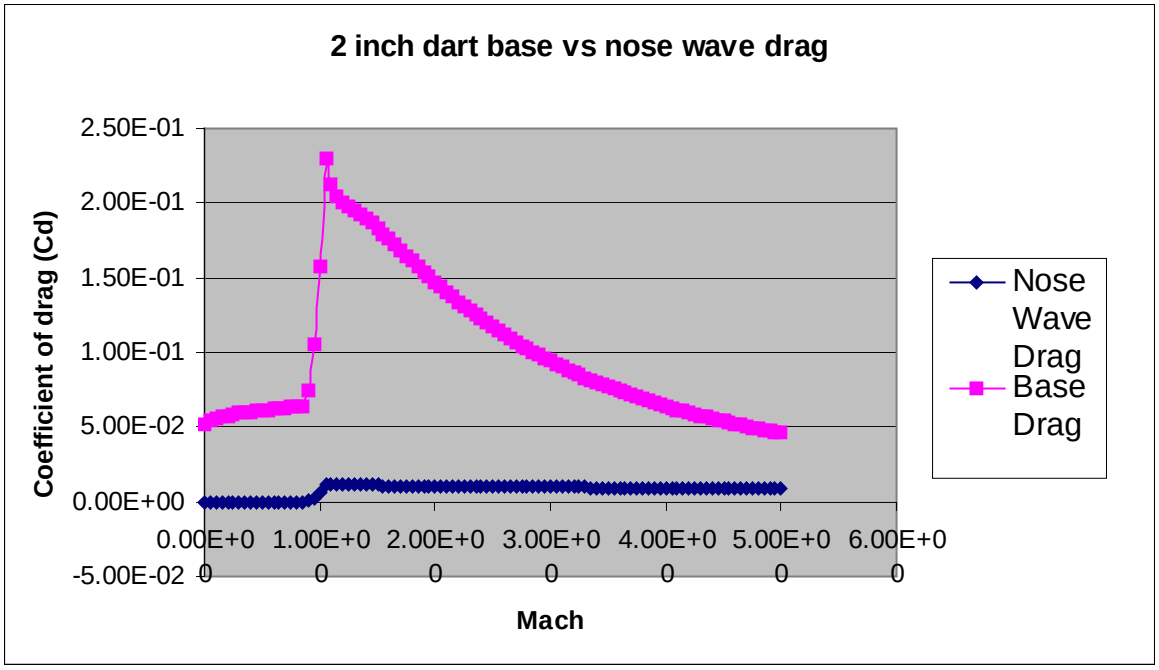

Figure 11: Dart nose wave drag vs. Booster base drag

These values were calculated using equations 28-31, 48,49,50,51. Since the 2-inch dart configuration burns out at just under Mach 5, the drag differential between these drag contributors is at a minimum. If the rocket were to remain in one piece, if the drag on the dart was larger than that on the booster, as the rocket slows, the base drag would increase much more rapidly than the nose wave drag. This condition would lead to eventual separation at some critical Mach number, assuming all other drag contributors meet the criteria given above. It is expected therefore that the 4-inch dart should separate from the booster, at a Mach number corresponding to the booster drag being larger than that on the dart. The question is when does this separation occur.

3.3 Modified Dart Stage separation

The drag calculated via equations 28-31, 48,49,50,51 for the components on the 4-inch dart are given in the following chart:

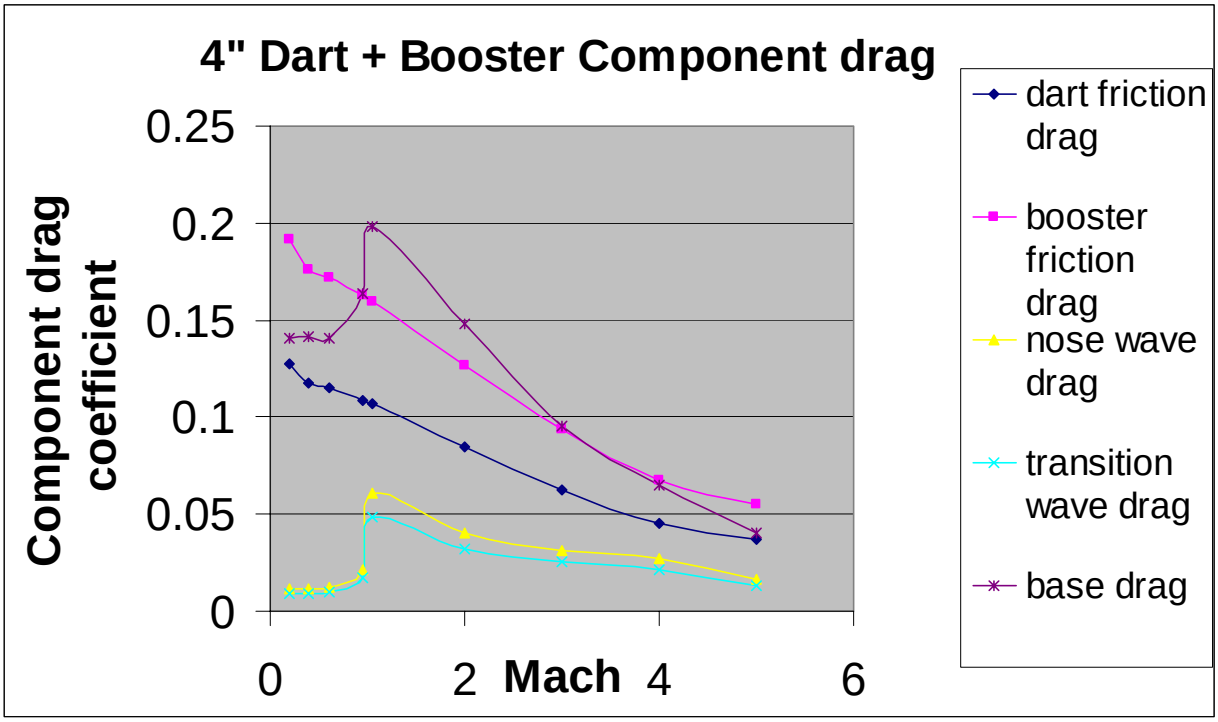

Figure 12: Component Drag on the 4-inch booster

The transition referred to in the graph in figure 12 above is the converging-diverging conical interface between the upper and lower stages. The drag on the diverging section, the section on the booster, is that shown in the graph above.

The base drag and the booster friction drag are the greatest individual components of drag once the motor has burned out. In the case of the smaller diameter darts, the drag differential is even greater.

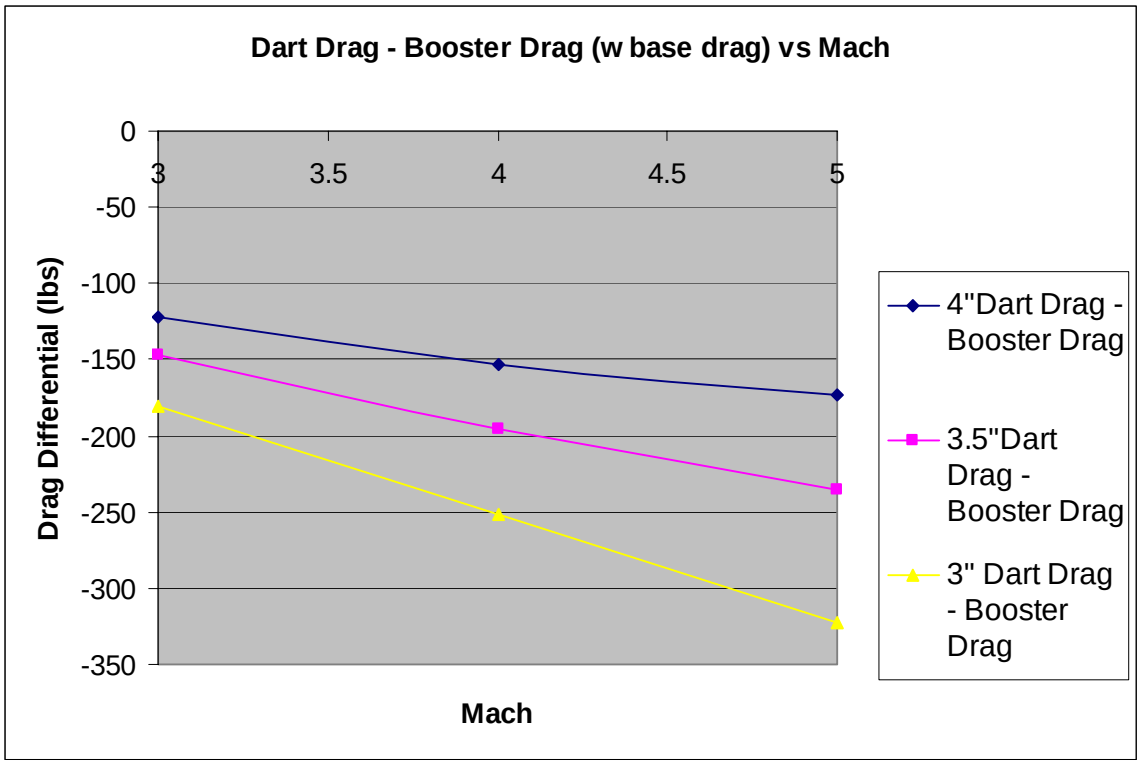

Figure 13: Drag differential between dart and booster

It can thus be concluded that even the 4-inch dart should separate, quite rapidly from the booster due to these effects. As the rocket slows due to drag and gravity, this differential will increase, due to the rapid increase in base drag with a decrease in speed, and given the lower maximum speed of the larger darts, makes this separation even more likely.

CHAPTER 4: STABILITY

4.1 Solution Development

The stability of an aircraft depends on the center of pressure being aft, of the center of gravity. The response of the aircraft to a change in angle of attack, due to a wind gust for example, must be such that an opposing moment causes a return towards the original angle of attack. The flow regime, subsonic, transonic and supersonic dictates the methods used to determine the stability parameters.

The normal force coefficient depends on pressure and shear forces in the same way that the axial force coefficient does. It can therefore be represented by the sum of two coefficients corresponding to these forces:

$C_N = C_{Np} + C_{Nf}$, where C_{Np} is the component resulting from pressure and C_{Nf} that resulting from friction.

4.1.1 Normal due to pressure

The elemental value of the normal force due to pressure is:

$$dN_p = -(p - p_\infty) \cdot r \cdot d\phi \cdot ds \cdot \cos(\phi) \cdot \cos(\delta) \tag{52}$$

Due to the symmetry of the pressure distribution:

$$N_p = -2 \cdot \int_0^l r \cdot dx \cdot \int_0^\pi (p - p_\infty) \cdot \cos(\phi) \cdot d\phi \tag{53}$$

The symmetry of the pressure distribution is based on the fact that even at angles of attack, there will be a plane of symmetry that bisects the body from the stagnation upwind side through the stagnation streamlines to the downstream side, thus establishing a plane of symmetry

The normal force coefficient due to pressure is therefore:

$$C_{Np} = \frac{N_p}{q \cdot S_{mid}} = -\frac{4 \cdot \lambda_l}{\pi} \cdot \int_0^1 \bar{r} \cdot d\tilde{x} \cdot \int_0^{\pi} C_p \cdot \cos(\phi) \cdot d\phi \tag{54}$$

At zero angle of attack C_{Np} equals zero. The flow at this angle is considered to be completely symmetric about the axis of the body and therefore produces no net normal force.

4.1.2 Normal force due to friction

The normal force due to friction can be determined if the shear force τ_0 is known over the whole body. The elementary value of the normal shear force acting on an elemental area of skin surface is:

$$dN_f = \tau_0 \cdot \sin(\delta) \cdot \cos(\delta) \cdot dS = \tau_0 \cdot r \cdot \sin(\delta) \cdot \cos(\phi) \cdot ds \cdot d\phi \tag{55}$$

Integrating to give the total normal force due to friction:

$$N_f = \int_{S_w} \tau_0 \cdot \sin(\delta) \cdot \cos(\phi) \cdot dS = 2 \cdot \int_0^l r \cdot \tan(\delta) \cdot dx \cdot \int_0^{\pi} \tau_0 \cdot \cos(\phi) \cdot d\phi \tag{56}$$

The normal force coefficient due to friction is therefore:

$$C_{Nf} = \frac{1}{q \cdot S_{mid}} \cdot \int_{S_w} \tau_0 \cdot \sin(\delta) \cdot \cos(\phi) \cdot dS = \frac{4 \cdot \lambda_l}{\pi} \cdot \int_0^1 \bar{r} \cdot \tan(\delta) \cdot d\tilde{x} \cdot \int_0^{\pi} c_{fx} \cdot \cos(\phi) \cdot d\phi \tag{57}$$

At zero angle of attack, as with the normal force coefficient due to pressure, C_{Nf} is equal to zero.

The distribution of this force can be obtained by differentiating this force with respect to the axial coordinate x.

$$\frac{dN_p}{dx} = -2 \cdot \int_0^{\pi} (p - p_\infty) \cdot r \cdot \cos(\phi) \cdot d\phi \tag{58}$$

The distribution of the normal force due to pressure coefficient is found in the same way.

$$\frac{dC_{Np}}{d\tilde{x}} = -\frac{4 \cdot \bar{r} \cdot \lambda_l}{\pi} \cdot \int_0^{\pi} C_p \cdot \cos(\phi) \cdot d\phi \tag{59}$$

This distribution provides the knowledge of the pitching moment about the center of gravity. Simple equations have been developed for typical body components, the nose, cylindrical afterbody and boat tail normal forces as a function of angle of attack are given respectively as:

For Ogival noses, derived from [21]:

$$Cnon = -.0071 \cdot M^2 - .0071 \cdot M + 2.415 \tag{60}$$

For the cylindrical portion also from [21]:

$$Cncyl = 2 \cdot \alpha + Cd_{\alpha - 90^o} \cdot \frac{A_p}{S_{ref}} \cdot \alpha^2 \tag{61}$$

For the boat tail, from [10]:

$$\frac{u}{U} = 1 - \varepsilon^2 \cdot \frac{\sec h^{-1}(T)}{\sqrt{1 - T^2}} + \varepsilon^4 \cdot \left(\frac{\sec h^{-1}(T)}{\sqrt{1 - T^2}} \right) +$$

$$\frac{M^2 \cdot \varepsilon^4}{1 - T^2} \cdot \left[-\left(\sec h^{-1}(T)\right)^2 + \frac{10 + T^2}{4} \cdot \frac{\sec h^{-1}(T)}{\sqrt{1 - T^2}} - (N - 1) \cdot T^2 \left(\frac{\sec h^{-1}(T)}{\sqrt{1 - T^2}} \right) \right] \tag{62}$$

$$T = \varepsilon \cdot \beta \tag{63}$$

$$N = \frac{\gamma + 1}{2} \cdot \frac{M^2}{\beta^2} \tag{60}$$

$$\frac{v}{V} = \varepsilon \cdot \left(1 - \varepsilon^2 \cdot \frac{\sec h^{-1}(T)}{\sqrt{1 - T^2}} \right) \tag{64}$$

$$Cpbt = \frac{2}{\gamma \cdot M^2} \cdot \left(\left(1 + \frac{\gamma - 1}{2} \cdot M^2 \left[1 - (1 + \phi_x)^2 - \phi_r{}^2 \right] \right)^{\frac{\gamma}{\gamma - 1}} - 1 \right) \tag{65}$$

$$\frac{u}{U} = 1 + \phi_x \tag{66}$$

$$\frac{v}{U} = \phi_r$$

Where u is the x (axial) and v is the transverse component of velocity. Given the pressure coefficient for the boat tail $Cpbt$, and taking the center of pressure of the boat tail as 1/3 the distance from the forward face, the coefficient of moment can be determined as from equation (58). The potential flow derivatives ϕ_x , ϕ_r were determined as in [10].

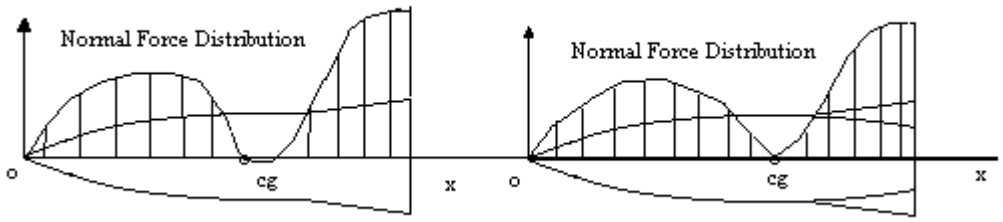

Figure 14: Slope of Normal force coefficient distribution

The body itself creates a positive pitching moment about the center of gravity. The addition of tail fins, or a flared section shown in figure 14 balances this with a negative pitching moment, thus stabilizing the vehicle. The forces on the fins may be determined with simplified equations

just as those on the body. Certain geometrical relationships must be considered for the fins however.

The theory of linearized supersonic flow, as given in [2], states that flow downstream of any given point is affected by that point within a cone expanding at the Mach angle:

$$\mu = \sin^{-1}(1/M) \tag{67}$$

Consequently the corners of a three dimensional fin create regions of pressure that are different than a two dimensional airfoil. Additionally, if these regions overlap, an additive effect results.

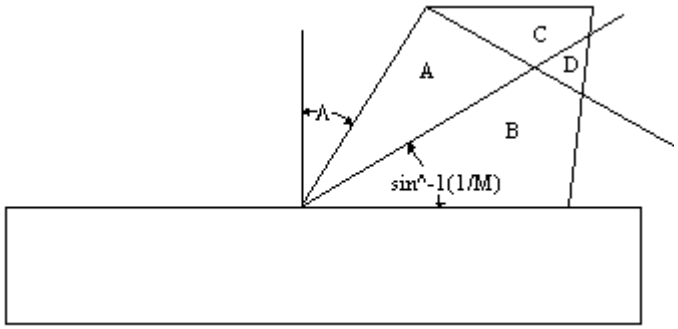

Figure 15: Pressure regions on a fin in super sonic flow

The theoretical coefficients of pressure, based on thin airfoil theory, are given, from [21]:

$$C_{P(a)} = \frac{4 \cdot \alpha}{\sqrt{\beta^2 - k^2}}$$

$$C_{P(b)} = \frac{8 \cdot \alpha}{\pi \cdot \sqrt{\beta^2 - k^2}} \cdot \left[\frac{\pi}{2} - \sin^{-1} \sqrt{\frac{n^2 - t_1^2}{1 - t_1^2}} \right] \tag{68}$$

$$C_{P(c)} = \frac{4 \cdot \alpha \cdot (a+1)}{\pi \cdot \beta \cdot \sqrt{a}} \cdot \tan^{-1} \sqrt{\frac{-2 \cdot a \cdot \beta \cdot y_2}{(1+a) \cdot (x_2 + \beta \cdot y_2)}} \tag{68}$$

$$C_{p(D)} = C_{P(B)} + C_{P(c)} + C_{P(A)}$$

The effect of the body in presence of the fin and the fins in presence of the body are given by the carry over factors:

$$Kwb = 0.3973 * dspan^2 + 0.567 * dspan + 1.0125 \tag{69}$$

$$Kbw = 0.6786 * dspan^2 + 1.3257 * dspan + 2E - 15 \tag{70}$$

These values were determined by curve fitting empirical data given in [21].

The points at which the fin forces act are assumed to be at the ¼ chord point for subsonic flow and at the ½ chord point in super sonic flow, which is typical practice in preliminary design. Given this information the pitching moment may be determined. The cp location in transonic flow will be discussed in section 4.5.

4.2 Pitching Moment

The pitching, as well as the lateral moment coefficient due to symmetry, like the coefficients already described, also depends on pressure and shear forces. $C_m = C_{mp} + C_{mf}$ Where C_{mp} is the component due to pressure and C_{mf} is that due to friction.

4.2.1 Moment due to pressure

For the component due to pressure, considered at the nose:

$$dM_p = -x \cdot (p - p_\infty) \cdot r \cdot \cos(\delta) \cdot \cos(\phi) \cdot ds \cdot d\phi - r^2 \cdot (p - p_\infty) \cdot \sin(\delta) \cdot \cos(\phi) \cdot ds \cdot d\phi \tag{71}$$

Integrating gives:

$$M_P = -2 \cdot \int_0^l x \cdot r \cdot dx \cdot \int_0^\pi (p - p_\infty) \cdot \cos(\phi) \cdot d\phi - 2 \cdot \int_0^l r^2 \cdot \tan(\delta) \cdot dx \cdot \int_0^\pi (p - p_\infty) \cdot \cos(\phi) \cdot d\phi \qquad (72)$$

and therefore:

$$\begin{aligned} C_{mp} &= \frac{M_p}{q \cdot S_{mid} \cdot l} = -\frac{4 \cdot \lambda_l}{\pi} \cdot \int_0^1 \bar{r} \cdot \tilde{x} \cdot d\tilde{x} \cdot \int_0^\pi C_p \cdot \cos(\phi) \cdot d\phi \\ &\quad -\frac{1}{\pi} \cdot \int_0^1 \bar{r}^2 \cdot \tan(\delta) \cdot d\tilde{x} \cdot \int_0^\pi C_p \cos(\phi) \cdot d\phi \end{aligned} \qquad (73)$$

If the body is slender, the second terms in the two equations above may be neglected.

4.2.2 Moment due to friction

The elemental force due to friction is:

$$\begin{aligned} dM_f &= x \cdot \tau_0 \cdot \sin(\delta) \cdot \cos(\phi) \cdot dS - r \cdot \tau_0 \cos(\delta) \cdot \cos(\phi) \cdot dS \\ &= x \cdot \tau_0 \cdot r \cdot \sin(\delta) \cdot \cos(\phi) \cdot ds \cdot d\phi - r^2 \cdot \tau_0 \cdot \cos(\delta) \cdot \cos(\phi) \cdot ds \cdot d\phi \end{aligned} \qquad (74)$$

Integrating:

$$\begin{aligned} M_f &= \int_{S_w} x \cdot \tau_0 \cdot \sin(\delta) \cdot \cos(\phi) \cdot dS - \int_{S_w} r \cdot \tau_0 \cdot \cos(\delta) \cdot \cos(\phi) \cdot dS \\ &= 2 \cdot \int_0^l x \cdot r \cdot \tan(\delta) \cdot dx \cdot \int_0^\pi \tau_0 \cdot \cos(\phi) \cdot d\phi - 2 \cdot \int_0^l r^2 \cdot dx \cdot \int_0^\pi \tau_0 \cdot \cos(\phi) \cdot d\phi \end{aligned} \qquad (75)$$

The component of the moment coefficient due to friction:

$$C_{mf} = \frac{4 \cdot \lambda_l}{\pi} \cdot \int_0^1 \tilde{x} \cdot \bar{r} \cdot \tan(\delta) \cdot d\tilde{x} \cdot \int_0^\pi c_{fx} \cdot \cos(\phi) \cdot d\phi - \frac{2}{\pi} \cdot \int_0^1 \bar{r}^2 \cdot d\tilde{x} \cdot \int_0^\pi c_{fx} \cdot \cos(\phi) \cdot d\phi \qquad (76)$$

4.3 Center of pressure

The position of the center of pressure is the position along the axis of symmetry at which the normal force is concentrated. This is determined by: $x_{cp} = \frac{M}{N}$, where M is the pitching moment and N is the normal force.

The center of pressure in relation to the length of the body is given by:

$$\bar{x}_{cp} = \frac{x_{cp}}{l} = \frac{C_m}{C_N} = \frac{C_{mp} + C_{mf}}{C_{Np} + C_{Nf}} \tag{77}$$

For small angles of attack, C_{mf} and C_{Nf} can be neglected. Substituting equations 54 and 73 for C_{Np} and C_{mp}:

$$\bar{x}_{cp} = \frac{\int_0^1 \bar{r} \cdot \tilde{x} \cdot d\tilde{x} \cdot \int_0^\pi C_p \cdot \cos(\phi) \cdot d\phi + \frac{1}{4 \cdot \lambda_l} \cdot \int_0^1 \bar{r}^2 \cdot \tan(\delta) \cdot d\tilde{x} \cdot \int_0^\pi C_p \cdot \cos(\phi) \cdot d\phi}{\int_0^1 \bar{r} \cdot d\tilde{x} \cdot \int_0^\pi C_p \cdot \cos(\phi) \cdot d\phi} \tag{78}$$

In the case of the analysis at hand, the Barrowman equations for the pressure distribution were used for the subsonic flow regime. For the super-sonic regime, non-rotational potential flow was assumed. The method of [10] was used to determine supersonic flow parameters, e.g. C_p, T_e, M_e, ρ_e etc., the subscript ∞ denoting, in this case, the values just outside the boundary layer. Hypersonic flight was not considered in this case as the maximum velocity of the original Super Loki configuration just reaches the lower end of this regime, and the larger dart will have more drag resulting in an even lower maximum speed.

4.4 Subsonic Flow

As mentioned the well-known Barrowman equations allow a simple and accurate way to determine the location of the center of pressure and normal force coefficient on a sub sonic rocket. Above M=0.3 and below M =1 the Karman-Tsien correction is used to correct for compressibility.

4.4.1 Change with angle of attack

The Barrowman equations may be extended to include angles of attack as follows:

$$C_{psub} = C_{P0} + \sin(\alpha) * (Cla - C_{P0}) \quad (79)$$

Where C_{P0} is the zero angle of attack of the whole rocket, as calculated by the Barrowman method, $Clab$ is the centroid of the projection of the rocket.

4.4.2 Pitching moment coefficient

The pitching moment is:

$$C_m = C_N \cdot (C_g - C_{psub}) \cdot \sin(\alpha) \quad (80)$$

There is a limit to the angle of attack to an airfoil and therefore its contribution to the stability of an aircraft. This is the stall angle and is a function of the aspect ratio $AR = span\text{^}2 / A_F$ where, $span$ includes the height of the fins, as well as the diameter of the body at the fin location. A_F is the area of the fins, and should include the rectangular area of a longitudinal cross section of the rocket having dimensions equal to the base length of the fins, and a width

equal to the diameter of the rocket at the fin location. The equation for the stall angle is therefore:

$$\alpha_{stallfins} = 15 \cdot \left(1 + \frac{1.1}{AR}\right) \tag{81}$$

When the pitching moment becomes zero, the center of pressure coincides with the center of gravity, and the maximum stable angle of attack has been reached. Any further increase in the absolute value of the angle of attack will move the center of pressure forward and destabilize the rocket. This angle cannot exceed the stall angle of the fins of course as the flow over the upper side of the fins would separate reducing their effectiveness, resulting in lower lift and therefore a reduction of the restoring moment.

4.5 Transonic Flow

From [2], the change in the location of the center of pressure on the body alone in transonic flow, for small angles of attack, does not change a significant amount from the subsonic value. From $M_\infty = 0.4$ to $M_\infty = 1.3$ the center of pressure moves about 2% of the length of the body. The body of revolution in this case having a parabolic nose and cylindrical afterbody. This is a typical geometry and does not significantly vary from an ogive nose-afterbody combination. Similarly, the center of pressure on a cone with a semi apex angle of 15º, moves only 4% towards the tip. Above Mach .3 he Karman-Tsien, rule may be used to modify the subsonic calculation of the center of pressure with the following equation:

$$C_{Ptrans} = \frac{C_{Psub}}{\sqrt{1 - M_\infty{}^2} + \left(M_\infty{}^2 \cdot \left\{1 + \left[(\gamma - 1)/2\right] M_\infty{}^2\right\} / 2 \cdot \sqrt{1 - M_\infty{}^2}\right) \cdot C_{Psub}} \tag{82}$$

Given the fact that the center of pressure does not move significantly for bodies of revolution, at small angles of attack, and that the relation given in (71) gives results of second order accuracy, the Karman-Tsien relation will be used and a 5% safety margin included when considering transonic stability. This 5% margin is based on the typical change in center of pressure location for the conical nose section, the worst-case scenario for the nose sections being considered, based on the experimental data sighted in [2]. Additional data was given in [17] for the Karman-Tsien method, among others, and shows that this method provides reasonable results given the restraints mentioned above. The movement of the center of pressure on the fins is more significant and generally moves from the ¼ chord point to the ½ chord point for airfoils. For three-dimensional airfoils, the movement of the center of pressure through the transonic range, the following relationship may be used:

$$\frac{cp}{c} = \frac{\beta \cdot AR - 2/3}{2 \cdot \beta \cdot AR - 1} \tag{83}$$

The assumption of small angles of attack is considered to be a valid one, as the design of the rocket is intended to be such that a high damping moment exists to limit the angle of attack to levels within the valid range of this analysis.

4.6 Supersonic Flow

In supersonic flow, unlike subsonic flow, the fluid properties at one point cannot affect the properties upstream of that point. A cone of influence bounded by the Mach angle, $\mu = \sin^{-1}(1/M)$ exists downstream of a given point. This situation results in a hyperbolic

equation for the equations of motion. This equation is often linearized in order to provide a first order of accuracy solution.

For preliminary design, this level of accuracy may be adequate, but a relatively simple second order solution for symmetrical, irrotational supersonic flow about a pointed body of revolution has been presented in [10]. This is a simplification from any real rocket due to the bluntness of the tip of the nose, and the rotation induced by the curved, detached shock wave that influences the flow downstream. Compared to the Method of Characteristics, which is considered to be the exact solution (within the accuracy of the computer solving the equations), the second order solution of [10] provides results that are within 1% of the this solution, as long as the angle of the tangent to the surface of the body is not greater than 0.94 of the conical variable

$$t = \frac{\beta \cdot r}{x} \tag{84}$$

Where $\beta = \sqrt{M^2 - 1}$. This method does not include the effects of viscosity or high temperatures. It is therefore a solution of the velocity potential given by:

$$\phi_{rr} + \frac{\phi_r}{r} - \beta^2 \cdot \phi_{xx} = M^2 \left(2 \cdot (N-1) \cdot \beta^2 \phi_x \phi_{xx} + 2 \cdot \phi_r \phi_{xr} + \phi_r \cdot \phi_{rr} + other\_cubic\_terms\right) \tag{85}$$

Where $N = \frac{\gamma + 1}{2} \cdot \frac{M^2}{\beta^2}$.

The equations of velocity, as developed in [10], at the surface of the body were given above and are repeated below.

$$\frac{u}{U} = 1 + \phi_x \tag{86}$$

$$\frac{v}{U} = \phi_r \tag{87}$$

Where u is the x (axial) and v is the transverse component of velocity. Using these components, the Coefficient of Pressure may be calculated with the following:

$$Cp = \frac{2}{\gamma \cdot M^2} \cdot \left(\left(1 + \frac{\gamma - 1}{2} \cdot M^2 \left[1 - \left(1 + \phi_x \right)^2 - {\phi_r}^2 \right] \right)^{\frac{\gamma}{\gamma - 1}} - 1 \right) \tag{88}$$

The method used in [10] to solve the potential equation involved using a line of sources. A linear source exists at the tip of the body followed by quadratic sources downstream along the axis of the body. Boundary conditions are applied so that the velocity at the surface of the body is of zero magnitude. The step-by-step method for the development of the linearization of equation (73) and that of the second order components of this equation is given in [10]. This method is very straightforward and has been reproduced in the sot.m MATLAB code in the appendix.

4.6.1 Angle of Attack in super sonic flow

An actual rocket in flight is rarely, if ever, perfectly aligned with the flow. Disturbing factors, such as wind gusts, thrust malalignment etc. cause the rocket to oscillate about the tangent of the trajectory of the mass center of the rocket. To determine how the rocket responds to a change in angle of attack: the zero angle of attack solution of the potential equation can be modified by considering a perturbation potential to this solution. According to [2], this perturbation potential, given the subscript 2, may be related to the zero angle of attack potential, subscript 1, as follows:

$$\varphi_2 = -\frac{1}{r}\left[r \cdot \left(\frac{\partial \varphi_1}{\partial r} \right) \right] \cdot \cos(\phi) \tag{89}$$

It can be seen from this equation that the perturbation potential due to angle of attack may be determined from the unperturbed potential and the geometry of the body. The primary parameter of importance being the calculation of the coefficient of pressure:

$$C_{p2} = -2 \cdot \left[-\overline{\varphi}_{1RX} \cos(\phi) + \frac{\alpha^2}{2} \cdot \left(4 \cdot \sin^2(\phi) - 1 \right) \right] \tag{90}$$

This equation does not however produce very accurate results, as it neglects the effects of friction and flow separation, and is not used in favor of the empirical equations for the nose and cylinder normal force provided above (equations 60 and 61).

4.7 Original Configuration Stability

The pressure distribution on the Original Super Loki dart configuration is shown in figure 16 below. Figure 17 shows a qualitative comparison for ogival and conical noses with that computed here.

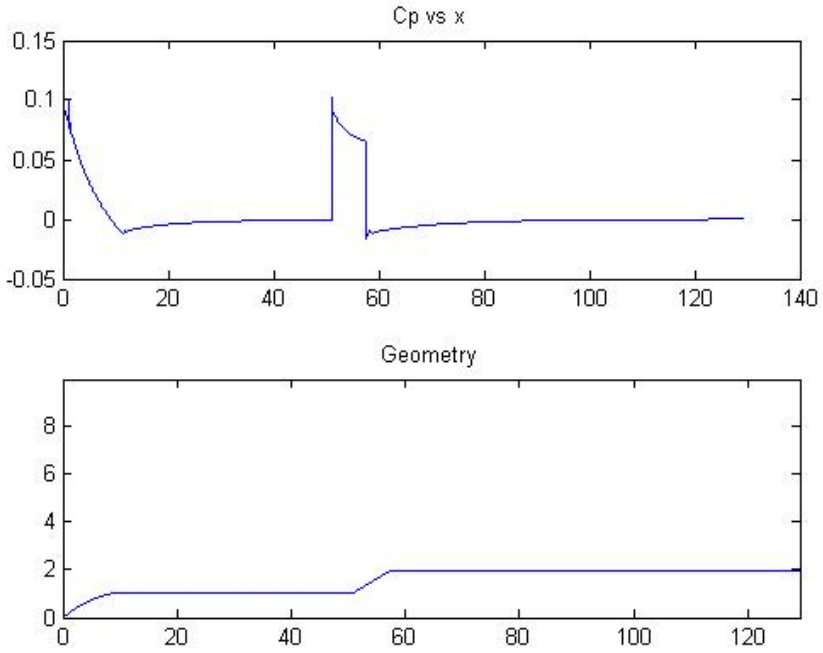

Figure 16: Pressure Distribution on original Super Loki (2 inch dart)

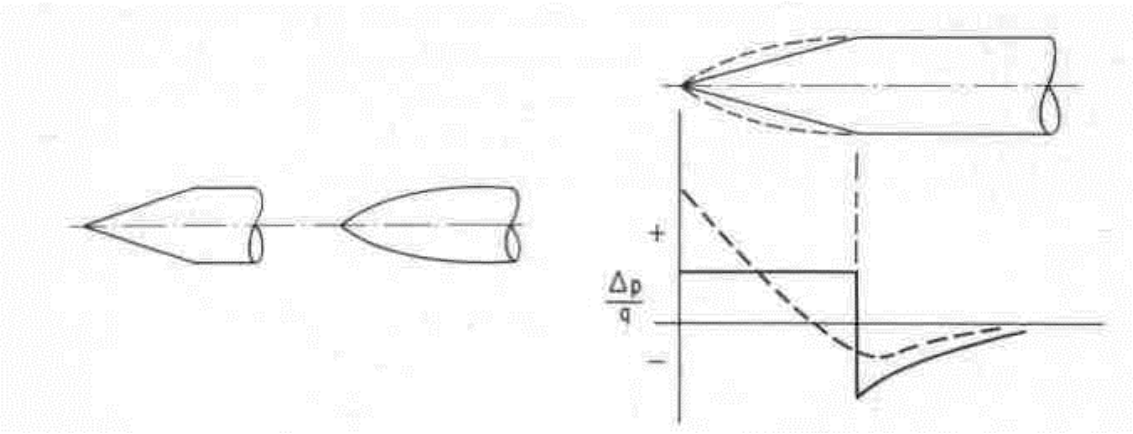

Figure 17: typical pressure distribution on conical and ogival nose/cylinder configurations

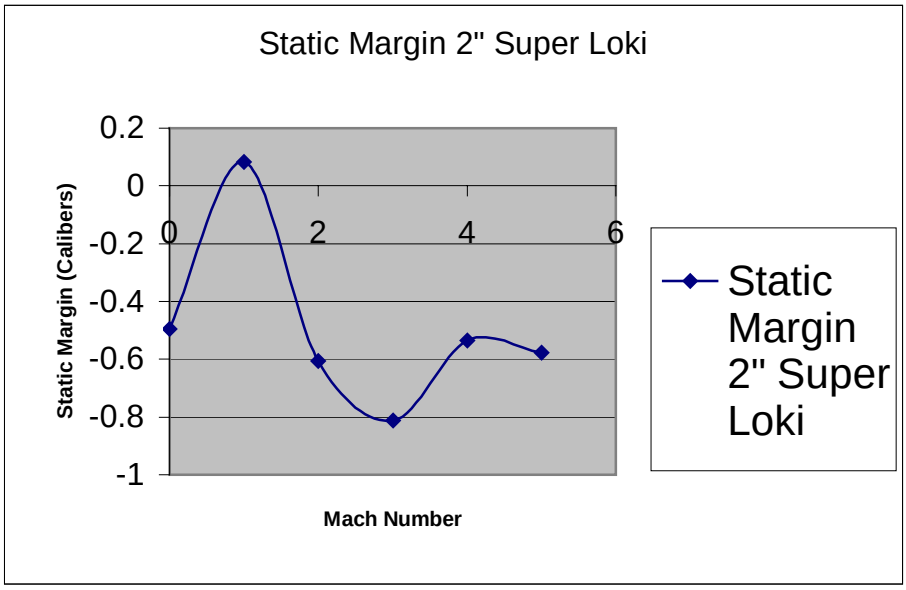

Figure 18: Original Configuration Static Margin

The figure above shows the calculated static margin of the original 2-inch diameter dart. The static margin is given in calibers (maximum diameters, in this case four inches for the booster). A negative static margin means that the center of pressure is behind the center of gravity and the rocket is statically stable. As can be seen above, the static margin becomes positive around Mach 1. It should be pointed out that these values are based on a theoretical calculation of the mass loss of propellant and therefore center of gravity movement. This result may lie, therefore within the margin of error of this calculation. It is difficult to say without having more detailed information about the booster performance. Additionally, as the Super Loki booster surplus is aging, the original performance may not be reached. This having been said, this analysis does not take into account the effects of spin imparted by the launcher. This was done purposely as a measure of safety in the design of the modified dart. It is believed therefore that the static margin indicated above, even though producing a non-stable condition, may be overcome by

the angular momentum imparted by the launcher. Being unable to fully quantify the performance of the aging motors and therefore the actual angular momentum of the modified dart-booster configuration, it is felt that a considerable safety margin is attained by designing the modified dart without including spin. In other words, the modified dart will be stable as long as the static margin remains negative. Further optimization may be attained in the future, however, once flight measurements of stability can be made.

4.8 Modified Configuration Stability

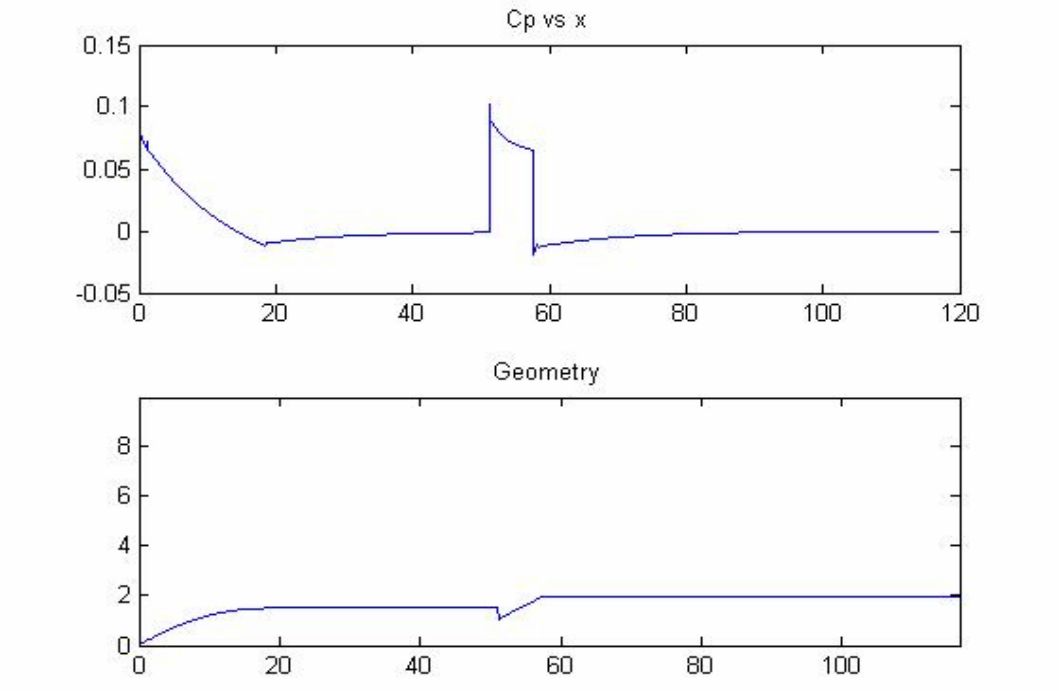

Figure 19: 3.5 inch dart pressure distribution and geometry

The pressure distribution for the body alone, given the three varying diameters, were computed. An example of the program output is shown above, for the 3.5-inch dart. Integration of these values allowed for the selection of the fin size required for stable flight.

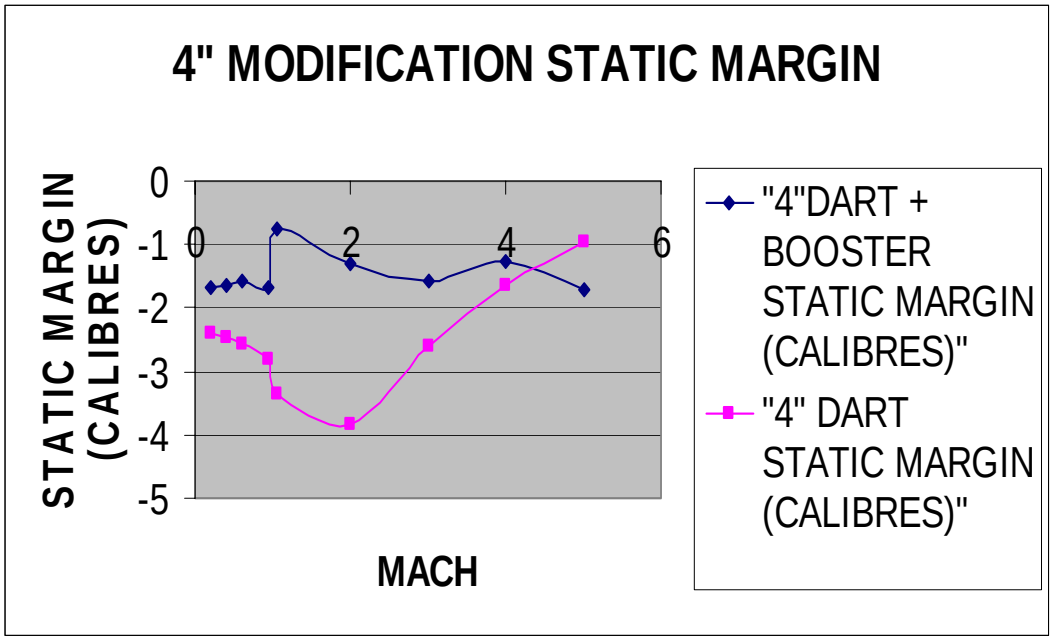

Figure 20: 4 Static margin for the 4-inch dart

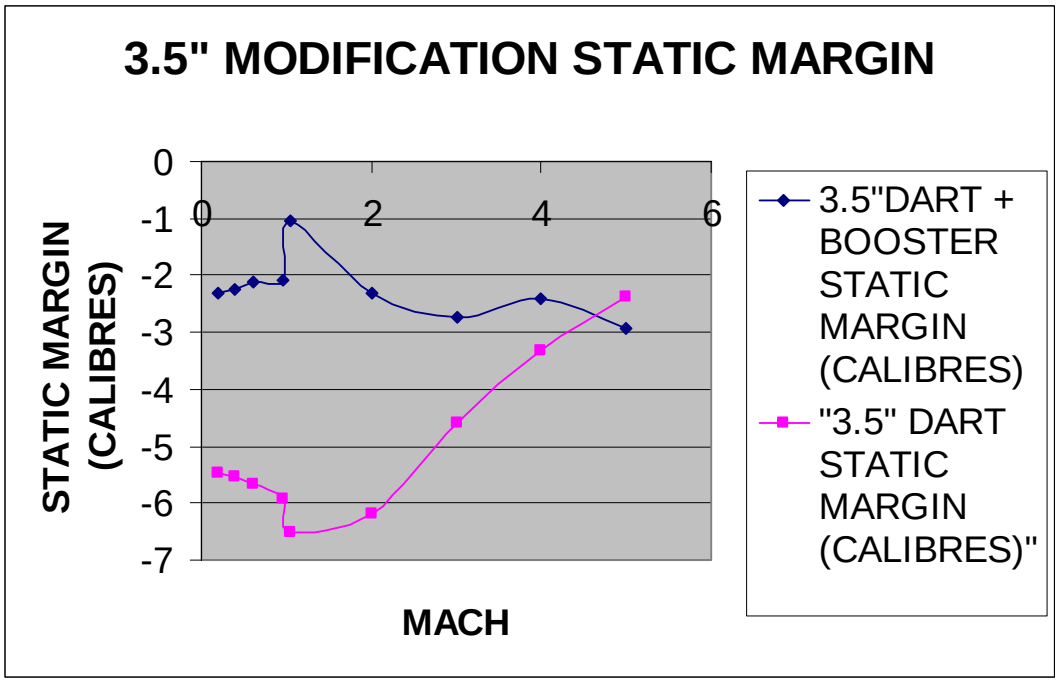

Figure 21: Static margin for the 3.5 inch dart

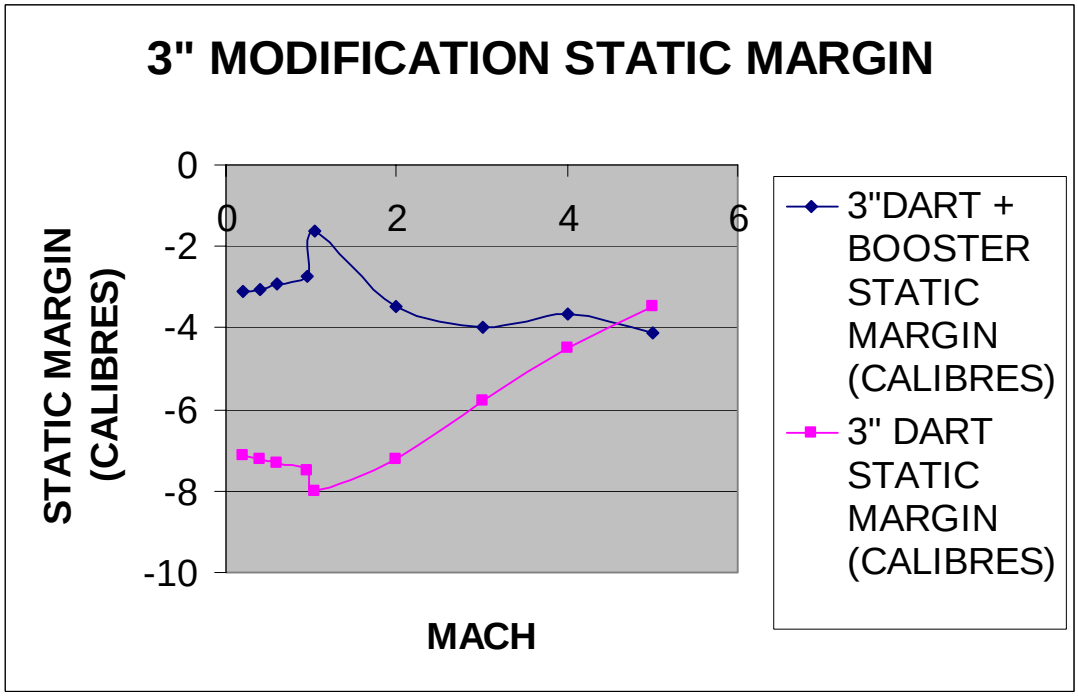

Figure 22: Static margin for the 3 inch dart

From the graphs above, it can be seen that the static margin is always negative for the configurations chosen. The static margin is defined as the distance between the center of pressure and the center of gravity. In this analysis, a negative static margin corresponds to a stable, or center of pressure, aft of the center of gravity. The units are in calibers or maximum diameters. The fin geometries considered were more dictated by the mechanical loading due to the launcher than by aerodynamic considerations. Ideally the smaller the fins the better from the aerodynamic sense. The smaller the dart fins, the farther aft the cp will move, thus allowing less mass to be used in the nose cone to move the cg forward. The fins must be able to withstand the mechanical loads however with a reasonable safety margin, and are therefore larger than if this condition need not be met. However from a stability standpoint the fins selected provide a stable platform for all reasonable angles of attack and Mach numbers considered reasonable.

Another way to assess static stability is the moment coefficient slope. Given a positive change in angle of attack, from a gust of wind for example, a negative moment should occur. This moment should push the rocket back toward the angle of attack that existed before the gust. As long as the change in moment is negative with increasing positive angle of attack, the rocket will be stable.

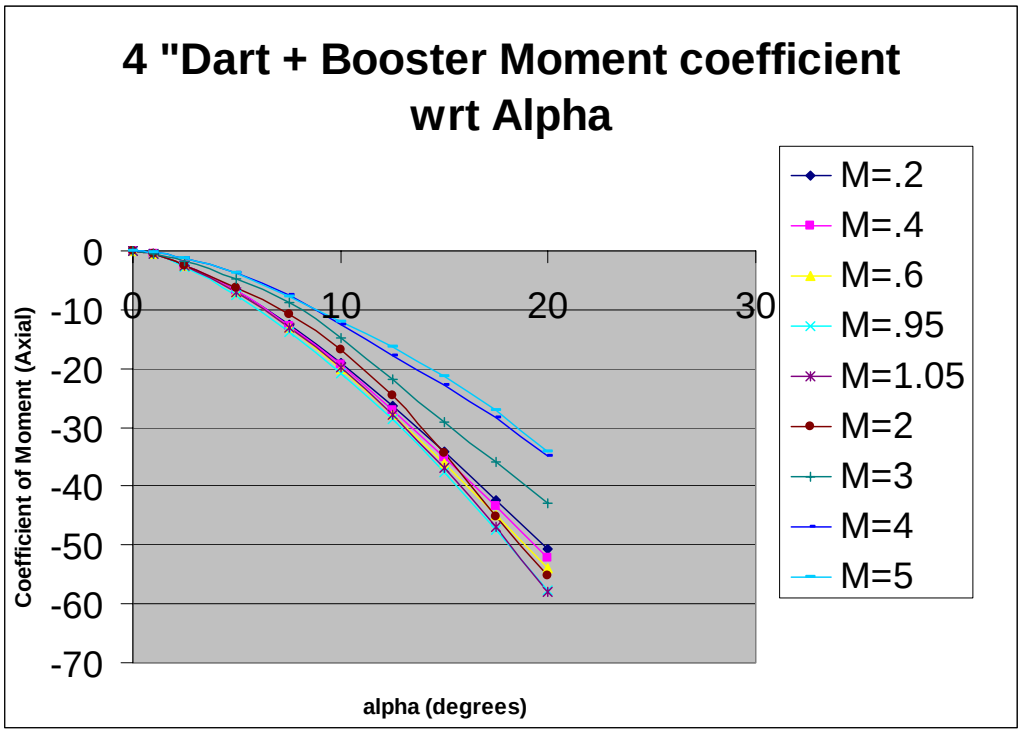

Figure 23: Coefficient of Moment for the 4-inch dart-booster combination

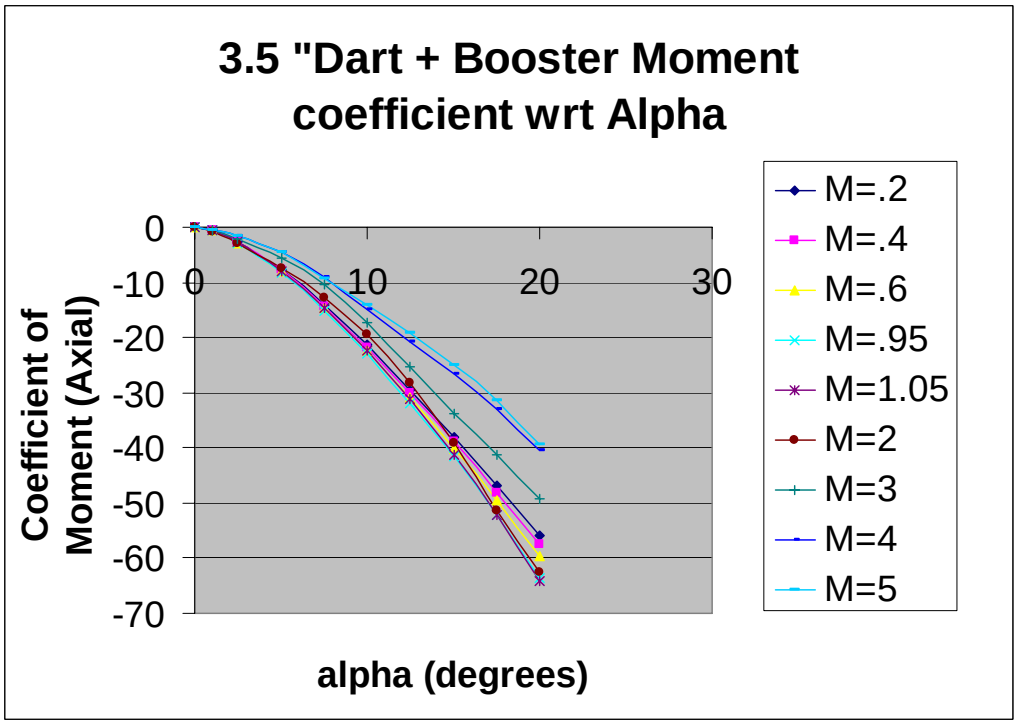

Figure 24: Coefficient of Moment for the 3.5-inch dart-booster combination

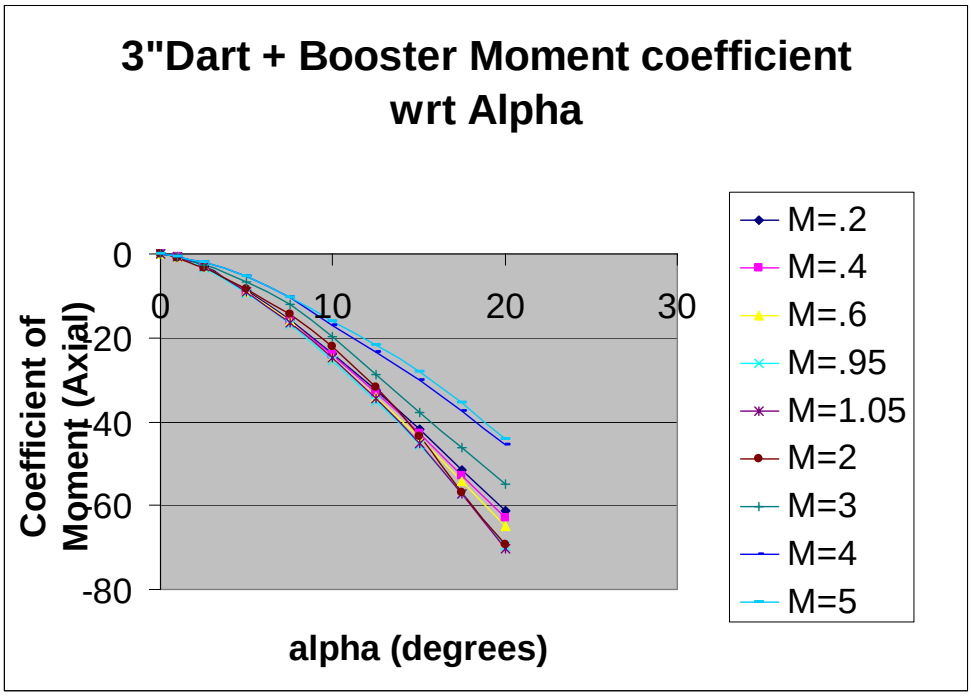

Figure 25: Coefficient of Moment for the 3-inch dart-booster combination

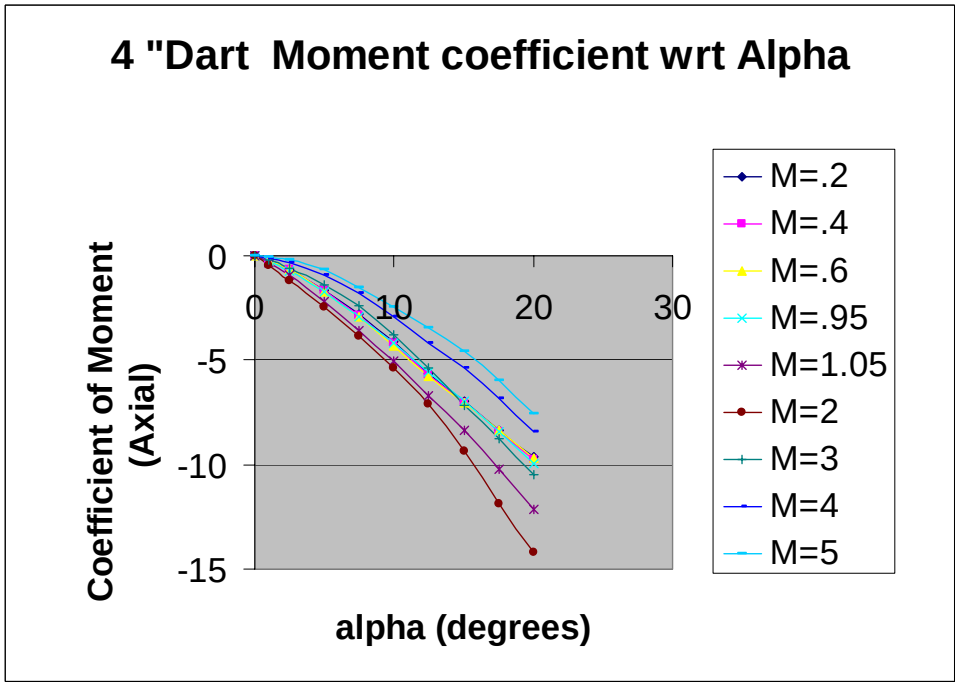

Figure 26:Coefficient of Moment for the 4-inch dart alone

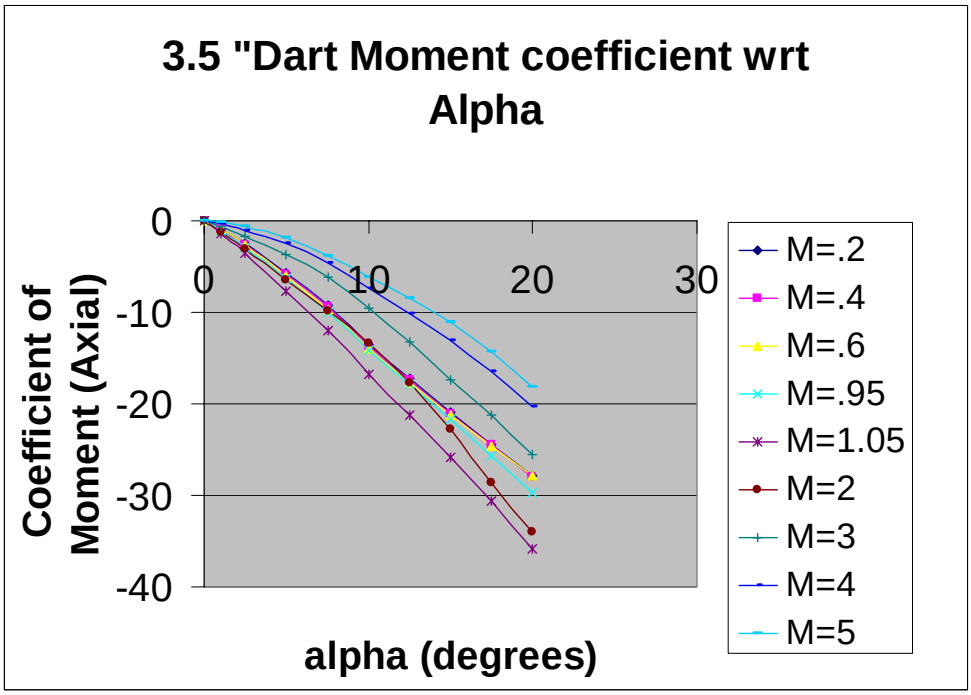

Figure 27: Coefficient of Moment for the 3.5-inch dart alone

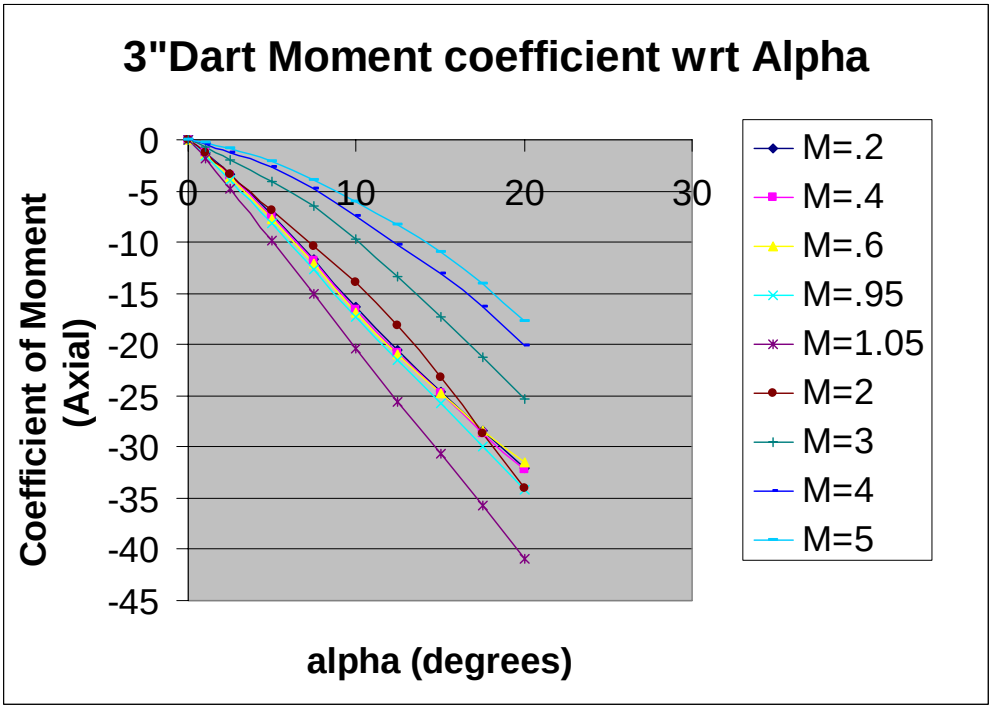

Figure 28: Coefficient of Moment for the 3-inch dart alone

The graphs above show that all configurations are stable as long as the center of gravity is far enough forward. This can be achieved in all configurations based on simple force balance equations. In order to design some flexibility into the rocket, a solid nosecone made of standard steel with a 3-2 inch diameter 12-inch long cutout used to insert lead ballast was used to locate the cg in the correct position for each configuration. The result is a necessarily large increase in total weight of the rocket and especially the dart. This large increase in weight and diameter results in a major impact on the max speed and altitude of the rocket. This is necessary however in order to position the center of gravity ahead of the center of pressure.

CHAPTER 5: TRAJECTORY ALTITUDE AND VELOCITY

5.1 Solution Development

The main goal, as stated, of this analysis is to develop a larger diameter upper stage for the Super Loki rocket. A reduction in apogee and increase in drag are inevitable due to the increase in cross sectional, nose semi angle and weight of the modified dart. These changes in performance are not considered significant, however, as the apogee altitude of the rocket is no longer one of the primary design parameters, as it was with the original configuration.

The purpose of the new design is to allow more versatile payloads to be carried to high altitudes, not necessarily to a specific altitude. The purpose of calculating the trajectory of the modified rocket is necessary, however, in order to estimate the payload deployment altitude. A good idea of the altitude of the rocket at apogee will be necessary to facilitate apogee acquisition with ground-based antennas.

Therefore, based on the method outlined in [21], an iterative method for calculating the velocity, this altitude may be estimated. The method used in this analysis differs slightly from that presented in [21] in that the thrust of the rocket is first used without drag to determine mass loss and acceleration. The velocity thus determined allows the calculation of the dynamic pressure and therefore the drag. This drag results in a reduction in the acceleration from thrust and a corresponding reduction in velocity. From the velocity the distance traveled may be determined. These equations are repeated from launch to apogee, for each time step. Multiple methods exist, one is given in [21], another method is outlined in [15].

5.2 Original Configuration

As a check on this method, the original configuration was considered and the results are indicated below:

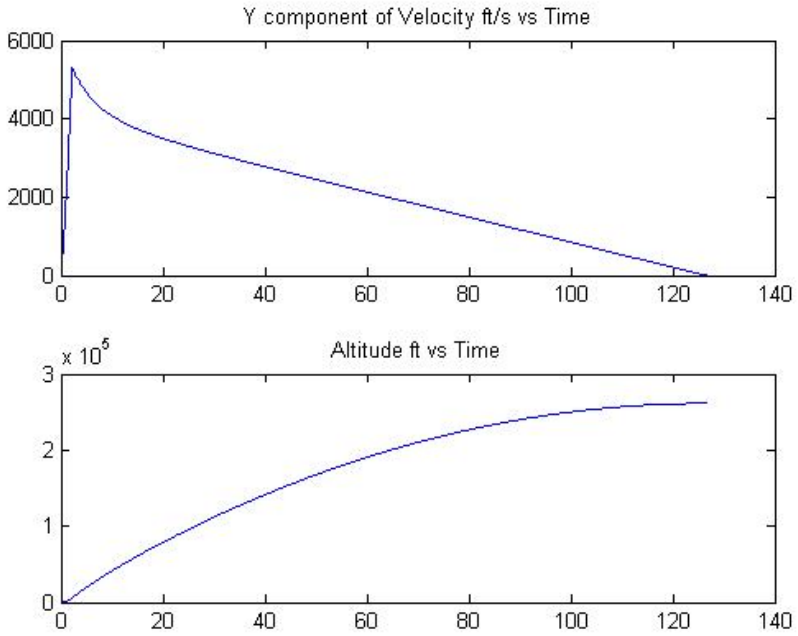

Figure 29: Original Configuration Calculated

Altitude and Velocity (260000ft and 5345ft/s or Mach 4.88)

The results are within 4% for max altitude and less than 1% for maximum Mach of the data given in [3].

5.3 Modified Configuration Trajectory (Altitude and Velocity)

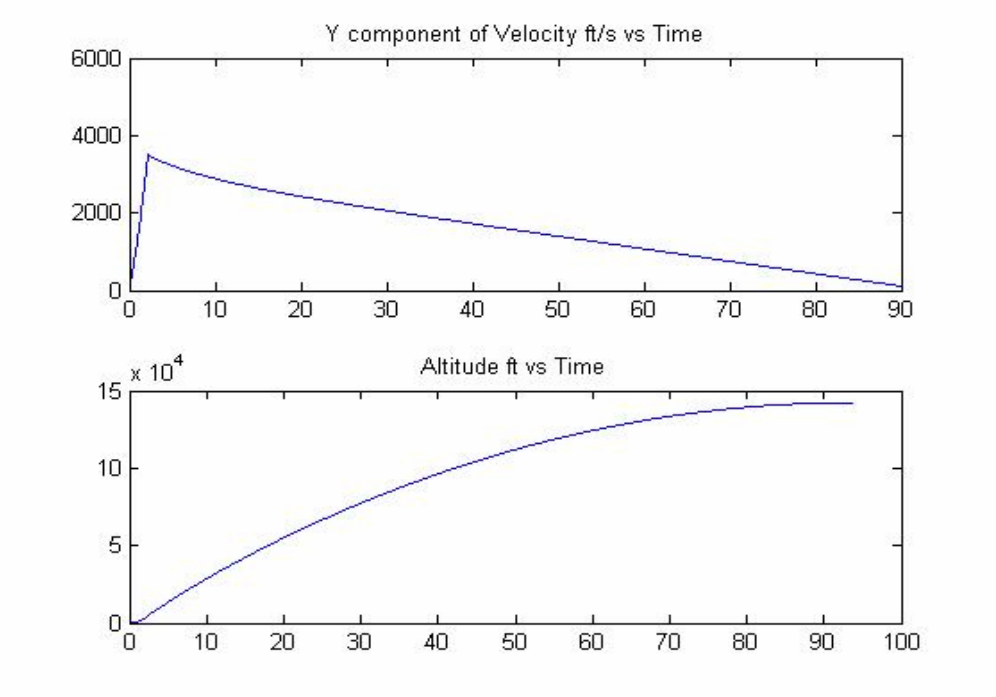

Figure 30: 4 inch dart Configuration Calculated

Altitude and Velocity (148000ft and 5345ft/s or Mach 3.1922)

Results from the 4-inch dart altitude and velocity calculation are shown in figure 30 above. As can be seen a large decrease in altitude and speed occur as a result of increased mass and drag. As mentioned earlier, however, this modification is designed to incorporate larger and possibly multiple payloads that may not necessarily need to reach the previous design altitude of the Super Loki. This reduction in performance is therefore deemed acceptable, and, as shown below, the loss in performance is not so severe for the 3-inch dart. Therefore a whole family of darts may be considered and the appropriate configuration chosen based on mission parameters.

Table 2: Altitudes achieved by various diameter darts

	4"	3.5"	3"
Max Altitude	148000	180000	232610

CHAPTER 6: MECHANICAL LOADING

6.1 Solution Development

Three primary events cause the largest loads on the rocket. These are the maximum thrust, and therefore maximum velocity and drag, the rocket/launcher interaction and the force of the ejection charge on the body.

6.1.1Thrust and Drag

The formulas for the drag coefficients on the nose are given by equations (48-49). The thrust is assumed to be the average thrust plus 20% to simulate a peak thrust. This choice is somewhat arbitrary in that no information is available on the thrust curve of the Super Loki, but has been based on typical curves of rocket performance. The peak thrust often occurs at the beginning of a solid rocket burn, but this thrust will be used at the maximum velocity, which usually occurs at the end of the burn. The average thrust is given by [3] as 17885N (4021 lbs). With the addition of 20% this thrust is 21462 N 4825 lb. The drag at maximum velocity is calculated by the MATLAB code, based on the method of [21]. This force is applied axially on the rocket body.

The axial force on the body tube is given by:

$$\sigma_x = \frac{Thrust + Drag}{\pi \cdot (r_o - r_i)^2} \tag{91}$$

Where r_o is the outer radius and r_i is the inner radius.

6.1.2 Torsion

The torsional force applied by the launcher on the rocket may be determined by the impulse momentum theorem. The impulse momentum equation, for angular momentum is:

$$\int_0^t M \cdot dt = I \cdot \Delta\omega \tag{92}$$

Which states that a moment applied over a given time is equal to the moment of inertia of a mass times the change in angular momentum over the given time period. In the case of the Super Loki, the rocket axis is aligned with the launcher axis. The moment applied by the launcher is aligned with the longitudinal axis of the rocket, which coincides with the moment of inertia aligned with the x, or longitudinal, axis I_x. The time difference is given in [3] for the Super Loki and assumed to be the same for the modified dart as a worst case. The moment of inertia is determined in IDEAS-9 in this analysis and is equal to $2019 \times 10^5 lb \cdot in^2$ for the 4-inch dart. The change in angular velocity is taken from the time difference and the known geometry of the launcher. The Super Loki launcher rotates the rocket by approximately180 degrees during the launch. If the amount of time it takes for the rocket to leave the launcher is known, the angular velocity is:

$$\omega = \frac{180}{time} \tag{93}$$

Since the angular velocity at the initial time of zero seconds is zero, the change in angular velocity may be taken as the final angular velocity.

The force on the fins as they traverse the launcher may then be determined by finding the moment imparted, divided by the radius of force application, i.e. the tips of the fins. The strain on the fins may then be determined to ensure that they do not deform plastically. The moment transferred to the rocket on the fins is considered to be equal to the moment applied to the body tube. The maximum torque applied to the tube due to this moment is given by:

$$\tau_{\max} = \frac{r_o}{r_i}\tau_{\min} \tag{94}$$

Where $\tau_{\min} = \frac{M_x \cdot r_i}{J}$ and $J = \frac{\pi}{2} \cdot \left(r_o^4 - r_i^4\right)$.

The maximum torque occurs at the outer radius. The principal stresses must be found on the body tube by use of the Mohr's circle. The center of this circle is given by:

$$C_{mohr} = \frac{\sigma_x + \sigma_y}{2} \tag{95}$$

σ_x is the stress due to the axial forces and σ_y is due to bending forces, which are considered equal to zero in this case. The Pythagorean theorem gives the radius of the Mohr's circle, where the center of the circle and the shear stress, due to torsion in this case, are the legs of the triangle:

$$R_{mohr} = \sqrt{\tau^2 + C^2{}_{mohr}} \tag{96}$$

The principal stresses are then:

$$\sigma_1 = C_{mohr} + R_{mohr} \tag{97}$$

$$\sigma_2 = C_{mohr} - R_{mohr} \tag{98}$$

Fin stress

$$\sigma_{fin} = \frac{6 \cdot F \cdot l}{B \cdot H^2} \tag{99}$$

Where the fin was treated as a flat plate

Results for these calculations will be given in section 6.3.

6.1.3 Ejection Charge

The ejection charge puts an outward pressure force on the cylindrical tube. It must be assured that the ejection charge is capable of ejecting the nose cone without rupturing the body cylinder. This is especially true if the body tube is made of composite material. If it is assumed that the pressure acting normal to the interior face of the nose cone, is constant throughout the internal volume, the stress acting on a longitudinal cross section of the cylindrical portion is given by:

$$\tau_n = \frac{P \cdot r}{t} \tag{100}$$

The maximum hoop stress is given by equation (100) above where t is the wall thickness, r is the inner radius, and P is the internal pressure.

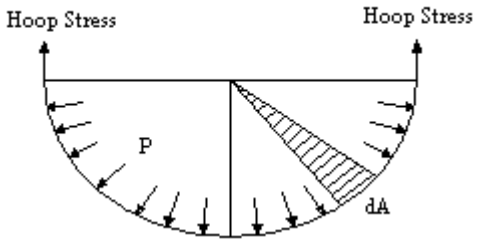

Figure 31: Pressure Distribution resulting in hoop stress

The composite tube was treated as a pressure vessel with an internal pressure applied to the inner surface. In order to deploy the payload, the friction between the payload, including the nose cone, and the composite tube must be overcome.

6.2 Original Configuration Mechanical Loading

The original configuration included two shear screws that secured the nose cone to the payload tube. Based on accelerometer data from [16] the force required to overcome the force necessary to shear these screws and to overcome friction of the original payload was no less than 414 psi.

$$F_{ejection} = 250 \cdot g \cdot 18.3lb \tag{101}$$

$$F_{ejection} = 4575lbf$$

$$A_{nose} = 11.04in^2$$

$$P_{no\,sec\,one} = \frac{F_{ejection}}{A_{nose}} \tag{102}$$

$$P_{no\,sec\,one} = 414.402\,psi$$

18.3lb being the weight of the entire dart assembly. To provide a large safety factor, a pressure of 1000psi was applied to the interior of the composite tube, the results are given in section 6.3.

6.3 Modified Configuration Mechanical Loading

Some preliminary hand calculations were performed to get an idea of the forces on the body during flight. Using equations 91,94,97,98,99 resulted in reasonable values given the assumed flight loads and material properties.

Compression and Torsion Load on the composite tube, from equations 97 and 98:

$$\sigma_1 = 2.15 \times 10^4 \, psi$$

Fin Loads, due to launcher moment equation (99) repeated here:

$$\sigma_{fins} = \frac{6 \cdot F_1 \cdot l}{B \cdot H_{th}^2} \tag{103}$$

$$\sigma_{fins} = 1.92 \times 10^4 \, psi$$

Hoop Stress due to ejection forces equation (100) repeated here:

$$\tau_n = \frac{P \cdot r_i}{t_{th}} \tag{104}$$

$$\tau_n = 1.5 \times 10^4 \, psi$$

The assumptions used in calculating the loads are that the axial moment of inertia of the entire rocket is represented to a reasonably accurate degree, by that calculated by the IDEAS-9 software package. It is also assumed that the rocket reaches the same angular velocity as that seen by the unmodified configuration. The forces on the fins are therefore based on these assumptions. The IDEAS-9 model therefore consisted of one half of the fin assembly with one fin, with the thrust force of 4825 lbs distributed on the aft surface and a 25 lb force applied to the upper edge of the fin:

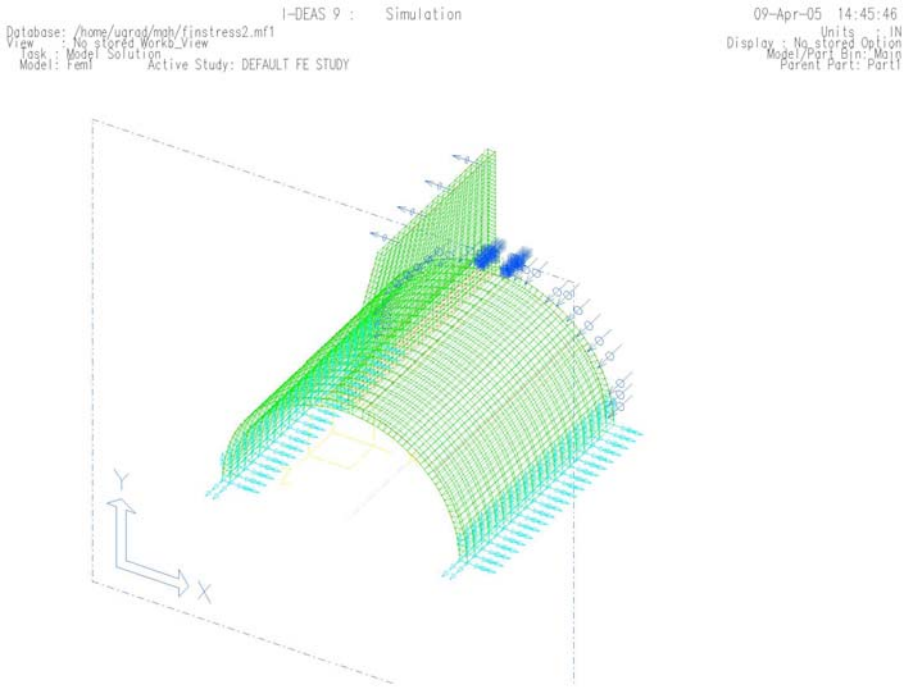

Figure 32: Steel fin can mesh (green) with forces (dark blue) and constraints (light blue)

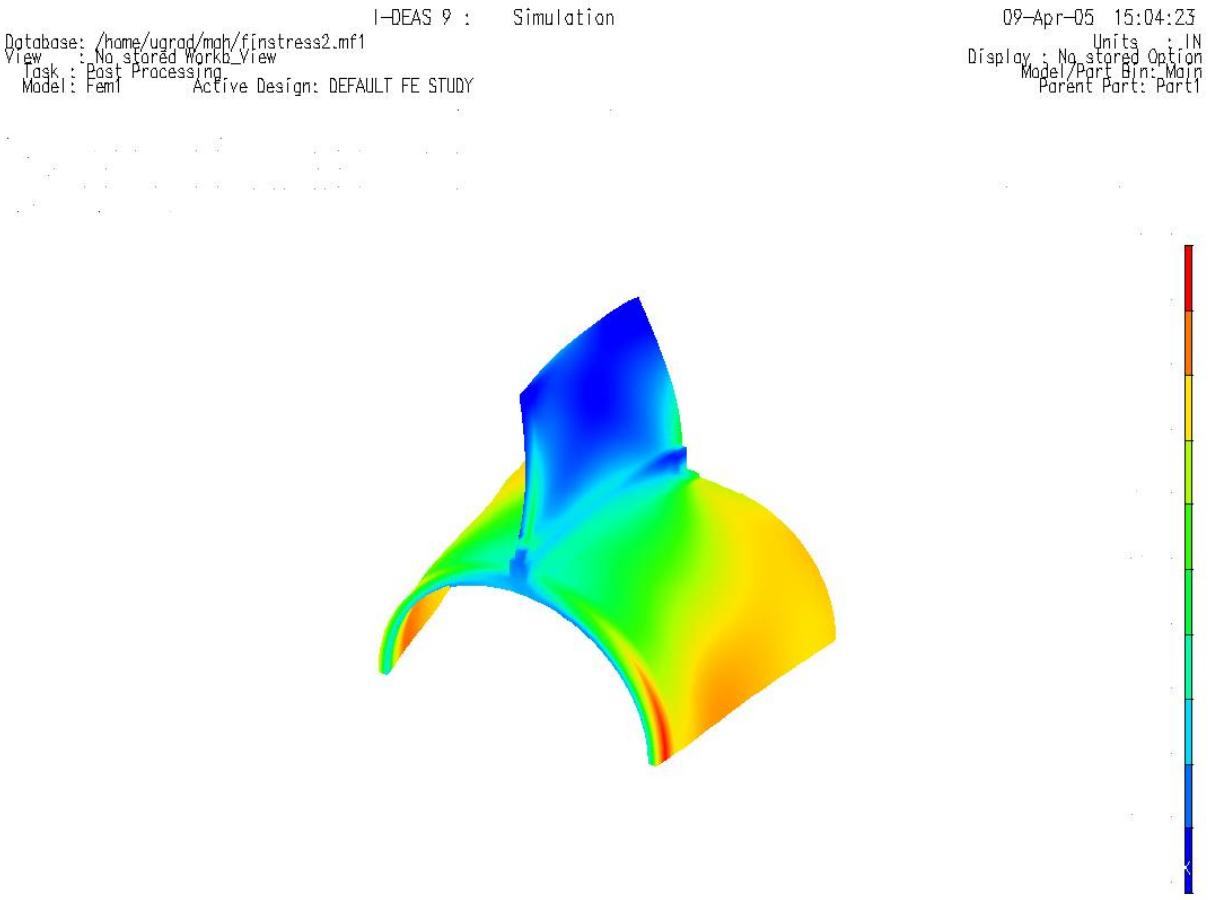

Figure 33: Stress and strain of fin can

The resulting stresses and deflections are given by ideas in the figure above as:

$\sigma_{VMMax} = 16000\,psi$ and the max displacement is $\varepsilon_{\max} = 0.00328 inches$

This value is less than the 19,200psi value determined by hand calculations, which will be considered in the material selection.

These values are well within the capabilities of standard steel and are considered conservative.

The body tube itself was assigned the properties of a generic carbon fiber material taken from [7]. The forces applied were a combination of the torsion force applied at the aft end and the axial force due to thrust plus the maximum drag force expected. The results are shown below:

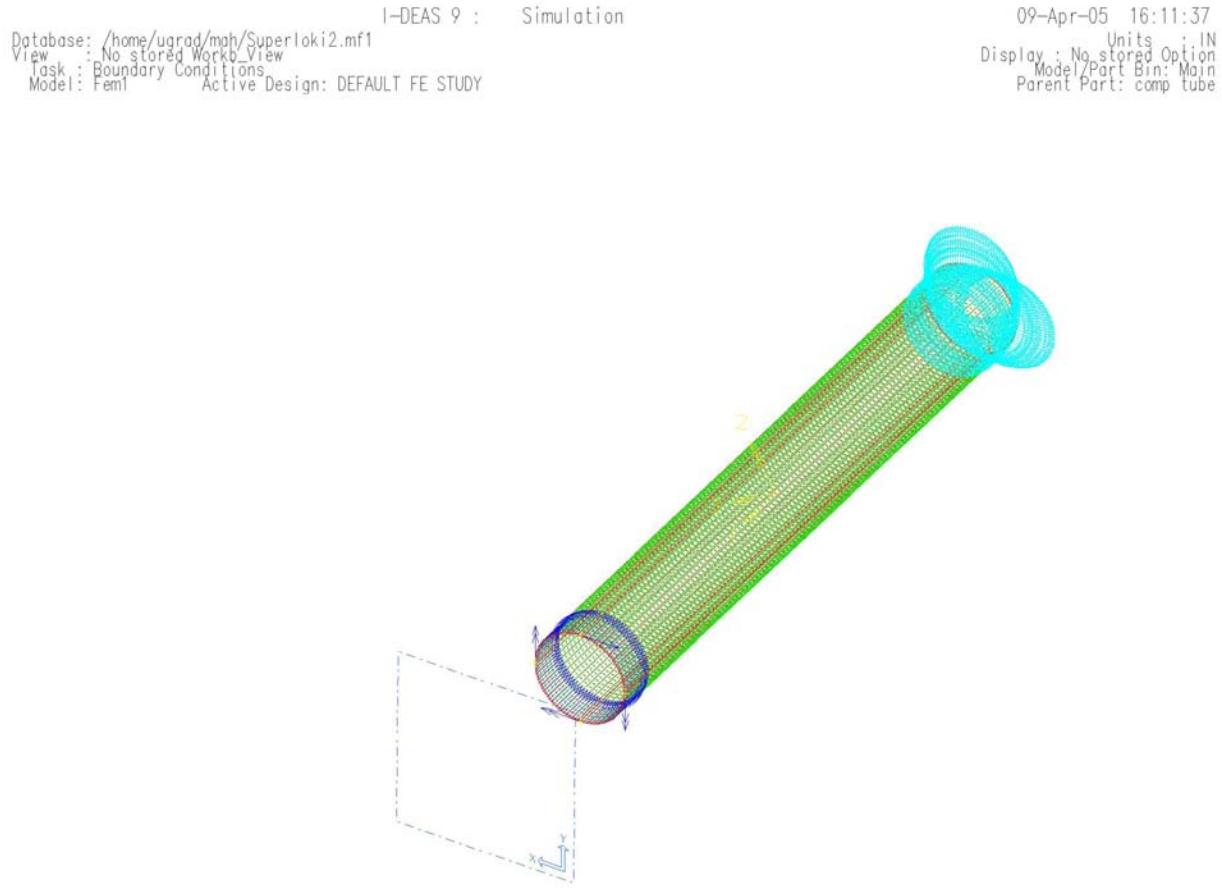

Figure 34: Composite Tube Mesh (green) with forces (dark blue) and constraints (light blue) under compression and torsion applied at one end

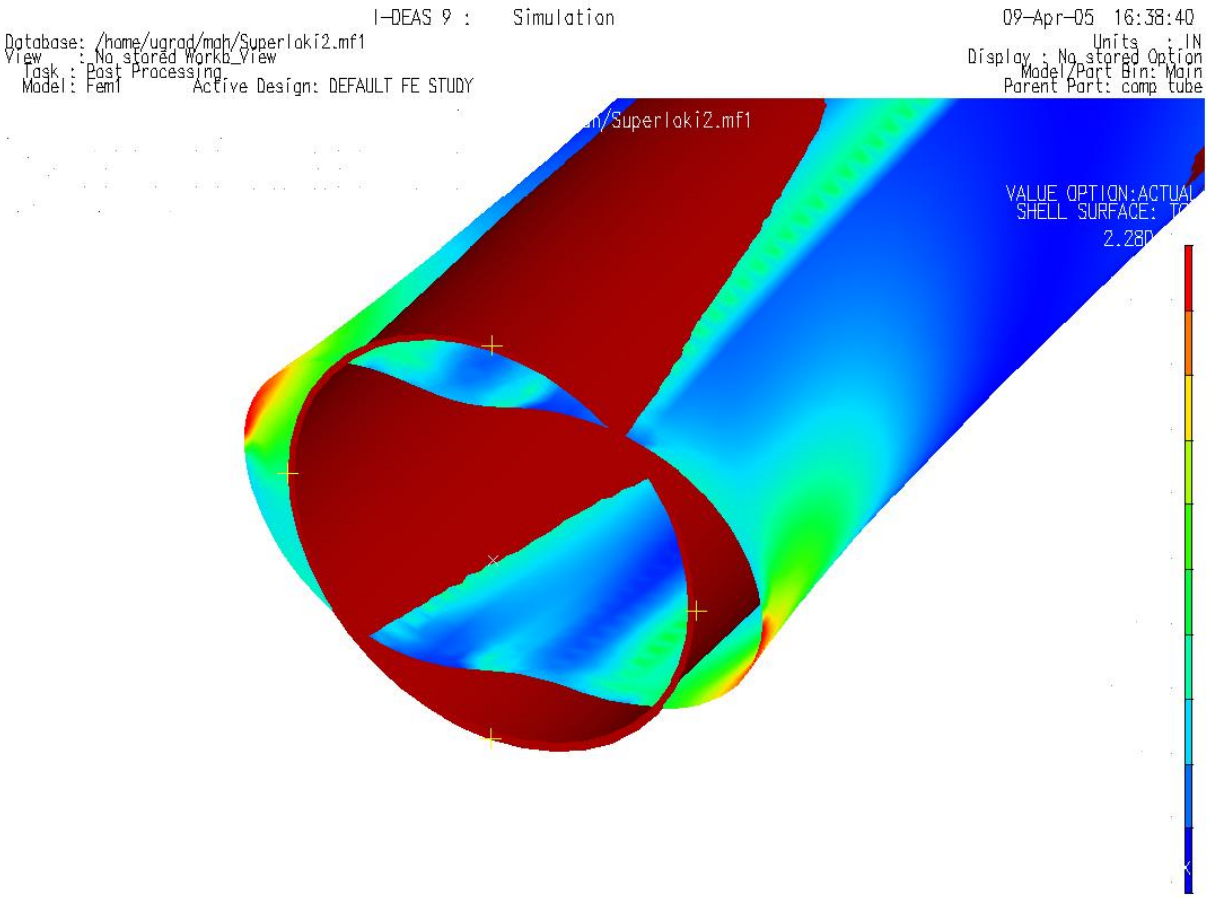

Figure 35: Composite tube stresses and strain

$\sigma_{VMMax} = 22800\,psi$ and the max displacement is $\varepsilon_{\max} = 0.0604 inches$

This value agrees closely with the 21,500 psi determined earlier and is greater in magnitude and was, therefore, considered in the material selection.

Again these values are within the limits of typical carbon fiber tubing readily available in standard sizes that correspond to the sizes recommended herein. The analysis for the forces due to the ejection charge is illustrated in the following figures.

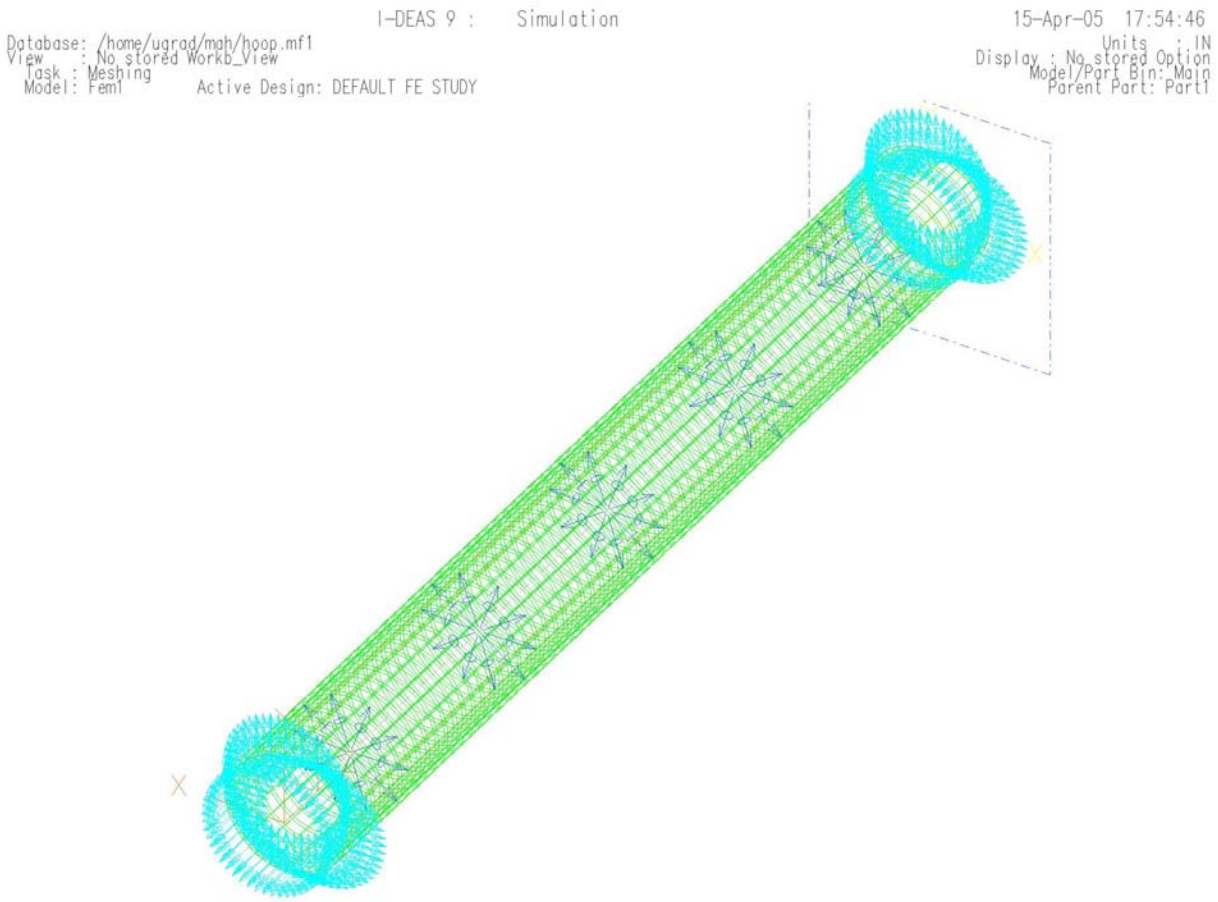

Figure 36: Composite Tube Mesh (green) with forces (dark blue) and constraints (light blue) under internal pressure applied on the inner surface

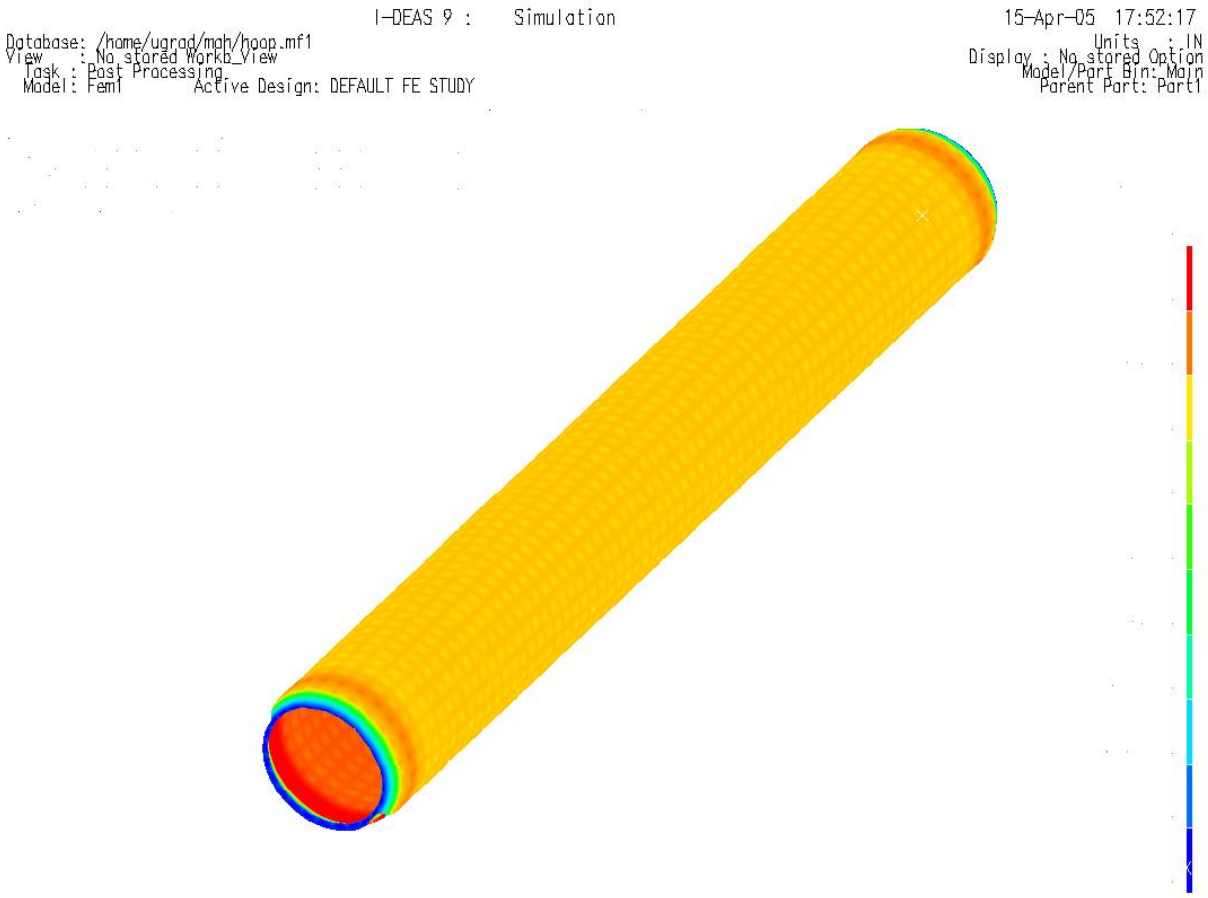

Figure 37: Composite tube stresses and strain

$\sigma_{VMMax} = 18100\,psi$ and the max displacement is $\varepsilon_{\max} = 0.0131 inches$

This value is greater than the 15,000psi calculated by hand and as usual will be the value used for material selection.

It can be concluded from these results that the materials selected meet the expected loads by the following safety factor:

$$MS = \frac{\sigma_{allowable}}{\sigma_{required}} - 1 \qquad (105)$$

Table 3: Mechancial Safety Factors

Safety Factor	
Fin Stresses	2.75
Tube compression + torque	0.32
Tube ejection	0.5

* These factors are based on typical material properties of steel and carbon fiber

CHAPTER 7: AERODYNAMIC HEATING

7.1 Solution Development

The tremendous acceleration of the Super Loki rocket during the 2.11-second burn period brings the rocket to an altitude of approximately 2000m (6562ft). At this altitude, the atmospheric density differs little from the sea level value. The speed at this point in the trajectory is approximately 1629m/s (5342ft/s), which at the burnout altitude, and density, produces a very large aerodynamic heat transfer rate. This heating takes place between the body and the surrounding gas and is caused by friction between the two and compression of the gas. The points of greatest heating on a supersonic vehicle are the leading edges of any object, such as wings, fins, pitot tubes, antennas, etc., protruding from the body, and of course the nose. These are stagnation regions, in which the flow comes to a stop relative to the body fixed coordinate system, and the temperature is, correspondingly, the stagnation temperature as shown in fig 41 from [5]

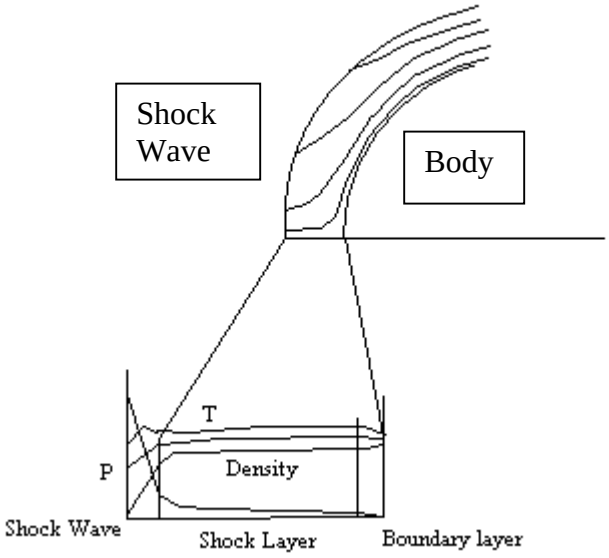

Figure 38: Stagnation region properties on a blunt body

The heating produced in the stagnation region in high-speed flow can easily exceed the thermal capacity of typical aerospace materials. This is especially true in the hypersonic regime, usually defined as beginning at Mach 5 and up. The original configuration of the Super Loki reaches a Mach number of about 4.86 and is therefore not considered Hypersonic. However, Hypersonic vehicles are usually designed to fly at very high altitudes and therefore very low densities. As mention the Super Loki moves very fast in high density air and produces a great deal of heat. It would seem therefore that common aerospace materials may be inadequate for use on such a vehicle. This is especially true of any composite components.

A comparison of stagnation heating and material melting temperatures would seem to back this claim up. A direct comparison is deceptive however and would only be true if all of the heat

generated went into the body. The friction and compression that causes the heating is between the air and the body and both are heated. As described in [6], most of the heat goes into the surrounding air. Consequently, the heating of the body would only reach the stagnation temperature if its high velocity were maintained constant for long enough to reach the steady state value. Convection in the boundary layer, conduction in the solid, and radiation all play a role in the surface heating temperature of a high-speed body. The stagnation temperature is given by, $T_{0\infty} = \frac{V_\infty^2}{2 \cdot cp} + T_\infty$, where cp is the specific heat at constant pressure, T_∞ is the free stream temperature (a function of altitude) and V_∞^2 is the square of the free stream velocity. A determination of the magnitude of the aerodynamic heating as a function of time is therefore essential, in determining the skin temperature of the rocket. Additionally, as will be seen, even though the steady state temperatures are not reached by the rocket surface, very high temperatures are nevertheless achieved. If these high temperatures were not reduced, significant body and payload temperatures would be experienced that would exceed maximums for these components. An ablative coating must be added to the exterior, in order to solve this heating problem. The design considerations involved in choosing the proper ablative material properties and thickness will be considered in section 7.1.8.

7.1.1 Atmospheric Data

During the ascent to apogee, nearly the entire atmosphere is traversed. Thermodynamic properties vary significantly during this journey. Static temperature, density, pressure, kinematic viscosity, and the speed of sound will affect the flow around the body as it increases

in altitude. In order to take into account the change in these ambient parameters, equations based on the standard atmospheric model [9] were used as functions of altitude.

The thermodynamic properties of the atmosphere used in the MATLAB analysis programs were modeled by curve fitting to data points taken from [9] or by established thermodynamic principles. The density and pressure of the atmosphere drop off rapidly with increasing altitude. The change in pressure with altitude is given by the following equation:

$$\ln\left(\frac{p}{p_{sl}}\right) = -\int \frac{g}{RT}\,dh \tag{106}$$

Where p is the pressure at a given altitude, p_{sl} is the pressure at sea level, g is the acceleration due to gravity, R is the universal gas constant, T is the temperature, and h is the height above sea level (altitude). Note that although SI and English units are given throughout the text, the charts below are given in English units, as the MATLAB code was written using this system of units. The final results of the MATLAB code simulation are easily convertible.

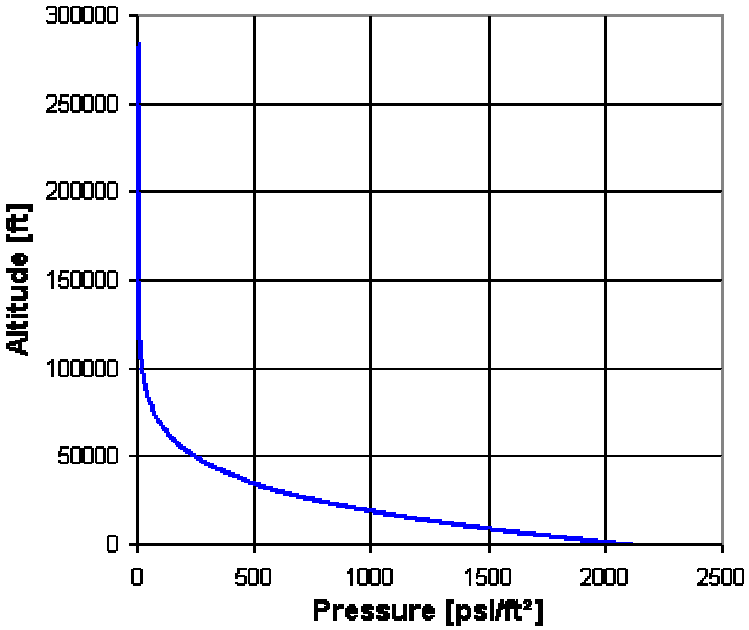

Figure 39: Atmospheric pressure drop with altitude

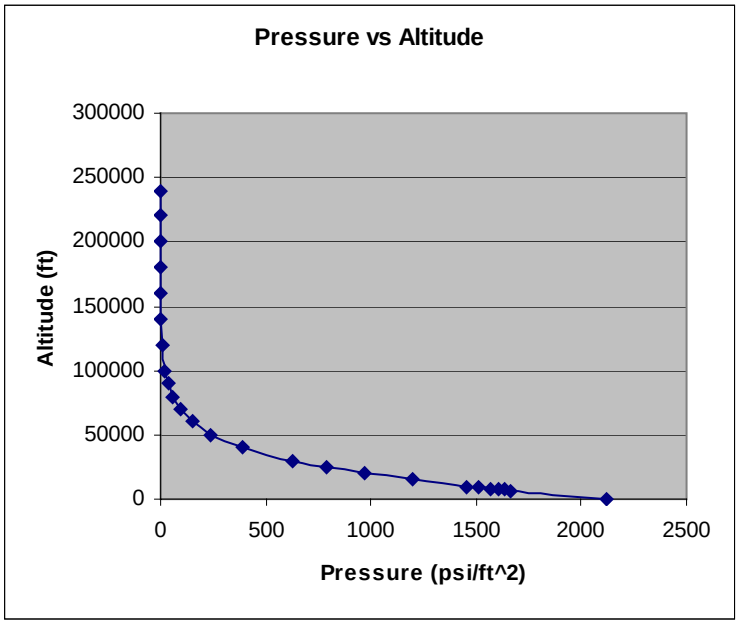

Figure 40: Calculated pressure drop with altitude

Density may be calculated by the equation of state:

$$\rho = \frac{p}{R \cdot T} \tag{107}$$

Where p is the pressure, R is the gas constant for air, T is the static temperature and ρ is of course the density. By curve fitting atmospheric data given by [9], an approximate equation based on altitude was obtained and used in the MATLAB code:

$$\rho = 0.0024 \cdot e^{-4\times10^{-5} \cdot h} \tag{108}$$

Where h is the altitude, and this equation gives density in slugs per cubic foot.

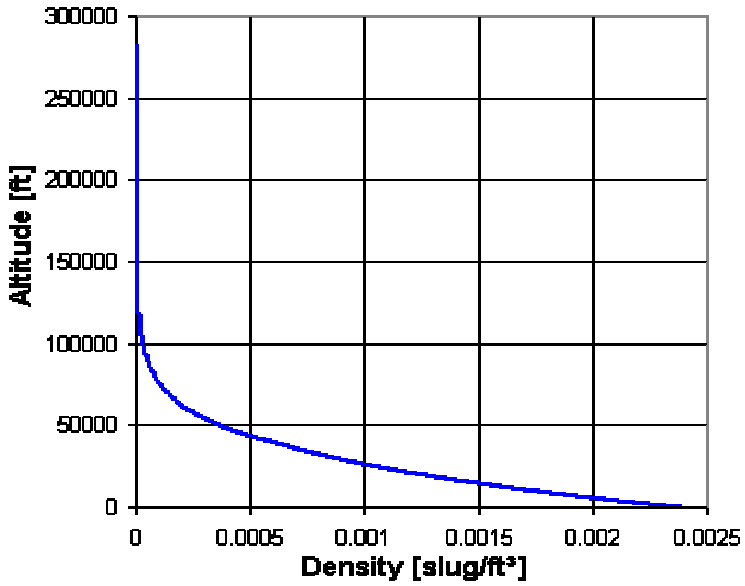

Figure 41: Change in density with altitude

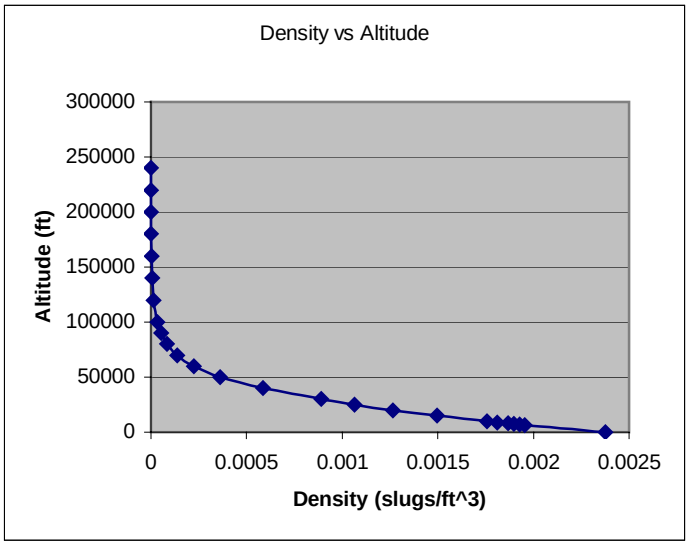

Figure 42: Calculated change in density with altitude

Temperature, and the speed of sound fluctuate significantly with altitude.

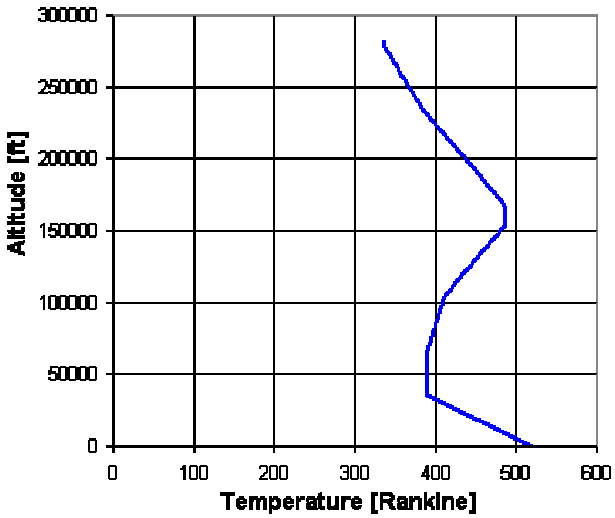

Figure 43: Change of temperature with altitude

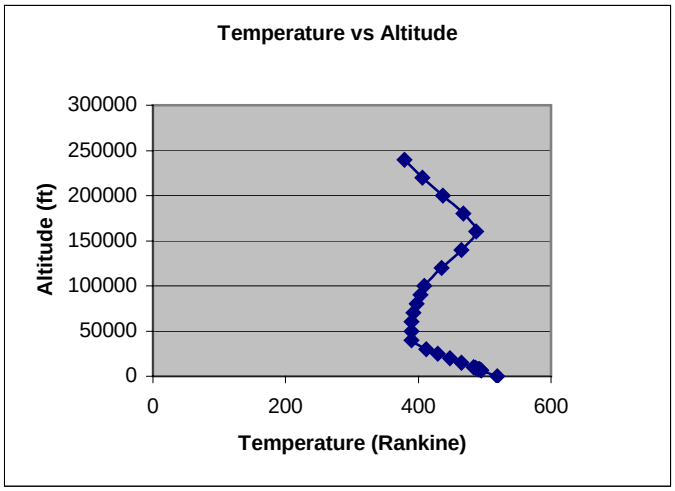

Figure 44: Calculated change in temperature with altitude

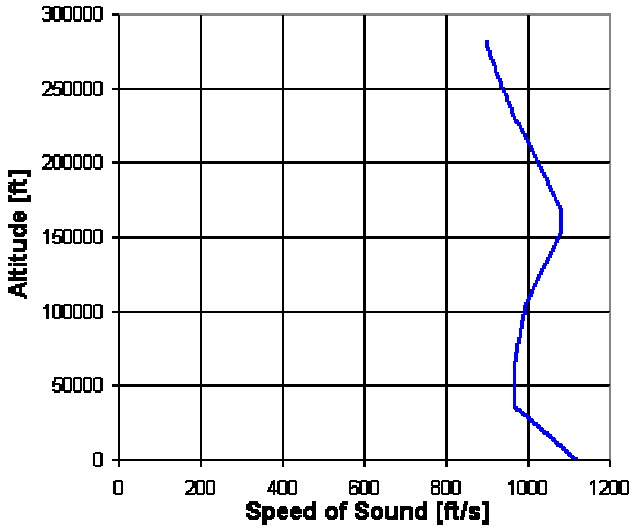

Figure 45: Change in speed of sound with altitude

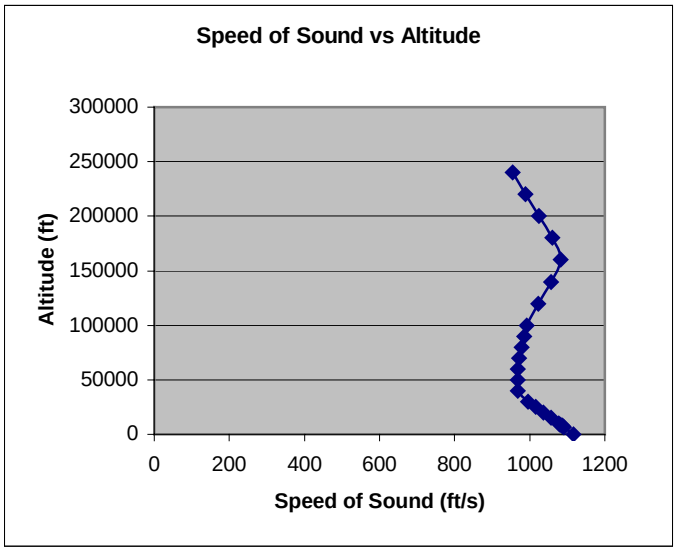

Figure 46: Calculated change in speed of sound with altitude

The equations for temperature and the speed of sound were found by curve fitting:

$$T = 4.415e-24 \cdot h^5 - 2.138e-18 * h^4 + 1.906e-13 \cdot h^3 3.295e-8 \cdot h^2 - 4.529e-523.1 \qquad (109)$$

$$a = -2.6359e-13 \cdot h^6 + 8.2428e-24 \cdot h^5 - 3.785e-18 \cdot h^4 + 4.6765e- \\ 13 \cdot h^3 + 1.6709e-8 \cdot h^2 - 4.5495e-3 \cdot h + 1116.5 \qquad (110)$$

The equation for kinematic viscosity may be calculated from the ambient temperature, and is derived from the Sutherland law of viscosity.

$$\mu = \frac{0.1458 \times 10^{-5} \cdot \sqrt{T}}{1 + 110/T} \qquad (111)$$

7.1.2 Shock wave detachment distance and shape

The Super Loki has an ogival nose cone with a blunted tip. The term ogival, refers to a body that is curved as a circular arc. Blunting means that the tip of the nose cone does not come to an infinitesimally small point but is rather flattened or rounded. This shape, the velocity of the vehicle, and the local thermodynamic properties of the atmosphere determine the shock wave characteristics. A pointed (no blunting), slender, supersonic vehicle has a shock wave attached to its tip, as long as the semi-vertex angle does not exceed some critical angle. This angle depends on the Mach number and charts to determine this angle are provided in [20]. If the semi vertex angle of a pointed body exceeds the critical angle, the shock wave will become detached.

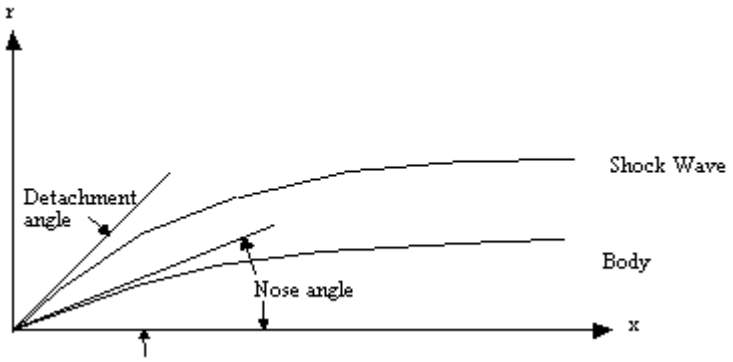

Figure 47: Shock wave attached to a pointed body

This is always the case for a blunt body and the distance of this detachment will be calculated below.

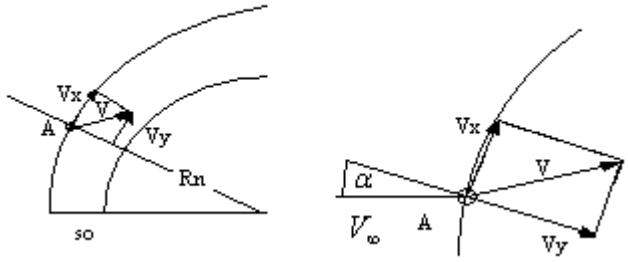

Figure 48: Velocity components in stagnation region

The flow in the stagnation region is of importance because this is where the highest temperatures are experienced. In the analysis of the aerodynamic forces the nose will be modeled as pointed in the preliminary MATLAB analysis. This simplification does not introduce significant error in the case of such slender bodies as the Super Loki and the modified dart. A justification of this simplification is given in [2]. The difference in the pressure distribution due to rotation is a function of the hypersonic similarity parameter $K_{HS} = M_\infty / \lambda_N$. For the original configuration of the Super Loki the maximum value of this parameter is about .916. At $K_{HS} = 1$ the error cause by assuming irrotational flow is about –5%. This underestimation of the pressure distribution for the Super Loki occurs over a very short period of the whole flight and diminishes rapidly, after motor burnout. The modified configurations also meet this criterion as the nose is of almost exactly the same slenderness ratio. For velocity, drag and pressure calculations, this error is deemed acceptable. However, in the calculation of stagnation point heating, the actual nose tip geometry must be considered. This is important because the amount of heat at the stagnation point is proportional to the area of the stagnation region, and the assumption of a pointed nose would result in significant errors in the calculation of heating.

As stated, at motor burn out, the rocket is at maximum velocity 5331 ft/s (1625 m/s), or Mach 4.86, and the altitude is 1524 m (5000ft). For the shape of the nose cone, a spherical radius of 0.00381 m (0.15 in) is given by [1] on the original configuration and the same bluntness is used on the modified darts. The solution method employed for stagnation point heating analysis is taken from [6].

7.1.3 Spherical tip

At $\phi = 0$ (see figure 38), aligned with the axis of symmetry, the detached shock is perpendicular to the free stream velocity, assuming a zero angle of attack, and is considered a normal shock at this point. The parameters immediately behind this shock wave are easily found from the isentropic normal shock tables [20]. The temperature, Mach number and density give the speed of sound and velocity on the downstream side of the shock wave. From the velocity behind the shock can be obtained the velocity gradient on the stagnation streamline and, consequently, the maximum, theoretical temperature on the body. As an example, the shockwave detachment distance, radius of curvature and shape are calculated below for the original configuration.

First the detachment distance s_{00} will be determined given the nose radius $R_N = 0.00381 \cdot m$ and density ratio $\rho_{SW} = 0.204$ across the shock wave at $M_\infty = 4.86$:

$$s_{00} = \rho_{SW} \cdot R_N \tag{112}$$

$$s_{00} = 6.203 \times 10^{-4} \cdot m$$

The radius of the shock wave may be calculated, along the stagnation streamline by:

$$R_{0sphere} = 0.78 \cdot R_N \cdot \left(1 + \sqrt{\frac{8 \cdot \rho_{SW}}{3}}\right) \tag{113}$$

$$R_{0sphere} = 5.166 \times 10^{-3} \cdot m$$

The temperature immediately behind the shock (Kelvin):

$$T_{SW} = 5.233$$

$$T_2 = T_1 \cdot T_{SW} \tag{114}$$

$$T_2 = 1456 \cdot K$$

Where T_1 is the temperature just ahead of the shock wave, and in this case is taken from the standard atmospheric tables at burnout altitude. From the detachment distance and the density ratio, the shock radius and shock shape can be determined:

7.1.4 Shock wave shape

$$x_{st}(rr) := -\left(s_{00} \cdot R_N\right) + \frac{rr^2}{2 \cdot R_{s0}} \tag{115}$$

Where rr is the vertical distance from the axis of symmetry to the shock wave.

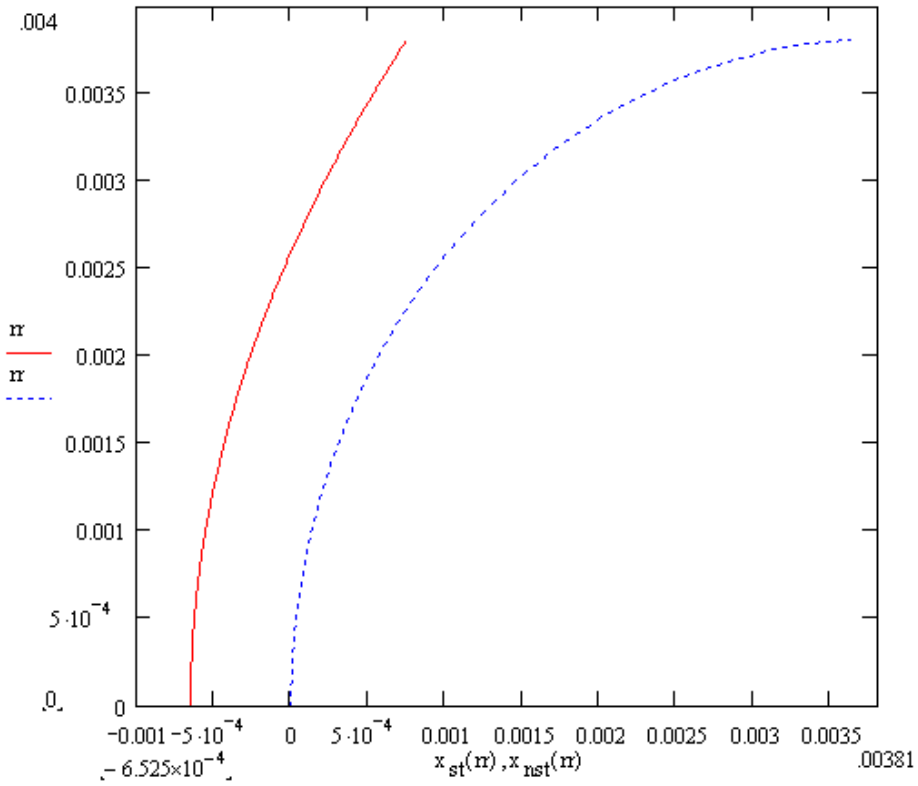

Figure 49: Shock wave detachment distance on a spherically blunted nose calculated

7.1.5 Velocity Gradient

The velocity gradient in the stagnation area is heavily dependent on the detachment distance and the free stream Mach number.

$$\beta_{sph} := \frac{V_1}{R_{s0}} \cdot \sqrt{1 - \left(1 - \rho\_n\right)^2} \tag{116}$$

$$\beta_{sph} = 1.906 \times 10^{5}\ s^{-1}$$

As a check on these results, some comparison with other methods and experiment is required. For the shock wave detachment distance, the following equation, from [2] gives similar results to those above;

$$s_{00} = \rho_{SW} \cdot R_N \tag{117}$$

$$s_{00} = 6.203 \times 10^{-4} \cdot m$$

$s_{0\_sph} := .78 \cdot \rho_{\_n}$ $\quad s_{0sph} := s_{0\_sph} \cdot R_N$ $\quad s_{0sph} = 6.074 \times 10^{-4}\ m$

This last result shows only a 7.4% difference between the two methods used.

As for the velocity gradient, the calculated value agrees well with experimental results. At the time in the flight considered here, 2.11 seconds (motor burnout), the Mach number is 4.86 and the non-dimensional velocity gradient is expected to be around 1.1-1.3.

$$\beta_{sph} := \frac{V_{2n}}{s_{0sph}} \cdot \sqrt{1 - \left(1 - \rho_{\_n}\right)^2} \tag{118}$$

$\beta_{sph} = 2.84 \times 10^{5}\ s^{-1}$ $\quad \frac{2 \cdot R_N}{V_1} \cdot \beta_{sph} = 1.332$

The theoretical results are typically high, a comparison is given in [2], and the actual value in this situation is probably closer to 1.1, the theoretical result being therefore a conservative one.

7.1.6 Transient Heating

The equation for the transient heating in the stagnation region is given by:

$$\frac{dT_w}{dt} = \frac{f_2}{G} \cdot \sqrt{\beta \cdot \rho_\infty \cdot \mu_\infty} \cdot g \cdot c_p \cdot \left(T_{aw} - T_w\right) - \frac{\varepsilon \cdot \sigma \cdot T_w^4}{G} \tag{119}$$

Where $f_2 = 0.763 \cdot \mathrm{Pr}^{-0.6}$, G is the heat absorption capacity of the skin material, βsph is the velocity gradient, ε is the emissivity of the material, and σ is the Stefan-Boltzmann constant. The procedure for calculating transient stagnation heating is the same for that on the rest of the body, with the exception that equation (119) is used as apposed to the equations (123) and (125), to follow. Therefore the description of this procedure will be given in section 7.1.7.

7.1.7 Nose Cone and Afterbody

As with the stagnation region, the nose cone and after body may, of course, be considered in the steady state. A more thorough analysis was desired however in order to determine just how far away from the steady state temperature the body temperature is during the entire flight. The steady state, insulated, i.e. adiabatic, wall temperature is easily determined as the free stream stagnation temperature multiplied by a recovery factor. The recovery factor is the ratio of the difference between the adiabatic wall and the free stream temperature and the difference between the stagnation and free stream temperature. This difference is a function of the Prandtl number, which is the ratio of frictional heating over the conduction heating. The recovery factor is given as follows for laminar and turbulent flow:

$$r_{lam} = Pr^{\frac{1}{2}} \qquad r_{turb} = Pr^{\frac{1}{3}} \tag{120}$$

Generally these quantities vary little and are equal to roughly 0.85 and 0.89 respectively. Therefore the steady state temperature for the composite portion of the body may be calculated with the following equation:

$$T_0 = T_{inf} \cdot \left[1 + r_{turb} \cdot \left[\frac{\left(\gamma_1 - 1 \right)}{2} \cdot M_2^2 \right] \right] \tag{121}$$

Where T_o is the adjusted temperature. At the maximum Mach of the original configuration the adjusted temperature is 2607 R. The stagnation temperature is:

$$T_0 := T_1 \cdot \left[1 + \left[\frac{\left(\gamma_1 - 1 \right)}{2} \cdot M_1^2 \right] \right] \tag{122}$$

and is equal to 2868 R.

This is a steady state temperature at the highest Mach number and at a constant altitude of 5000 ft. This is not meant to represent actual body temperature in flight but to illustrate the method of calculation of the stagnation and adiabatic temperatures and provide a reality check to the quantities calculated in the transient case.

In considering the transient temperature of the composite body, material properties and thermodynamic properties must be considered. The atmospheric properties, as functions of altitude were considered in section 7.1.1. General information on related materials, fiberglass and epoxy, steel etc. were available from [7]. The specific heat of the generic composite material chosen, 38% epoxy, was taken as .21, density as 89 lb/ft^3, emissivity as 0.88 and the

wall thickness as 0.01042ft. In order to calculate the skin temperature, the following equations from [6], were used:

Laminar;

$$\frac{d}{dt}T_w = \frac{1}{JG}\cdot\left[\frac{1}{2\cdot Pr^{\frac{2}{3}}}\cdot\left(\frac{T_{aw}-T_w}{T_{0inf}-T_{inf}}\right)\right]\cdot c_{finf}\cdot\frac{\rho_{inf}\cdot a_{inf}^{3}}{2}\cdot M_{inf}^{3}-\frac{\varepsilon\cdot\sigma\cdot T_w^{4}}{G} \tag{123}$$

$$T_{aw}-T_{inf} = Pr^{\frac{1}{2}}\cdot\left(T_{0inf}-T_{inf}\right) \tag{124}$$

Turbulent;

$$\frac{d}{dt}T_w = \frac{1}{JG}\cdot\left[.6\cdot\left(\frac{T_{aw}-T_w}{T_{0inf}-T_{inf}}\right)\right]\cdot c_{finf}\cdot\frac{\rho_{inf}\cdot a_{inf}^{3}}{2}\cdot M_{inf}^{3}-\frac{\varepsilon\cdot\sigma\cdot T_w^{4}}{G} \tag{125}$$

$$T_{aw}-T_{inf} = Pr^{1/3}\cdot(T_{0\,inf}-T_{inf}) \tag{126}$$

Equations (123) and (125) are first order but fourth degree with variable coefficients. In order to solve these equations the fourth order Runge-Kutta method serves well. The iterative procedure for the Runge-Kutta method as applied to a 40-foot long rocket with an ogival nose cone, applied 20 feet aft of the tip of the nose is laid out, as an example in [6]. The results based on this example are shown below and compare well with the results in [6].

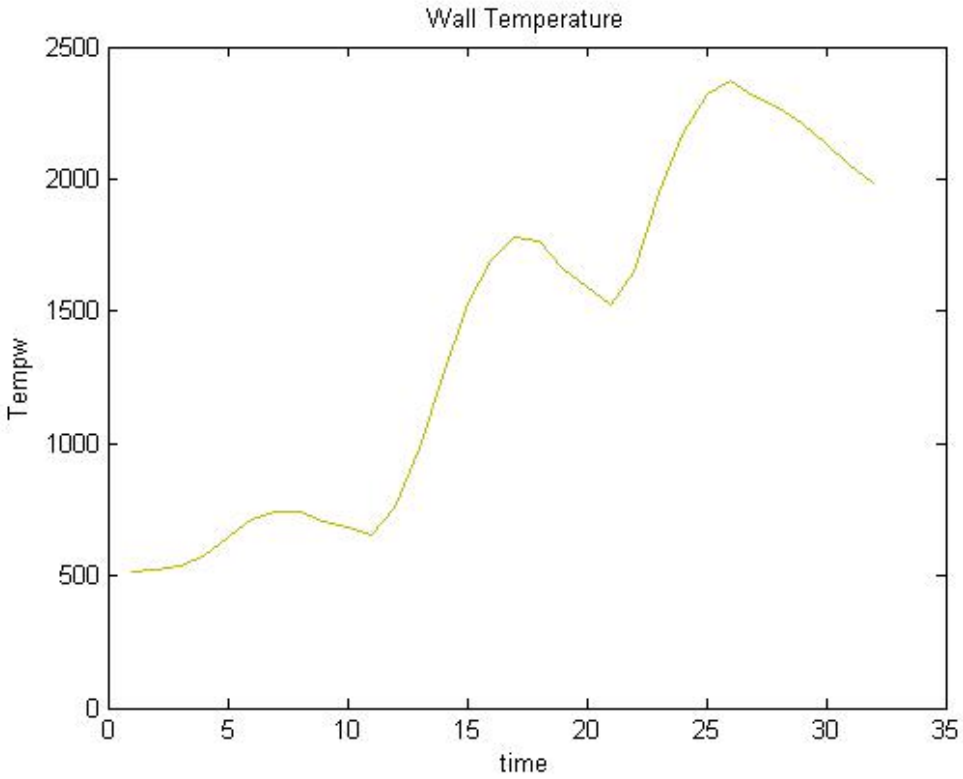

Figure 50: Calculated skin temperature on an example rocket 40 ft aft of the nose

There is a slight difference between these two results however, and it comes from a different method of calculating the skin friction coefficient. In [6] a nomograph was used, whereas the methods outlined in chapter 3 were used here. The difference grows with altitude and the maximum error is about 10% above 160000ft. Almost all of the heating imparted to the Super Loki occurs well below this atmosphere and this error is therefore not considered significant.

7.1.8 Ablation

In order to alleviate the heating caused by the high velocities of the Super Loki, the tip of the nose is blunted. This blunting distributes the heat transferred to the nose over a larger area, thereby reducing the maximum temperature at any one point. The blunting in this case is not

considered to be sufficient to reduce the temperatures below tolerable values. Greater blunting would increase the drag of the rocket, which is already increased by the increased diameter, and should be therefore minimized. For this reason, at least in part, an ablative coating was used in the original design of the Super Loki.

As the modified dart will not be of significantly differing geometry or material, in regards to heating, ablative coating is considered to be a likely necessity. A calculation of skin temperature without ablation will be calculated for the stagnation region as well as downstream. Although the composite portion of the body is not normally a stagnation region, it does experience significant heating. Therefore the calculation of the heating over the nose and cylindrical portion will be made in addition to the stagnation region. The other parts of the body, the fins and booster, were never coated on the original design with ablative material and are therefore not expected to experience heating that would affect their performance significantly. These portions of the body will not be considered in the temperature calculations.

In figure 38, repeated here for convenience:

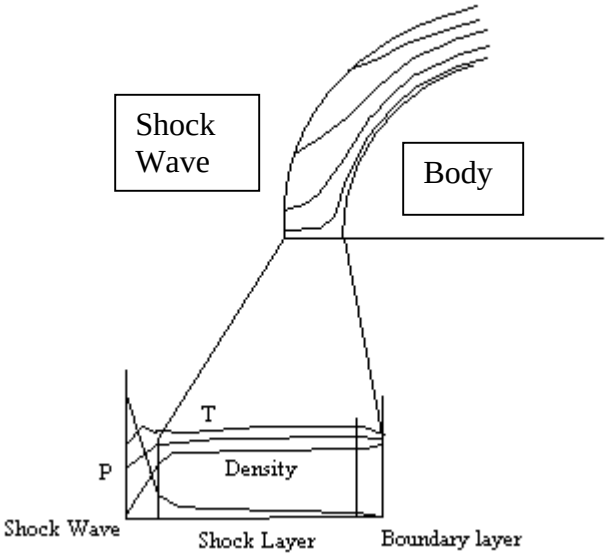

Figure 51: Stagnation region properties on a blunt body

The thickness of the ablative material is given by, θ. As the material ablates, this thickness is reduced. The rate of mass loss of ablative material is:

$$m = \frac{q_0}{\left(q^*\right)^0} \cdot \left(1 - \frac{q_r}{q_0}\right) \cdot kgf/(m^2 \cdot s) \tag{127}$$

Where q_0 is the heat transfer without ablation, $\left(q^*\right)^0$ is the heat transfer due to ablation and q_r is the heat transfer due to radiation which is determined during the temperature calculation in the MATLAB code, according to the method in [6].

$$q_0 = \frac{0.763}{\mathrm{Pr}^{2/3}} \cdot \sqrt{\tilde{\lambda} \cdot \rho_0^{'} \cdot \mu_0^{'}} \cdot \left(i_r - i_w\right) \cdot \left(\overline{\rho}_w \cdot \overline{\mu}_w\right)^{1/10} \tag{128}$$

Where $\tilde{\lambda}$ is the velocity gradient along the stagnation streamline between the detached shockwave and the surface of the body. $\rho_0^{'}$ and $\mu_0^{'}$ are the stagnation density and kinematic

viscosity behind the shock. i_r and i_w are the recovery and wall enthalpies respectively. The quantity containing, $\overline{\rho}_w\ \overline{\mu}_w$ may be represented by:

$$\left(\overline{\rho}_w \cdot \overline{\mu}_w\right)^{1/10} = \left(\frac{\rho_w \cdot \mu_w}{\rho_0^{'} \mu_0^{'}}\right)^{1/10} \tag{129}$$

The recovery enthalpy may be determined by the following:

$$i_r = i_\delta + r \cdot \frac{V_\infty^2}{2} \tag{130}$$

Where i_δ is the enthalpy just outside of the boundary layer and r is the recovery factor, given by:

$$r = \mathrm{Pr}^m \tag{131}$$

Where $m = 1/2$ or $m = 1/3$ for laminar or turbulent flow respectively. The wall enthalpy is determined from the temperature calculated by the method outlined in [6] for stagnation flow. The heat transfer with ablation:

$$\left(q^*\right)^0 = c_p \cdot T_i + \overline{m} \cdot \left(i_v + \beta \cdot \left(i_0 - i_w\right)\right) \tag{132}$$

Where c_p is the specific heat at constant pressure, T_i is the surface temperature of the ablative material $\overline{m}$ is a coefficient that represents the ratio of sublimation to fusion. Fusion is the liquification of the ablative material, and sublimation is vaporization of the ablative material. The most effective ablators, called char ablators, sublimate with very little fusion. In the limiting case of total sublimation, $\overline{m} = 1$. This will be the case considered here, as many

ablators such as Teflon (C2F4) do not have a significant fusion phase. i_v is the vaporization enthalpy, β is the transpiration coefficient and is given as:

$$\beta_L = N \cdot \left(\frac{29}{\overline{\mu}_v} \right)^{\overline{\alpha}} \tag{133}$$

for laminar

and

$$\beta_T = \frac{1}{3} \cdot \beta_L \tag{134}$$

for turbulent. From experimental data, N and $\overline{\alpha}$ lie in the range:

$0.67 \le N \le 0.72$ and $0.25 \le \overline{\alpha} \le 0.4$

$\overline{\mu}_v$ is the molecular weight of the ablative material and i_0 is the stagnation enthalpy. N, $\overline{\alpha}$, $\overline{\mu}_v$, i_v must be determined from manufacturer data. The mass transfer rate for each point on the body being known, the thickness of ablative material required to protect the rocket body can be determined. Given the material properties for Teflon a commonly used ablator, the effective heat of ablation is:

$$q^*_{0teflon} = \frac{527 \cdot i_0}{i_0 + 1165} + 5 \cdot i_0 \tag{135}$$

Where i_0 is the stagnation enthalpy just after the shock wave and is given by:

$$i_0 = cp \cdot T_t + \frac{V_\infty^2}{2} \tag{136}$$

The mass loss over a given time period can now be calculated by:

$$\dot{m} = \frac{q_0}{q_0^*} \tag{137}$$

Given the thermal conductivity and specific heat of the ablative material, the thickness of the material as a function of this mass loss can be calculated with the following:

$$\theta = \frac{kb}{cb * \dot{m}} \tag{138}$$

Where θ is the thickness of the material, kb is the thermal conductivity and cb is the specific heat of the ablative material.

7.2 Original Configuration Thermal Loading

The method for determining the skin temperature of the unprotected rocket was based on the method in [6] and this method was then applied to the 2 inch Super Loki dart giving the following results:

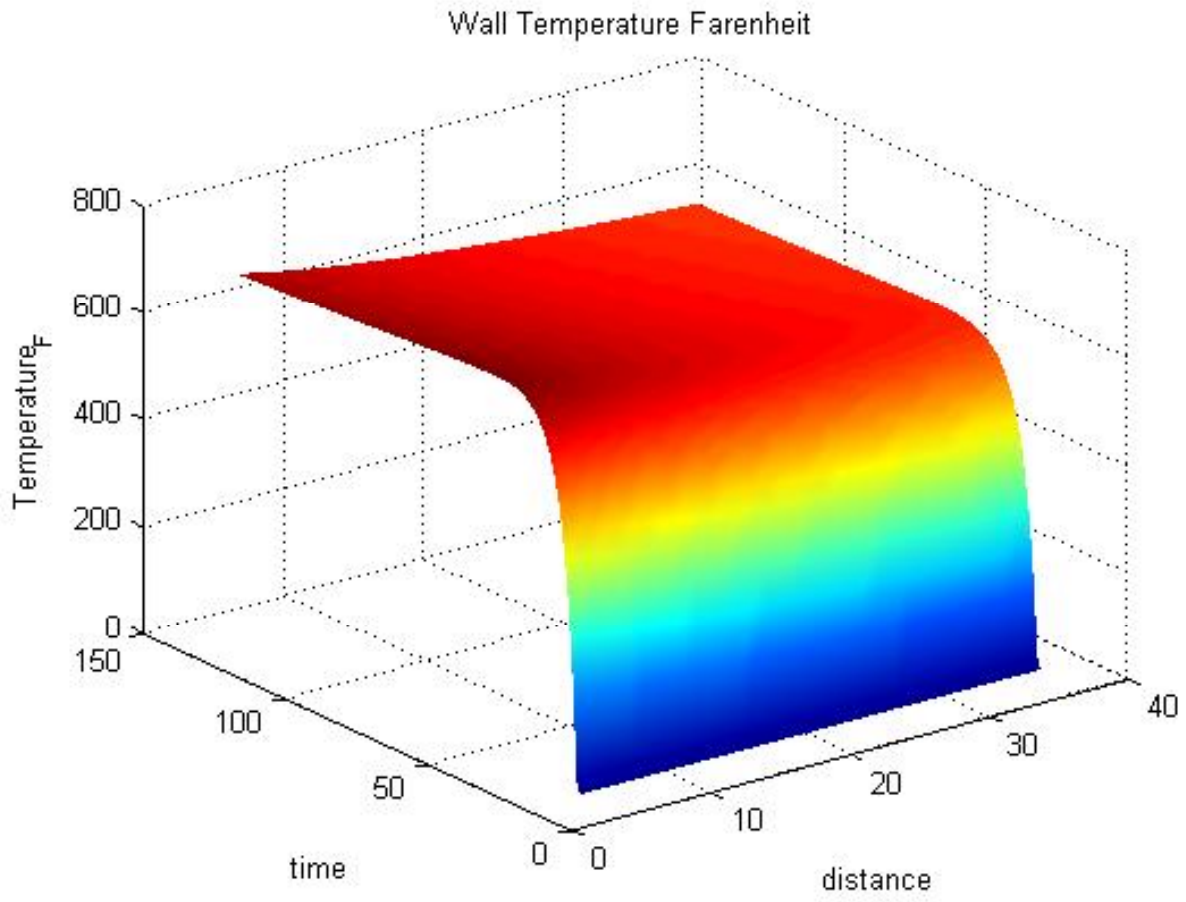

Figure 52: Temperature Fahrenheit along Composite section vs. time

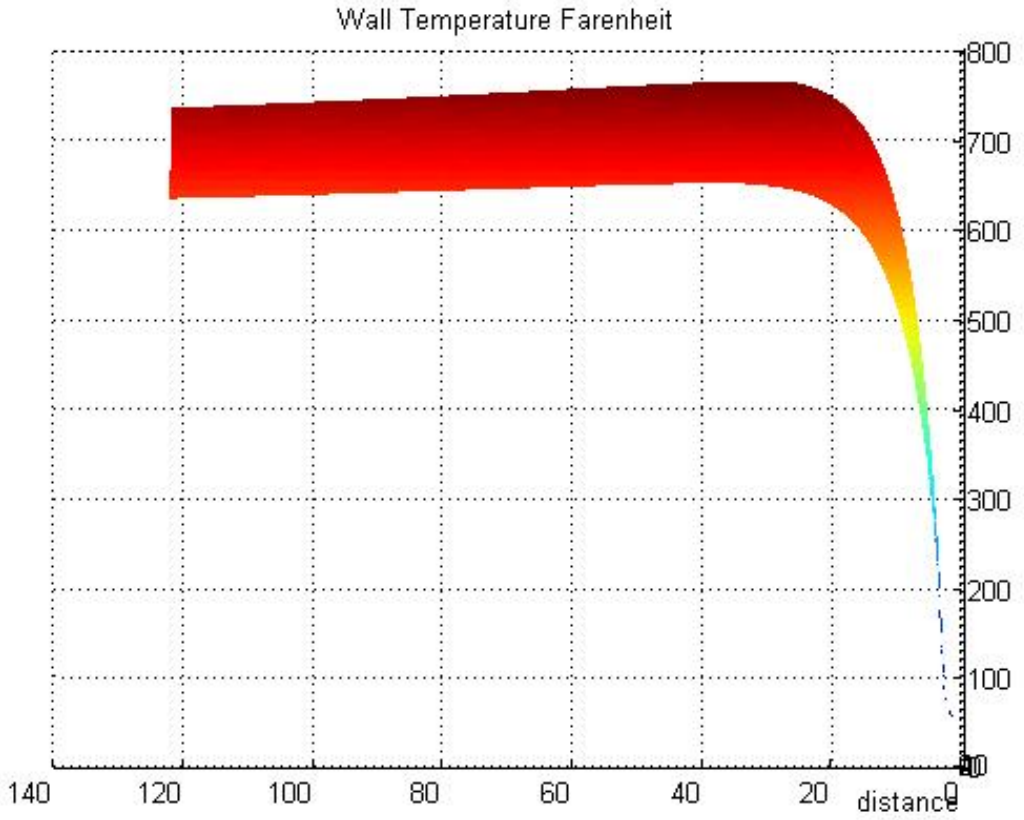

Figure 53: Temperature Fahrenheit along Composite section vs. time

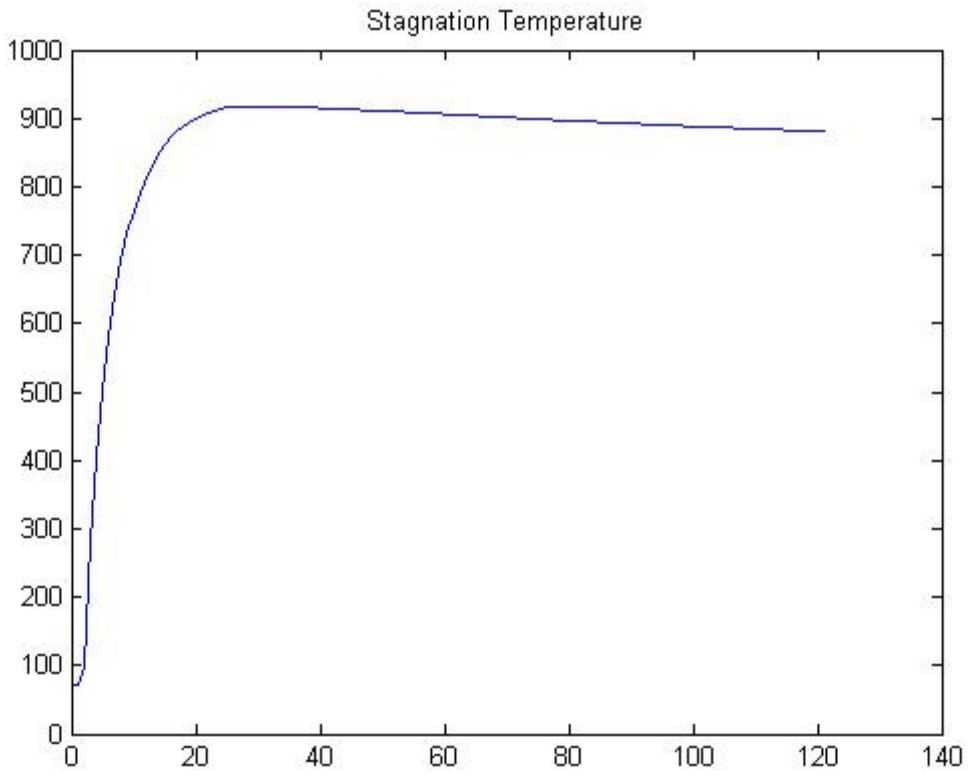

Figure 54: Stagnation Temperature

These values thus calculated, the theoretical ablative coating thickness required to protect the composite section of the body, nose cone and payload may be estimated. Again being conservative, the temperature profile of the existing dart was considered and as the resulting thickness of ablative material was not excessive, this estimate is recommended until actual flight-testing can be done. The thickness of ablative material in inches in the blunt nose stagnation region is given below:

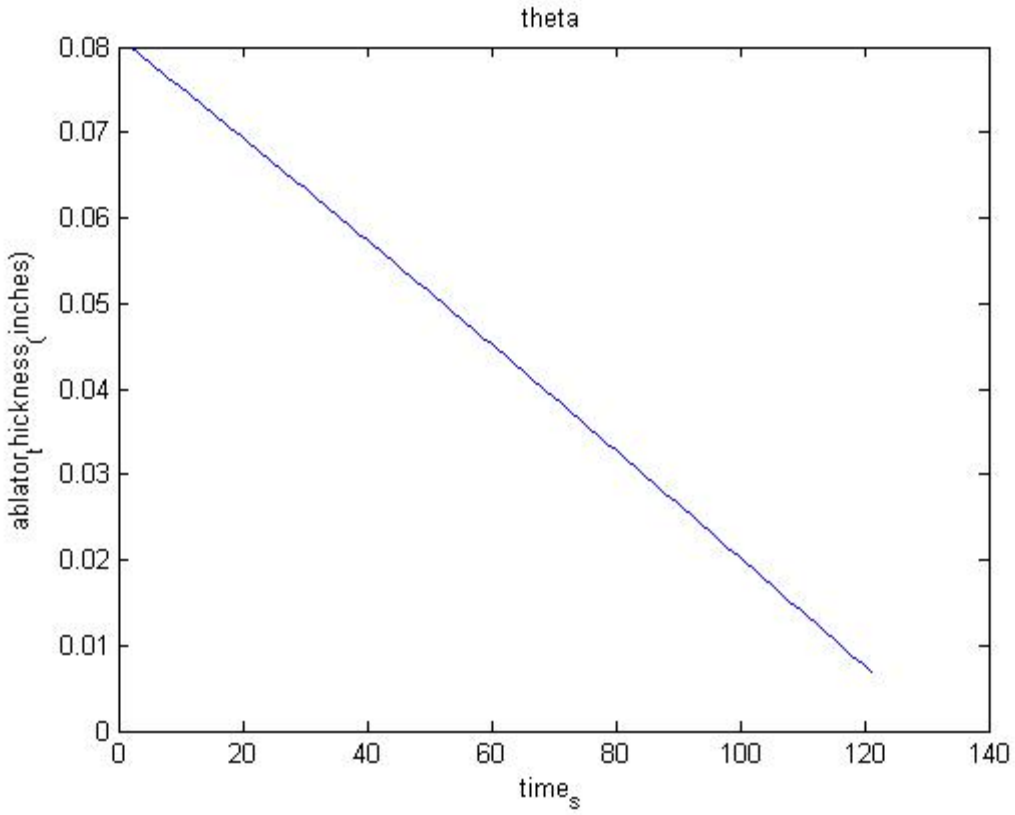

Figure 55: Ablator thickness in inches vs. time (original configuration)

The results above were based on the material properties of commonly available commercial ablators used for aerospace purposes readily available. Based on the results above, a thickness of .125 inches is recommended using typical values of these common ablators:

$$kb = 0.00003889 \frac{Btu \cdot ft}{ft \cdot s \cdot R}$$

$$cb = 0.4 \cdot \frac{Btu}{lb \cdot ft}$$

7.3 Modified Configuration Thermal Loading

The heating on the larger diameter darts is expected to be somewhat lower than that of the 2-inch dart. It is recommended however that the level of heating for the unmodified configuration be used for the modified darts in order to provide a conservative estimate of the ablative coating needed. Therefore, no specific temperature or ablative coating thickness analysis was done for these configurations.

CHAPTER 8: SUMMARY

The procedure related to the development of the applicable equations thus far, was to begin with a 3 inch outer diameter upper stage dart and increase this diameter in half inch increments up to 4 inches. This last value being the greatest considered reasonable given the diameter of the booster and the launcher dimensions. Given a diameter, an estimate of the center of gravity location was made, and then the line of sources method [10] was used to find the center of pressure of the body at zero angle of attack. Based on these results, the size of the dart fins was determined such that the center of pressure of the whole rocket fell behind the cg. This was done for all configurations at varying Mach numbers considered to be within the expected ranges. Then, using the empirical relationships the center of pressure, moment and normal forces on the whole body were determined to ensure static stability. These equations do not include the additional stability imparted by the launcher induced spin and are therefore considered conservative. Mechanical loads were then considered and validated with the IDEAS-9 software package. The thermal loads were obtained using a Runge-Kutta time stepping method and the thickness of ablative coating to maintain reasonable skin temperatures was performed. The results are given below. Appropriate codes used to solve all of the design problems are given in the appendices.

The stage separation problem is diminished by the large contribution of base drag on the booster. The friction drag on the booster is a good deal larger than the dart friction drag and there is nearly as much wave drag on the booster as on the dart as well. It can be concluded with confidence therefore that even the four-inch dart will separate from the booster at motor burnout.

All configurations can be made stable, as long as the recommended materials and dimensions are adhered to. The weight balance of the rocket components can be manipulated to locate the center of gravity at a position forward of the center of pressure, and varying material weights and payload weight can be accommodated within a reasonable range through the use of lead ballast inserts in the nosecone.

The method of calculating the velocity and altitude of the rocket, while basic, provides valuable information. Although this method assumes a zero angle of attack throughout the flight and does not include wind or other perturbing factors, it nevertheless gives a reasonable estimate of the velocity and altitude of the rocket. As long as the rocket is optimally stable, i.e. having a static margin of 1-2 calibers, immediately after separation from the launcher, and due to the fact that the rocket is spun, the angle of attack should remain small, i.e. less than 5º. Knowing the apogee altitude facilitates payload telemetry acquisition for ground base antennas. Given the velocity history, the methods needed to calculate the flow around the rocket body can be determined and the stability and heating can be calculated.

The mechanical loading on the rocket is not excessive when considering the high strength of typical materials available, i.e. steel and carbon fiber tubing. The analysis between the initial hand calculations and the finite element analysis showed good agreement. The maximum difference between these calculations is 20% and the largest results were considered in the analysis. The mechanical analysis showed that a carbon fiber tube can be used that will withstand the loads experienced by the rocket.

Given this material, the skin temperature and ablative coating required to alleviate aerodynamic heating can be determined. The stagnation temperature calculation required the determination of the flow in the stagnation region just behind the detached shock wave. The

well-known relationships for isentropic flow across a normal shock wave were used for this purpose. This led to the calculation of the enthalpy and velocity gradient along the stagnation streamline coinciding with the axis of symmetry of the rocket. The heat transfer to the surface and the effective heat of ablation can then be calculated and therefore the mass loss and thickness of ablative coating required.

CHAPTER 9: CONCLUSIONS AND RECOMMENDATIONS

Based on the analysis and results sited above, the 4 inch dart is recommended for maximizing payload space, using a carbon fiber cylindrical tube 35 inches in length with a steel nose cone and a lead insert used to ballast the body based on exact measurements of component weights. The dart fin assembly should be made of steel and the fins should be reinforced with welded "L" brackets, as opposed to welding the fins to the cylinder with no other support. The nose cone should be ogival to maximize the volume and therefore weight of the nose cone and thus bring the cg forward to a point no further aft than 51 inches from the tip of the 18-inch long nose cone, for all configurations. The nose cone may be modified without necessarily going out of design boundaries but the ogive is recommended as having slightly better drag characteristics than the cone, and both the cone and ogive are superior in mach ranges exceeding 2.0 when compared to other nose shapes. A general selection criteria chart is given in [21].

Common ablative materials are quite capable of shielding the body from the heating experienced as long as it is applied to the recommended depth. It is further recommended that the smaller diameter darts be considered as allowing flexibility in mission requirements and for additional conservatism, a somewhat less than 4 inch dart could be used to ensure drag separation occurs as expected. A descriptive table of the three configurations is given below:

Table 4: Configuration Results

	4"	3.5"	3"
Wtotalfull	100	89	79
Wtotalempty	63	52	42
Wdart	52	40	31
Original Volume		.062ft^3	
	4"	3.5"	3"
Vol Ratio	0.224ft^3	0.181ft^3	0.131ft^3
Max Mach	3.1922	3.86	4.38
Max Altitude	148000	180000	232610

Finally, the volume ratio given in table 4 shows that a significant increase can be achieved, and that if higher altitudes are desired than one of the smaller diameter darts can be used to meet these requirements while still increasing the payload capacity.

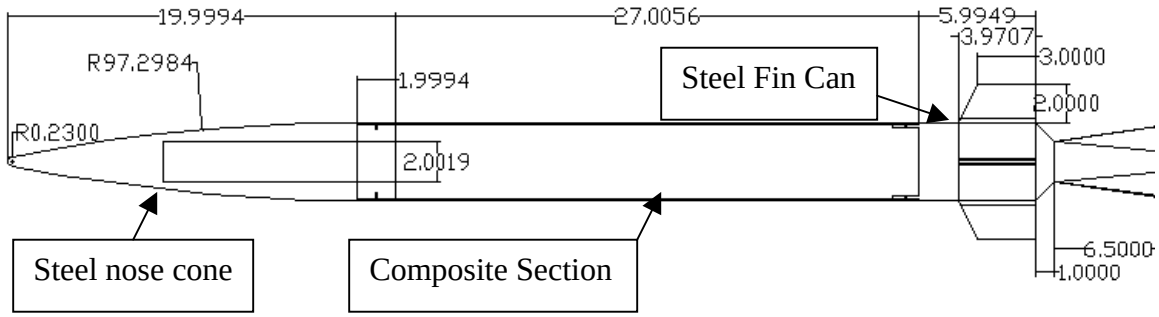

Figure 56: 4-inch dart attached to booster transition

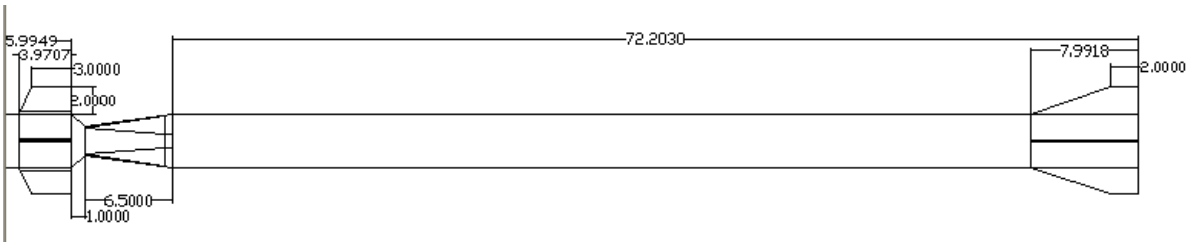

Figure 57: 4-inch dart attached to booster transition

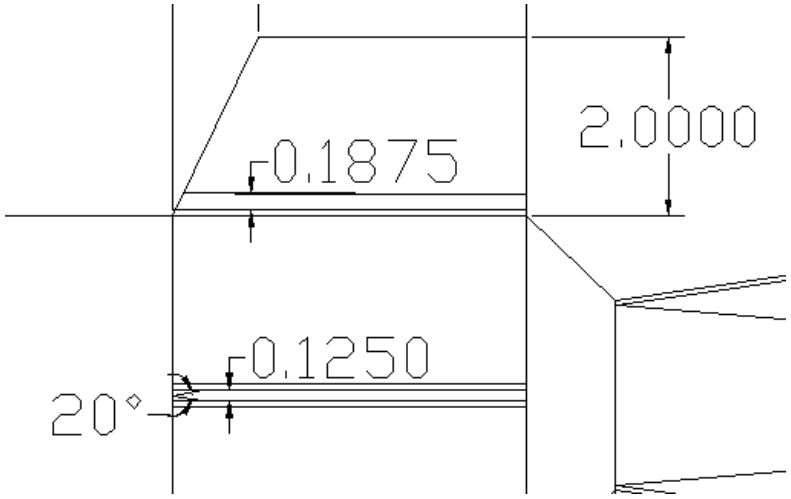

Figure 58: 4-inch dart fin detail

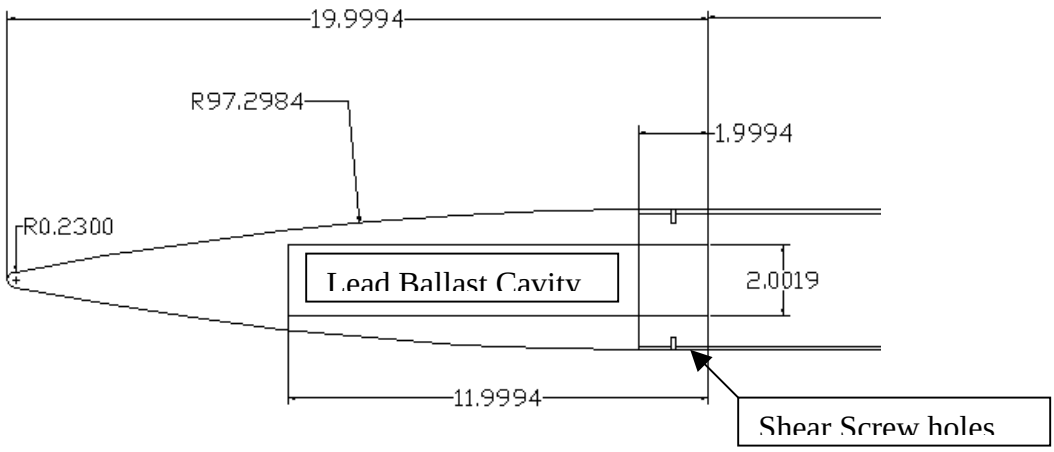

Figure 59: 4-inch dart nose detail

The 3.5 and 3-inch darts are identical to the 4-inch dart with the exception that the total fin span is kept constant to met launcher constraints, and of course the diameter. All other dimensions are simply scaled by the same factor as the outer diameter.

APPENDIX A CODE: VELOCITY ALTITUDE PRESSURE DISTRIBUTION AND GEOMETRY

Rocket input file sl2.m

```
%%%%%%%%%%Environment miscellaneous%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%As of 23-Feb-05, alpha only used for fins program%%%%%%%%%%%%%%%%
alpha=5;%deg angle of attack
alpha=alpha*pi/180;%rad
gamma=1.4;
g=-32.174;
%%%%%%%%%%Environment miscellaneous%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


%%%%%%%%%%%%%%%%%%%Nose%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Nose=1;%ogive                                  %
Ln=.93875;%ft                                  %
dn=.17708;%ft                                  %
na=10;%deg nose angle                          %
na=na*pi/180;%rad                              %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%nmod=;%if idealized conical nose
st=2;%number of stages
%%%%%%%%%%%%%%%%%%%Conical
Section%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
maxcs=1;%max number of conical sections for any config
%
CS=[1;1];%number of conical transitions
%
CS=2*CS;
%
Lcs=[4.263,4.263;4.805,4.731];%ft x location of beginning of transitions
%
dLcs=[.541667,.46875;0,0];%ft delta x conical sections
%
dcs=[.177083333333,.177083333333;.333333333,.083333333];%ft diam at beg and
end of transitions %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%Fins%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
sf=[2;1];%number of sets of fins per stage                      %
nf=[4,4;4,0];%number of fins per set                            %
xf=[3.81708,0;10.083,0];%ft leading edge x fins                 %
rf=[.17708333333,0;.3333333333,0];%ft radius at fins            %
rf=rf/2;%ft                                                     %
af=[.459167,0;.67917,0];%ft base length fins                    %
bf=[.29608,0;.16667,0];%ft tip length fins                      %
mf=[.16308,0;.5125,0];%ft base - tip fins                       %
hf=[.16308,0;.16667,0];%ft height fins                          %
dle=[10,0;10,0];%deg                                            %
tmac=[.00625,0;.00625,0];%ft thicknes at mean aerodynamic chord  %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%Motor%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%
Th=[4021,0];%Lbs thrust avg
%
```

```
isp=[232,0];%s specific impulse
%
de=[.333333,0];%ft exit diameter
%
mp=[37.6,0];%Lbs mass of propellent
%
%%%%%%%%%%%%%%%%%%Motor%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%Body%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%
Ldart=4.333;%ft length of super loki dart when attached to booster
%
Lboost=6.559;%ft length of super loki booster
%
Lb=[10.7633,4.75];%ft length of each configuration (stage)
%
Srefboost=pi/4*(dcs(2,1)^2-dn^2);%exposed area of booster with dart attached
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%Body%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%
%%%%%%%%%%%%%%%%Mass,Launch Angle,Stage Init%%%%%%%
mt=[73,18.3];%Lbs mass of each stage (full)        %
dt=.05;%s time increment                           %
la=85;%deg launch angle                            %
la=la*pi/180;%rad                                  %
stage=1;% initialize stage                         %
%%%%%%%%%%%%%%%%Mass,Launch Angle,Stage Init%%%%%%%
Mbo=0;%Mach at burnout init
'Variables have been input'
subcp;
```

Subcp.m (subsonic center of pressure)

```
'nose type 1 for ogive: 2 for parabolic: 3 for conical'
if Nosetype==1
    xn=.466*Ln;
        elseif Nosetype==2
            xn=.5*Ln;
        else
            xn=2/3*Ln;

end

if Nosetype == 3

    xn=xn-nmod;
end
Cnn=2;
st=2;
maxcs=2;
CS=[2,2];
Cncs(maxcs+1,st)=zeros;
```

```
Cnmcs(maxcs+1,st)=zeros;


for j=1:st
    i=1;
    while i<=CS(j)
        Cncs(i,j)=2*[(dcs(i+1,j)/dn)^2-(dcs(i,j)/dn)^2];
        xcs(i,j)=Lcs(i,j)+dLcs(i,j)/3*[1+(1-dcs(i,j)/dcs(i+1,j))/(1-
(dcs(i,j)/dcs(i+1,j))^2)];
        Cnmcs(i,j)=Cncs(i,j)*xcs(i,j);
    i=i+2;
    end
end

for j=1:st
    if j==1;
     for i=1:sf(j)
            'Fin set'
            aaf(i,j)=mf(i)+bf(i)/2-af(i)/2;
            lf(i,j)=sqrt(aaf(i)^2+hf(i)^2);
            dle(i,j)=dle(i,j)*pi/180;

Cnf(i,j)=(4*nf(i,j)*(hf(i,j)/dn)^2)/(1+sqrt(1+((2*lf(i,j))/(af(i,j)+bf(i,j))
)^2));
        if nf(i,j)==6
            kfb(i,j)=1+.5*rf(i,j)/(hf(i,j)+rf(i,j));
        else
            kfb(i,j)=1+rf(i,j)/(hf(i,j)+rf(i,j));
        end
        Cnf(i,j)=kfb(i,j)*Cnf(i,j);

xxf(i,j)=xf(i,j)+(mf(i,j)*(af(i,j)+2*bf(i,j)))/(3*(af(i,j)+bf(i,j)))...
            +1/6*(af(i,j)+bf(i,j)-(af(i,j)*bf(i,j)/(af(i,j)+bf(i,j))));
        Cnmf(i,j)=Cnf(i,j)*xxf(i,j);
    end
   else
       for i=1:sf(j)
            nf(i,j)=nf(i,j-1);
            xf(i,j)=xf(i,j-1);
            rf(i,j)=rf(i,j-1);
            af(i,j)=af(i,j-1);
            bf(i,j)=bf(i,j-1);
            mf(i,j)=mf(i,j-1);
            hf(i,j)=hf(i,j-1);
            dle(i,j)=dle(i,j-1);
            tmac(i,j)=tmac(i,j-1);
            aaf(i,j)=mf(i)+bf(i)/2-af(i)/2;
            lf(i,j)=sqrt(aaf(i)^2+hf(i)^2);

Cnf(i,j)=(4*nf(i)*(hf(i)/dn)^2)/(1+sqrt(1+((2*lf(i))/(af(i)+bf(i)))^2));
                if nf(i,j)==6
                    kfb(i,j)=1+.5*rf(i,j)/(hf(i,j)+rf(i,j));
                else
                    kfb(i,j)=1+rf(i,j)/(hf(i,j)+rf(i,j));
                end
```

```
            Cnf(i,j)=kfb(i,j)*Cnf(i,j);

xxf(i,j)=xf(i,j)+(mf(i,j)*(af(i,j)+2*bf(i,j)))/(3*(af(i,j)+bf(i,j)))...
                +1/6*(af(i,j)+bf(i,j)-(af(i,j)*bf(i,j)/(af(i,j)+bf(i,j))));
            Cnmf(i,j)=Cnf(i,j)*xxf(i,j);
        end
    end
end
for j=1:st
Cn(j)=Cnn+sum(Cncs(:,j))+sum(Cnf(:,j));
xcpsub(j)=(Cnn*xn+sum(Cnmcs(:,j))+sum(Cnmf(:,j)))/Cn(j);
end
Cn
xcpsub
veld;
```

Veld.m (velocity altitude calculation)

```
%This program follows subcpms program and requires variables from that
%file
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%UNITS IN FEET%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
gamma=1.4;
g=-32.174;
tb(st)=zeros;
bbbb=0;
cccc=0;
for j=1:st
    if CS(j)>0
    dnn(j)=max(dcs(:,j));
    Sref(j)=pi*((dnn(j)/2)^2);
    else
        Sref(j)=pi*(dn/2)^2;
    end
    if Th(j)>0;
        ue(j)=g*isp(j)
        Ae(j)=pi*(de(j)/2)^2;
        dm(j)=Th(j)/ue(j)*-g;
        tb(j)=-mp(j)/dm(j);
        mmod(j)=floor(tb(j)/dt);
        dm(j)=-mp(j)/mmod(j);
    else
        isp(j)=0;
        ue(j)=0;
        de(j)=0;
        Ae(j)=0;
        mp(j)=0;
        dm(j)=0;
    end
   mass(j)=mt(j);
end
%cumulative sum of burnout times
```

```
if j>1
if tb(j)>0;
tb(j)=tb(j)+tb(j-1);
else tb(j)=0;
end
end
zz=2;
time(1)=0;
Cdobw(1)=0;
Cdobs(1)=0;
Cdobf(1)=0;
Cdoww(1,1)=0;
Cdof(1)=0;
Cdof(1)=0;
time(zz)=dt;
V(zz)=zeros;
Vx(zz)=zeros;
Vy(zz)=zeros;
alt(zz)=zeros;
sps(zz)=zeros;
beta(zz)=zeros;
rho(zz)=zeros;
q(zz)=zeros;
dvy(zz)=zeros;
dvx(zz)=zeros;
rho(1)=.0024;
aa(1)=1116;
mass(1)=mass(1)+dm(1);
j=1;
for j=1:st
            for i=1:sf(j)
                    Cmac(i,j)=(af(i,j)+bf(i,j))/2;
                    Ale(i,j)=atan(hf(i,j)/mf(i,j));
                    bs(i,j)=2*(hf(i,j)+rf(i,j));%span
                    Sw(i,j)=(.5*mf(i,j)+(af(i,j)-mf(i,j)))*hf(i,j);%Fin area
            end
end
for j=1:st
    if tb(j)>0
          while time(zz)<=tb(j)

                V(zz)=-ue(j)*log(mt(j)/mass(j));
                Vx(zz)=V(zz)*cos(la);
                Vy(zz)=V(zz)*sin(la);
                alt(zz)=alt(zz-1)+Vy(zz)*dt;
                sps(zz)=-2.6359e-030*alt(zz)^6 + 8.2428e-024*alt(zz)^5-
3.785e-018*alt(zz)^4 ...
                        +4.6765e-013*alt(zz)^3+1.6709e-008*alt(zz)^2-
4.5495e-003*alt(zz)+1116.5;
                M(zz)=V(zz)/sps(zz);
                beta(zz)=sqrt(M(zz)^2-1);
                rho(zz)=.0024*exp(-4*10^(-5)*alt(zz));
                q(zz)=.5*rho(zz)*V(zz)^2;
%Body Wave drag coefficient
                    if M(zz)>1
```

```
Cdobw(zz)=(1.59+1.83/M(zz)^2)*(atan(.5/(Ln/dnn(j))))^1.69;%may need actual
nos diam dn
                    else
                        Cdobw(zz)=0;
                    end

%Body Base drag coefficient
                    if M(zz)>1
                        Cdobs(zz)=(1-Ae(j)/Sref(j))*(.25/M(zz));
                    else
                        Cdobs(zz)=(1-Ae(j)/Sref(j))*(.12+.13*M(zz)^2);
                    end
%Body Friction drag coefficient
                    Cdobf(zz)=.053*(Lb(j)/dnn(j))*(M(zz)/(q(zz)*Lb(j)))^.2;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                    % Fin Drag Coefficients
%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                    for i=1:sf(j)
                        Male(i,j)=M(zz)*cos(Ale(i,j));
                            if M(zz)>=1
                                %Cdoww(i,j)=.002;

Cdoww(i,j)=nf(i,j)*(2/(gamma*Male(i,j)^2))*sin(dle(i,j))^2*cos(Ale(i,j))*tma
c(i,j)...

*bs(i,j)/Sref(j)*((((gamma+1)*Male(i,j)^2)/2)^(gamma/(gamma-1))*...
                                        ((gamma+1)/(2*gamma*Male(i,j)^2-(gamma-
1)))^(1/(gamma-1))-1);
                            else
                                Cdoww(i,j)=0;
                            end

Cdowf(i,j)=nf(i,j)*(2*Sw(i,j)/Sref(j))*(.0133*(M(zz)/(q(zz)*Cmac(i,j)))^.2);
                    end
                    Cdow(zz)=sum(Cdoww(:,j));
                    Cdof(zz)=sum(Cdowf(:,j));

                    Cd(zz)=Cdobf(zz)+Cdobs(zz)+Cdobw(zz)+Cdow(zz)+Cdof(zz);
                    D(zz)=q(zz)*Cd(zz)*Sref(j);
                    dvx(zz)=(-D(zz)/mass(j));%new25aug
                    dvy(zz)=(-D(zz)/mass(j)+g*sin(la));%new25aug
                    Vx(zz)=Vx(zz)+dvx(zz);
                    Vy(zz)=Vy(zz)+dvy(zz);
                    V(zz)=sqrt(Vx(zz)^2+Vy(zz)^2);
                    alt(zz)=alt(zz-1)+Vy(zz)*dt+.5*dvy(zz)*dt;
                    sps(zz)=-2.6359e-030*alt(zz)^6 + 8.2428e-024*alt(zz)^5-
3.785e-018*alt(zz)^4 ...
                            +4.6765e-013*alt(zz)^3+1.6709e-008*alt(zz)^2-
4.5495e-003*alt(zz)+1116.5;
                    M(zz)=V(zz)/sps(zz);
                    beta(zz)=sqrt(M(zz)^2-1);
```

```
                    rho(zz)=.0024*exp(-4*10^(-5)*alt(zz));
                    q(zz)=.5*rho(zz)*V(zz)^2;
                    mass(j)=mass(j)+dm(j);
                    zz=zz+1;
                    time(zz)=time(zz-1)+dt;
                    cccc=cccc+1;
            end
            else
           zz=zz-1;
            while Vy(zz)>1
                zz=zz+1;
                 V(zz)=V(zz-1)+g*dt;%Tsiolkovskii Equation (No Drag)
                 Vx(zz)=V(zz)*cos(la);
                 Vy(zz)=V(zz)*sin(la);
                 alt(zz)=alt(zz-1)+Vy(zz)*dt;%+.5*dvy(zz)*dt;
                 sps(zz)=-2.6359e-030*alt(zz)^6 + 8.2428e-024*alt(zz)^5-
3.785e-018*alt(zz)^4 ...
                 +4.6765e-013*alt(zz)^3+1.6709e-008*alt(zz)^2-4.5495e-
003*alt(zz)+1116.5;
                 M(zz)=V(zz)/sps(zz);
                 beta(zz)=sqrt(M(zz)^2-1);
                 rho(zz)=.0024*exp(-4*10^(-5)*alt(zz));
                 q(zz)=.5*rho(zz)*V(zz)^2;
%Body Wave drag coefficient
                    if M(zz)>1

Cdobw(zz)=(1.59+1.83/M(zz)^2)*(atan(.5/(Ln/dnn(j))))^1.69;%may need actual
nos diam dn
                    else
                        Cdobw(zz)=0;
                    end
%Body Base drag coefficient
                    if M(zz)>1
                        Cdobs(zz)=(1-Ae(j)/Sref(j))*(.25/M(zz));
                    else
                        Cdobs(zz)=(1-Ae(j)/Sref(j))*(.12+.13*M(zz)^2);
                    end
%Body Friction drag coefficient
                    Cdobf(zz)=.053*(Lb(j)/dnn(j))*(M(zz)/(q(zz)*Lb(j)))^.2;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                    % Fin Drag Coefficients
%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                  for i=1:sf(j)
                        Male(i,j)=M(zz)*cos(Ale(i,j));
                            if M(zz)>=1

Cdoww(i,j)=nf(i,j)*(2/(gamma*Male(i,j)^2))*sin(dle(i,j))^2*cos(Ale(i,j))*tma
c(i,j)...

*bs(i,j)/Sref(j)*((((gamma+1)*Male(i,j)^2)/2)^(gamma/(gamma-1))*...
                                        ((gamma+1)/(2*gamma*Male(i,j)^2-(gamma-
1)))^(1/(gamma-1))-1);
```

```
                            else
                                Cdoww(i,j)=0;
                            end

Cdowf(i,j)=nf(i,j)*(2*Sw(i,j)/Sref(j))*(.0133*(M(zz)/(q(zz)*Cmac(i,j)))^.2);
                        end
                        Cdow(zz)=sum(Cdoww(:,j));
                        Cdof(zz)=sum(Cdowf(:,j));
                        Cd(zz)=Cdobf(zz)+Cdobs(zz)+Cdobw(zz)+Cdow(zz)+Cdof(zz);
                        D(zz)=q(zz)*Cd(zz)*Sref(j);
                        dvx(zz)=(-D(zz)/mass(j));%new25aug
                        dvy(zz)=(-D(zz)/mass(j));%new25aug
                        Vx(zz)=Vx(zz)+dvx(zz);
                        Vy(zz)=Vy(zz)+dvy(zz);
                        V(zz)=sqrt(Vx(zz)^2+Vy(zz)^2);
                        alt(zz)=alt(zz-1)+Vy(zz)*dt+.5*dvy(zz)*dt;
                        sps(zz)=-2.6359e-030*alt(zz)^6 + 8.2428e-024*alt(zz)^5-
3.785e-018*alt(zz)^4 ...
                            +4.6765e-013*alt(zz)^3+1.6709e-008*alt(zz)^2-
4.5495e-003*alt(zz)+1116.5;
                        M(zz)=V(zz)/sps(zz);
                        beta(zz)=sqrt(M(zz)^2-1);
                        rho(zz)=.0024*exp(-4*10^(-5)*alt(zz));
                        q(zz)=.5*rho(zz)*V(zz)^2;
                        time(zz)=time(zz-1)+dt;
                        bbbb=bbbb+1;
            end

        end
end
figure(1);
plot(time,M);
hold on;
plot(time,4.86);
axis([0 120 0 6]);
title('Mach Number vs Time');

figure(2);
plot(time,Cd);
axis([0 2 0 2]);
title('Cd vs Time');

figure(3);
subplot(2,2,1),plot(time,Cdobw);
axis([0 20 0 .3]);
title('Cd body wave vs Time');
subplot(2,2,2),plot(time,Cdobs);
axis([0 20 0 .1]);
title('Cd body base vs Time');
subplot(2,2,3),plot(time,Cdobf);
title('Cd body friction vs Time');
axis([0 20 0 1]);


figure(4);
```

```
subplot(2,1,1),plot(time,Cdow);
axis([0 120 0 .1]);
title('Cd fin wave vs Time');
subplot(2,1,2),plot(time,Cdof);
axis([0 120 0 2]);
title('Cd fin friction vs Time');

figure(5);
%subplot(3,1,1),plot(time,V);
%title('Velocity ft/s vs Time');
subplot(2,1,1),plot(time,Vy);
title('Y component of Velocity ft/s vs Time');
axis([0 140 0 6000]);
subplot(2,1,2),plot(time,alt);
title('Altitude ft vs Time');
rg4;
```

Rg4.m (rocket geometry)

```
dxmod=.008;
dn=dn*12;
Ln=Ln*12;
Lcs=Lcs*12;
dcs=dcs*12;
Lb=Lb*12;
dLcs=12*dLcs;
st=stage;
cmod=.001;
j=st;
x(1)=1;
%parabola%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%r(i)=dn/2*sqrt(x(i)/Ln);                                              %
%rp(i)=.5*(dn/2/(sqrt(x(i)/Ln)*Ln));                                   %
%rdp(i)=-.25*(dn/2/((x(i)/Ln)^(1.5)*Ln^2));                            %
%ogive%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%r(i)=sqrt(R^2-(x(i)-L(1))^2)+(rsm(2)-rho);                            %
%rp(i)=(-2*x(i)+2*L(1))/(2*sqrt(R^2-(x(i)-L(1))^2));                   %
%rdp(i)=-(-2*x(i)+2*L(1))^2/(4*(R^2-((x(i)-L(1))^2))^(3/2))...         %
%    -1/sqrt(R^2-((x(i)-L(1))^2));                                     %
%cone%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%r(i)=x*dn/(2*Ln);                                                     %
%rp(i)=dn/(2*Ln);                                                      %
%rdp(i)=0;                                                             %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
if Nose==1
    R=((dn/2)^2+Ln^2)/(dn);
    rp(1)=(-2*x(1)+2*Ln)/(2*sqrt(R^2-(x(1)-Ln)^2));

    r(1)=rp(1);
    rdp(1)=0;
elseif Nose==2
        rp(1)=.5*(dn/2/(sqrt(x(1)/Ln)*Ln));
        r(1)=rp(1);
        rdp(1)=0;
```

```
else
            rp(1)=dn/(2*Ln);
            r(1)=rp(1);
            rdp(1)=0;
end
MinM=sqrt(na^2*.94^2+1);%%%%%%%%%%%%%%%%%%%%Must us as trigger to start
program
zz=1;
while M(zz)<MinM
    zz=zz+1;
end
dxc=.5*sqrt(MinM^1-1);
x(2)=dxc+x(1);
dx=(Ln-x(2))/(1/dxmod-2);

    if Nose==1
        r(2)=sqrt(R^2-(x(2)-Ln)^2)+(dn/2-R);
        rp(2)=(-2*x(2)+2*Ln)/(2*sqrt(R^2-(x(2)-Ln)^2));
        rdp(2)=-(-2*x(2)+2*Ln)^2/(4*(R^2-((x(2)-Ln)^2))^(3/2))...
            -1/sqrt(R^2-((x(2)-Ln)^2));
    elseif Nose==2
        r(2)=(dn/2)*sqrt(x(2)/Ln);
        rp(2)=.5*(dn/2/(sqrt(x(2)/Ln)*Ln));
        rdp(2)=-.25*(dn/2/((x(2)/Ln)^(1.5)*Ln^2));
    else
        r(2)=x(2)*dn/(2*Ln);
        rp(2)=dn/(2*Ln);
        rdp(2)=0;
    end
if Nose==1
    for i=3:1/dxmod
        x(i)=x(i-1)+dx;
        r(i)=sqrt(R^2-(x(i)-Ln)^2)+(dn/2-R);
        rp(i)=(-2*x(i)+2*Ln)/(2*sqrt(R^2-(x(i)-Ln)^2));
        rdp(i)=-(-2*x(i)+2*Ln)^2/(4*(R^2-((x(i)-Ln)^2))^(3/2))...
                -1/sqrt(R^2-((x(i)-Ln)^2));
    end
elseif Nose==2
    for i=3:1/dxmod
        x(i)=x(i-1)+dx;
        r(i)=dn/2*sqrt(x(i)/Ln);
        rp(i)=.5*(dn/2/(sqrt(x(i)/Ln)*Ln));
        rdp(i)=-.25*(dn/2/((x(i)/Ln)^(1.5)*Ln^2));
    end
else
    for i=3:1/dxmod
        x(i)=x(i-1)+dx;
        r(i)=x(i)*dn/(2*Ln);
        rp(i)=dn/(2*Ln);
        rdp(i)=0;
    end
end
ku=2;
    if CS(j)>0%body with conical sections
```

```
                dx=(Lcs(1,j)-Ln)*dxmod;%first cylindrical section
                while x(i)<Lcs(1,j)
                    x(i+1)=x(i)+dx;
                    r(i+1)=dn/2;
                    rp(i+1)=0;
                    rdp(i+1)=0;
                    i=i+1;
                end
                x(i-1)=Lcs(1,j)-cmod;
                x(i)=Lcs(1,j)+cmod;
            while ku<=CS(j)

                if rem(ku,2)==0 %conical transition
                    dx=(Lcs(ku,j)-Lcs(ku-1,j)-2*cmod)*dxmod;
                    slope(ku,j)=(dcs(ku,j)/2-dcs(ku-1,j)/2)/(Lcs(ku,j)-
Lcs(ku-1,j));
                    r(i)=slope(ku,j)*x(i)+(dcs(ku,j)/2-
slope(ku,j)*Lcs(ku,j));
                    rp(i)=slope(ku,j);
                    while x(i)<Lcs(ku,j);
                        x(i+1)=x(i)+dx;
                        r(i+1)=slope(ku,j)*x(i+1)+(dcs(ku,j)/2-
slope(ku,j)*Lcs(ku,j));
                        rp(i+1)=slope(ku,j);
                        rdp(i+1)=0;
                        i=i+1;
                    end
                    r(i-1)=slope(ku,j)*x(i-1)+(dcs(ku,j)/2-
slope(ku,j)*Lcs(ku,j));
                    x(i)=Lcs(ku,j)+cmod;
                else%cylindrical section
                    r(i)=dcs(ku)/2;
                    rp(i)=0;
                    dx=(Lcs(ku,j)-Lcs(ku-1,j)-2*cmod)*dxmod;
                    while x(i)<Lcs(ku,j)
                        x(i+1)=x(i)+dx;
                        r(i+1)=dcs(ku)/2;
                        rp(i+1)=0;
                        rdp(i+1)=0;
                        i=i+1;
                    end
                    x(i)=Lcs(ku,j)+cmod;
                    %i=i-1;
                end
                ku=ku+1;
            end
            r(i)=dcs(CS(j),j)/2;
            rp(i)=0;
            dx=(Lb(j)-Lcs(CS(j),j))*dxmod;%last cylindrical section
            while x(i)<Lb(j)%-dx-cmod
                x(i+1)=x(i)+dx;
                r(i+1)=dcs(CS(j),j)/2;
                rp(i+1)=0;
                rdp(i+1)=0;
                i=i+1;
```

```
            end
    else% body with no conical sections
            dx=(Lb(j)-Ln)*dxmod;
            while x(i)<Lb(j)-dx-cmod
                x(i+1)=x(i)+dx;
                r(i+1)=dn/2;
                rp(i+1)=0;
                rdp(i+1)=0;
                i=i+1;
            end
            %end
end
figure(6);
plot(x,r);
axis ([0 max(x) 0 max(r)*10])
title('Rocket Geometry');
soth;
```

Soth.m (second order theory, calcs pressure and velocity at surface of body)

```
dthmod=20;
dtheta=pi/dthmod;
Corner=0;
nocorner=0;
Nocorner=0;
corner=0;
neither=0;
N=length(x);
zz=1;%dcs
while M(zz)<max(M)
    zz=zz+1;
end
spsound=sps(zz);
V=V(zz);
M=M(zz);
gamma=1.4;
beta=sqrt(M^2-1);
NN=.5*(gamma+1)*(M^2/beta^2);
phi(N,N)=zeros;
phix(N,N)=zeros;
phirbeta(N,N)=zeros;
phixx(N,N)=zeros;
cphi(N,N)=zeros;
cphix(N,N)=zeros;;
cphirbeta(N,N)=zeros;
cphixx(N,N)=zeros;
cphixrbeta(N,N)=zeros;
kphi(N,N)=zeros;
kphix(N,N)=zeros;;
kphirbeta(N,N)=zeros;
kphixx(N,N)=zeros;
kphixrbeta(N,N)=zeros;
a(N,N)=zeros;
b(N,N)=zeros;
```

```
c(N,N)=zeros;
d(N,N)=zeros;
e(N,N)=zeros;
f(N,N)=zeros;
g(N,N)=zeros;
h(N,N)=zeros;
ii(N,N)=zeros;
jj(N,N)=zeros;
k(N,N)=zeros;
l(N,N)=zeros;
m(N,N)=zeros;
ta(N)=zeros;
tt(N)=zeros;
t(N)=zeros;
phit(N)=zeros;
phixt(N)=zeros;
phirbetat(N)=zeros;
phixxt(N)=zeros;
phixrbetat(N)=zeros;
ut(N)=zeros;
vt(N)=zeros;
xcpb(N+1)=zeros;
xcpbn(N+1)=zeros;
xcpbd(N+1)=zeros;
kj(5)=zeros;
kkk(5)=zeros;
kl(5)=zeros;
km(5)=zeros;
kn(5)=zeros;
kp(5)=zeros;
kq(5)=zeros;
kr(5)=zeros;
ks(5)=zeros;
for j=1:2
    for i=1:N
    t(i,j)=beta*r(i)/x(i);
    tt(i,j)=sqrt(1-t(i,j)^2);
    a(i,j)=(.5+.25*t(i,j)^2)*asech(t(i,j))-.75*tt(i,j);%role in quad source
solution -phi/x^2 linear n/a
    b(i,j)=asech(t(i,j))-tt(i,j);%role in quad -dphidx/x linear -phi/x
    c(i,j)=.5*(tt(i,j)/t(i,j)-t(i,j)*asech(t(i,j)));%quad dphidr/bx linear
n/a
    d(i,j)=asech(t(i,j));%quad -d^2phi/dx^2 linear -dphi/dx
    e(i,j)=tt(i,j)/t(i,j);%quad d^2phi/dxdr linear dphi/dr
    f(i,j)=1/tt(i,j);%quad n/a linear -x*d^2phi/dx^2
    end
end
%linear source strength
S(1)=1/beta*rp(1)/e(1,1);
%linear source contribution
for i=1:N
    phi(i,1)=S(1)*x(i)*b(i,1);
    phix(i,1)=S(1)*d(i,1);
    phirbeta(i,1)=S(1)*e(i,1);
    phixx(i,1)=S(1)*f(i,1)/x(i);
```

```
end
phit(1)=phi(1,1);
phixt(1)=phix(1,1);
phirbetat(1)=phirbeta(1,1);
phixxt(1)=phixx(1,1);
phirr(1)=-beta^2*phixxt(1)-beta*phirbetat(1)/r(1);%27
phixr(1)= -rdp(1)+rp(1)*phirr(1);%29
psix(1)=-(-phixxt(1)*(-phit(1)+NN*r(1)*beta*phirbetat(1))-phixt(1)*(-
phixt(1)-NN*r(1)*phixr(1))...
     +.75*r(1)*phixr(1)*beta^2*phirbetat(1)^2);
psir(1)=-phixr(1)*(-phit(1)+NN*r(1)*beta*phirbetat(1))-
phixt(1)*((NN+1)*beta*phirbetat(1)+NN*r(1)*phirr(1))...
        -.25*(beta*phirbetat(1))^2*(beta*phirbetat(1)+3*r(1)*phirr(1));
psi2(1)=-phixt(1)*(-phit(1)+NN*r(1)*beta*phirbetat(1))-
.25*r(1)*(beta*phirbetat(1))^3;
phir2(1)=rp(1)/beta*(1-phixt(1));%59
ddd(1)=M^2*(phixt(1)*(NN*r(1)*beta*phirbetat(1)-
phit(1))+.25*r(1)*(beta*phirbetat(1))^3);%58%changed i's to on's
aa(1)=M^2/beta*psir(1);%57
bb(1)=M^2*psix(1);%56
ccc(1)=phir2(1)-aa(1);%60
ss(1)=ccc(1)/(rp(1)/beta);
for i=1:N
u(i,1)=ss(1)*phix(i,1);
v(i,1)=ss(1)*phirbeta(i,1);
end
ut(1)=u(1,1);
vt(1)=v(1,1);
ut(1)=bb(1)+ut(1);
vt(1)=aa(1)+vt(1);
ut(1)=1-ut(1);
%First quadratic source strength
Sa(2)=(rp(2)/beta-phirbeta(2,1));
S(2)=Sa(2)/(x(2)*c(2,2));
%First quadratic source contribution
for i=2:N
     phi(i,2)=S(2)*x(i)^2*a(i,2);
     phix(i,2)=S(2)*x(i)*b(i,2);
     phirbeta(i,2)=S(2)*x(i)*c(i,2);
     phixx(i,2)=S(2)*d(i,2);
end
            phit(2)=sum(phi(2,:));
            phixt(2)=sum(phix(2,:));
            phirbetat(2)=sum(phirbeta(2,:));
            phixxt(2)=sum(phixx(2,:));
            phirr(2)=-beta^2*phixxt(2)-beta*phirbetat(2)/r(2);%27
            phixr(2)= -rdp(2)+rp(2)*phirr(2);%29
            psix(2)=-(-phixxt(2)*(-phit(2)+NN*r(2)*beta*phirbetat(2))-
phixt(2)*(-phixt(2)-NN*r(2)*phixr(2))...
                +.75*r(2)*phixr(2)*beta^2*phirbetat(2)^2);

            psir(2)=-phixr(2)*(-phit(2)+NN*r(2)*beta*phirbetat(2))-
phixt(2)*((NN+1)*beta*phirbetat(2)+NN*r(2)*phirr(2))...
                -
.25*(beta*phirbetat(2))^2*(beta*phirbetat(2)+3*r(2)*phirr(2));
```

```
            psi2(2)=-phixt(2)*(-phit(2)+NN*r(2)*beta*phirbetat(2))-
.25*r(2)*(beta*phirbetat(2))^3;
            phir2(2)=rp(2)/beta*(1-phixt(2));%59
            ddd(2)=M^2*(phixt(2)*(NN*r(2)*beta*phirbetat(2)-
phit(2))+.25*r(2)*(beta*phirbetat(2))^3);%58%changed i's to j's
            aa(2)=M^2/beta*psir(2);%57
            bb(2)=M^2*psix(2);%56
            ccc(2)=phir2(2)-aa(2);%60
            ssa(2)=ccc(2)-v(2,1);
            ss(2)=ssa(2)/Sa(2);
for i=2:N
u(i,2)=ss(2)*phix(i,2);
v(i,2)=ss(2)*phirbeta(i,2);
end
ut(2)=u(2,2)+u(1,2);
vt(2)=v(2,2)+v(1,2);
ut(2)=bb(2)+ut(2);
vt(2)=aa(2)+vt(2);
ut(2)=1-ut(2);

%Subsequent quadratic sources
kk=1;
%for j=3:N
    j=3;
while j<=N
    if x(j)~=Lcs(kk,stage)+cmod
             Nocorner=Nocorner+1;
        for i=j:N%changed 3 to j%%
            ta(i,j)=x(i)-(x(j-1)-beta*r(j-1));
            t(i,j)=beta*r(i)/(ta(i,j));
            tt(i,j)=sqrt(1-t(i,j)^2);
            a(i,j)=(.5+.25*t(i,j)^2)*asech(t(i,j))-.75*tt(i,j);%role in quad
source solution -phi/x^2 linear n/a
            b(i,j)=asech(t(i,j))-tt(i,j);%role in quad -dphidx/x linear -
phi/x
            c(i,j)=.5*(tt(i,j)/t(i,j)-t(i,j)*asech(t(i,j)));%quad dphidr/bx
linear n/a
            d(i,j)=asech(t(i,j));%quad -d^2phi/dx^2 linear -dphi/dx
       end
    else
        Corner=Corner+1;
        for i=j:N
            if i==j
                    ta(i,j)=x(i)-(x(j)-beta*r(j));
                    cb(i)=sqrt(ta(i,j));
                    cc(i)=ta(i,j)*cb(kk);
                    t(i,j)=1.0000000000000000;
                    ll(i,j)=0.0000000000000000;
                    g(i,j)=0;
                    h(i,j)=0;
                    ii(i,j)=0;
                    jj(i,j)=1.0000;
                    k(i,j)=1.0000;
                    l(i,j)=.125;
                    m(i,j)=.375;
```

```
            else
                    ta(i,j)=x(i)-(x(j)-beta*r(j));%%%%%%check j value for
r%%%%%%%%%%%%%%
                    cb(i)=sqrt(ta(i,j));
                    cc(i)=ta(i,j)*cb(kk);
                    t(i,j)=(beta*r(i))/(ta(i,j));
                    ll(i,j)=(1-t(i,j))/(1+t(i,j));
                    [K,E]=ellipke(ll(i,j));
                    g(i,j)=.400140584959*sqrt(1+t(i,j))*((3+t(i,j))*K-4*E);
                    h(i,j)=1.80063263231*sqrt(1+t(i,j))*(K-E);
                    ii(i,j)=.600210877438*sqrt(1+t(i,j))*(1/t(i,j)*E-K);
                    jj(i,j)=.900316316157*1/sqrt(1+t(i,j))*K;

k(i,j)=.900316316157*1/sqrt(1+t(i,j))*((1+t(i,j))/t(i,j)*E-K);
                    l(i,j)=.450158158079*1/((1-t(i,j))*sqrt(1+t(i,j)))*(K-
E);
                    m(i,j)=.450158158079*1/((1-
t(i,j))*sqrt(1+t(i,j)))*(1/t(i,j)*E-K);
            end
        end
            kk=kk+1;
            if kk>CS(stage)
                kk=kk-1;
            end
    end
j=j+1;
end

kk=1;
for j=3:N
       Vvv(1)=phirbeta(j,1);
       for z=2:j-1
            Vvv(z)=Vvv(z-1)+phirbeta(j,z);
       end
       VV(j)=Vvv(z);
        Sa(j)=rp(j)/beta-VV(j);
            if x(j)==Lcs(kk,stage)+cmod
                corner=corner+1;
                S(j)=Sa(j)*cb(j);
                for i=j:N
                    cphi(i,j)=S(j)*h(i,j)*cb(i);
                    cphix(i,j)=(S(j)*jj(i,j))/cb(i);
                    cphirbeta(i,j)=(S(j)*k(i,j))/cb(i);
                    cphixx(i,j)=-(S(j)*l(i,j))/(ta(i,j)*cb(i));%took out neg
sign
                    cphixrbeta(i,j)=(S(j)*m(i,j))/(ta(i,j)*cb(i));
                end

                 kj(kk)=3*beta*phixx(j,j);
                 kkk(kk)=7*beta^2*phixx(j,j);
                 kl(kk)=phirr(j-1)-kkk(kk);
                 km(kk)=rp(j)*kl(kk);
                 kn(kk)=phixr(j-1)-kj(kk);
                 kp(kk)=rdp(j)+km(kk)-kn(kk);
                 kq(kk)=beta^2*rp(j);
```

```
                    kr(kk)=beta-kq(kk);
                    ks(kk)=(cb(j)*kp(kk))/kr(kk);

                    for i=j:N
                    kphi(i,j)=ks(kk)*ta(i,j)*cb(i)*g(i,j);%+cphi(i,j);
                    kphix(i,j)=ks(kk)*cb(i)*h(i,j);%+cphix(i,j);
                    kphirbeta(i,j)=ks(kk)*cb(i)*ii(i,j);%+cphirbeta(i,j);
                    kphixx(i,j)=(ks(kk)*jj(i,j))/cb(i);%+cphixx(i,j);

kphixrbeta(i,j)=(ks(kk)*k(i,j))/cb(i);%+cphixrbeta(i,j);

                    phi(i,j)=kphi(i,j)+cphi(i,j);
                    phix(i,j)=kphix(i,j)+cphix(i,j);
                    phirbeta(i,j)=kphirbeta(i,j)+cphirbeta(i,j);
                    phixx(i,j)=kphixx(i,j)+cphixx(i,j);
                    phixrbeta(i,j)=kphixrbeta(i,j)+cphixrbeta(i,j);
                    end
                    phit(j)=sum(phi(j,:));
                    phixt(j)=sum(phix(j,:));
                    phirbetat(j)=sum(phirbeta(j,:));
                    phixxt(j)=sum(phixx(j,:));
                    phirr(j)=-beta^2*phixxt(j)-beta*phirbetat(j)/r(j);%27
                    phixr(j)= -rdp(j)+rp(j)*phirr(j);%29

                    psix(j)=-(-phixxt(j)*(-
phit(j)+NN*r(j)*beta*phirbetat(j))-phixt(j)*(-phixt(j)-NN*r(j)*phixr(j))...
                           +.75*r(j)*phixr(j)*beta^2*phirbetat(j)^2);

                    psir(j)=-phixr(j)*(-phit(j)+NN*r(j)*beta*phirbetat(j))-
phixt(j)*((NN+1)*beta*phirbetat(j)+NN*r(j)*phirr(j))...
                           -
.25*(beta*phirbetat(j))^2*(beta*phirbetat(j)+3*r(j)*phirr(j));
                    psi2(j)=-phixt(j)*(-phit(j)+NN*r(j)*beta*phirbetat(j))-
.25*r(j)*(beta*phirbetat(j))^3;
                    phir2(j)=rp(j)/beta*(1-phixt(j));%59
                    aa(j)=M^2/beta*psir(j);%57
                    bb(j)=M^2*psix(j);%56
                    ccc(j)=phir2(j)-aa(j);%60
                    ddd(j)=M^2*(phixt(j)*(NN*r(j)*beta*phirbetat(j)-
phit(j))+.25*r(j)*(beta*phirbetat(j))^3);%58% changed i's to j's
                    srr(j)=ddd(j-1)-ddd(j);
                    ss(j)=srr(j)/cphix(j,j);
                    for i=j:N
                    suu(i,j)=ss(j)*cphixx(i,j);
                    svv(i,j)=ss(j)*cphixrbeta(i,j);
                    end
                    cmm(j)=ccc(j)-sum(v(j,:))-svv(j,j);%+svv(j,j));%changed
uu to vv
                    kss(j)=cmm(j)/Sa(j);%cm
                    for i=j:N
                    kuu(i,j)=kss(j)*kphix(i,j);
                    kvv(i,j)=kss(j)*kphirbeta(i,j);
                    end
                    css(j)=cmm(j)/kphixrbeta(j,j);
                    for i=j:N
```

```
                    cuu(i,j)=css(j)*kphixx(i,j);
                    cvv(i,j)=css(j)*kphixrbeta(i,j);
                    u(i,j)=suu(i,j)+kuu(i,j)+cuu(i,j);
                    v(i,j)=svv(i,j)+kvv(i,j)+cvv(i,j);
                    end
                    ut(j)=sum(u(j,:));
                    vt(j)=sum(v(j,:));
                    ut(j)=bb(j)+ut(j);
                    vt(j)=aa(j)+vt(j);
                    ut(j)=1-ut(j);
                   kk=kk+1;
                       if kk>CS(stage)
                          kk=kk-1;
                       end
            else
                nocorner=nocorner+1;
                Sb(j)=ta(j,j)*c(j,j);
                S(j)=Sa(j)/Sb(j);
                for i=j:N
                    phi(i,j)=S(j)*ta(i,j)^2*a(i,j);
                    phix(i,j)=S(j)*ta(i,j)*b(i,j);
                    phirbeta(i,j)=S(j)*ta(i,j)*c(i,j);
                    phixx(i,j)=S(j)*d(i,j);
                end
                                phit(j)=sum(phi(j,:));
                                phixt(j)=sum(phix(j,:));
                                phirbetat(j)=sum(phirbeta(j,:));
                                phixxt(j)=sum(phixx(j,:));
                                phirr(j)=-beta^2*phixxt(j)-
beta*phirbetat(j)/r(j);
                                phixr(j)= -rdp(j)+rp(j)*phirr(j);
                                psix(j)=-(-phixxt(j)*(-
phit(j)+NN*r(j)*beta*phirbetat(j))-phixt(j)*(-phixt(j)-NN*r(j)*phixr(j))...

+.75*r(j)*phixr(j)*beta^2*phirbetat(j)^2);
                                psir(j)=-phixr(j)*(-
phit(j)+NN*r(j)*beta*phirbetat(j))-
phixt(j)*((NN+1)*beta*phirbetat(j)+NN*r(j)*phirr(j))...
                                        -
.25*(beta*phirbetat(j))^2*(beta*phirbetat(j)+3*r(j)*phirr(j));
                                psi2(j)=-phixt(j)*(-
phit(j)+NN*r(j)*beta*phirbetat(j))-.25*r(j)*(beta*phirbetat(j))^3;
                                phir2(j)=rp(j)/beta*(1-phixt(j));%59

ddd(j)=M^2*(phixt(j)*(NN*r(j)*beta*phirbetat(j)-
phit(j))+.25*r(j)*(beta*phirbetat(j))^3);%58
                                aa(j)=M^2/beta*psir(j);%57
                                bb(j)=M^2*psix(j);%56
                                ccc(j)=phir2(j)-aa(j);%60
                                ssa(j)=ccc(j)-sum(v(j,:));
                                ss(j)=ssa(j)/Sa(j);
                                for i=j:N
                                u(i,j)=ss(j)*phix(i,j);
                                v(i,j)=ss(j)*phirbeta(i,j);
                                end
```

```
                                ut(j)=sum(u(j,:));
                                vt(j)=sum(v(j,:));
                                ut(j)=bb(j)+ut(j);
                                vt(j)=aa(j)+vt(j);
                                ut(j)=1-ut(j);
                end
 end
for i=1:N
    Vel(i)=sqrt(ut(i)^2+vt(i)^2);
end
rmax=max(r);
dCnp(1)=0;
for i=1:N
    Cp(i)=2/(gamma*M^2)*((1+(M^2*(gamma-1))/2*(1-ut(i)^2-
(beta*vt(i))^2))^(gamma/(gamma-1))-1);%removed beta*vt...
    Cp1(i)=2*(phixt(i)/Vel(i)+.5*rp(i)^2);
end
for i=N:-1:1
    Cp(i+1)=Cp(i);
    phixt(i+1)=phixt(i);
    Vel(i+1)=Vel(i);
    rp(i+1)=rp(i);
    r(i+1)=r(i);
    vt(i+1)=vt(i);
    ut(i+1)=ut(i);
    phixr(i+1)=phixr(i);
    x(i+1)=x(i);
end
Cp(1)=Cp(2)+Cp(2)*.2;
Cp(3)=(Cp(2)+Cp(4))/2;
x(1)=0;
r(1)=0;
ut(1)=0;
vt(1)=0;
phixt(1)=0;
rp(1)=1;
phixr(1)=0;
Vel(1)=.5;
Cp2(1,1)=Cp1(2)+Cp1(2)*.2;
figure(7);
subplot(2,1,1),plot(x,Cp);
title('Cp vs x');
nnn=20;
for i=1:N+1
    eta=0;
    for j=1:nnn
        Cp1(i)=(-phixt(i)/Vel(i)+.5*rp(i)^2);
        Cp2(i,j)=-.5*(-(phixr(i)*cos(eta)+alpha^2/2*(4*sin(eta)^2-1))-
1.2*Cp(i));
        eta=eta+pi/nnn;
        jjj(j)=j;
    end
    Cpp2(i)=sum(Cp2(i,:));
end
for j=1:nnn
```

```
Cp2(1,j)=Cp2(2,j)+Cp2(2,j)*.2;
end
slr=Lb(st)/dn(st);
Cmp(1)=0;
for i=2:N
Cmp(i)=-4*slr/pi*(r(i)*(x(i)/Lb(st))^2/(2*rmax)-r(i-1)*(x(i-
1)/Lb(st))^2/(2*rmax))*Cpp2(i)-...
    1/pi*((r(i)/rmax)^2*x(i)*rp(i)-(r(i-1)/rmax)^2*x(i-1)*rp(i-1))*Cpp2(i);
end
phi(1)=0;
cpb(1)=0;
lambda=Lb(stage)/max(dcs(:,stage));
rmax=max(dcs(:,stage))/2;
xcpb(1)=.01;
for i=2:N+1
    if alpha~=0
        eta(1)=0;
        for j=2:nnn

            eta(j)=eta(j-1)+pi/nnn;
            dCnpdx(i)=(-4*lambda*r(i))/(pi*rmax)*Cp(i)*(sin(eta(j)-
sin(eta(j-1))));
        end
    else

xcpbn(i)=r(i)/(2*rmax)*(x(i)/Lb(st))^2*Cp(i)+1/(4*Lb(st)/(rmax))*(r(i)/rmax)
^2*rp(i)*Cp(i)*x(i);
        xcpbd(i)=r(i)/rmax*x(i)/Lb(st)*Cp(i);
        xcpbn(i)=xcpbn(i)+xcpbn(i-1);
        xcpbd(i)=xcpbd(i)+xcpbd(i-1);
    end
end
xcpsuper=xcpbn(N+1)/xcpbd(N+1)*Lb(st)
xcpsuper=xcpsuper;

cpbody=sum(cpb);
subplot(2,1,2),plot(x,r);
title('Geometry');
axis ([0 max(x) 0 max(r)*5]);

figure(8);
plot(x,ut);
title('X velocity/M vs X position');
figure(9);
plot(x,dCnp);
title('Normal Pressure Force Coeff, vs X');

figure(10);
plot(x,Vel);
title('Vel vs x');
axis([1 max(x) min(Vel) max(Vel)]);
```

APPENDIX B CODE: SKIN TEMPERATURE CALCULATION

```
%# wrt time
lambda=5.3;%fineness ratio
Cd=1.1;
cdw=2.2;
Af=.025212;%ft^2
g=32.17;%ft/s^2
cth=.99863;%3 degrees off of vertical
dt=1;
s=121;%of seconds of flight after first 2 sec
Mass=18.3;%lb
h(s)=zeros;
v(s)=zeros;
M(s)=zeros;
a(s)=zeros;
rho(s)=zeros;

h(1)=0;
v(1)=0.0000000001;
M(1)=0;
rho(1)=0.00237717;
P(1)=14.69595;
a(1)=1116.45;
Temp(1)=518.67;

h(2)=2625;
v(2)=2137;
M(2)=1.93161;
rho(2)=.00219987;
P(2)=13.35439;
a(2)=1106.3294;
Temp(2)=509.3088;

h(3)=6562;
v(3)=5342;
M(3)=4.896;
rho(3)=0.00195313;
P(3)=11.5296;
a(3)=1090.97;
Temp(3)=495.268;
acc(s)=zeros;
%velocity and Mach as fxn of time
for i=3:s-1
    v(i+1)=v(3)*sin(1.48353)-g*i-12.1773*i;%-
((Cd*.5*rho(i)*v(i)^2*Af)/Mass)*i;
    h(i+1)=v(3)*sin(1.48353)*i-g/2*i^2-1386.69*i;
    rho(i+1)=.0028*exp(-4*10^(-5)*h(i+1));
    Temp(i+1)=4.415e-024*h(i+1)^5-2.138e-018*h(i+1)^4+1.906e-013*h(i+1)^3
...
        +3.295e-008*h(i+1)^2-4.579e-003*h(i+1)+523.1;
    P(i+1)=15.131*exp(-4*10^(-5)*h(i+1));
    a(i+1)=5.076704e-019*h(i+1)^4 - 3.871439e-013*h(i+1)^3+8.890910e-
08*h(i+1)^2 ...
    -6.837720e-003*h(i+1)+1.132801e+003;
    M(i+1)=v(i+1)/a(i+1);
end
```

```
sim=M/(lambda);%similarity parameter
for i=1:s
    Temp(i)=(a(i)/49.02)^2;
    acubed(i)=49.02^3*Temp(i)^1.5;
    mu(i)=3.733e-007*sqrt(Temp(i)/519)*((1+.505*(Temp(i)/519))/1.505);
    dTemp(i)=6.0526e-005*v(i)^2+5.917e-002*v(i)+.0000001;
    for j=1:34
        Re(i,j)=(rho(i)*v(i)*(.900233+j/12))/mu(i);
    end
end
figure(9);
plot(Temp);
title('Tempurature1');
%Calculate Skin Temp Runge-Kutta Method
Tempw(s,34)=zeros;
for j=1:34
    Tempw(1,j)=519;
end
TTempw(s,34)=zeros;
J=778;%Mechanical Equivalent of Heat(ft*lb)/BTU
ts=.027917;%Cylinder Thickness Composit Tube
w=89.856;%Specific Weight of Fiber Glass Tube lb/ft^3
em=.9;%Emissivity of the surface (Guess, Must be confirmed)
bol=4.8e-013;%Boltzmanns constant BTU/(s*ft^2*R^4)
c=.21;%Specific heat of fiberglass must be confirmed, is function of temp
G=c*ts*w;
dt=.5;%second
lam=0;
turb=0;
for i=1:s-1
    for j=1:34
        if Re(i,j)<10^6%laminar
            Pr(i,j)=7.882e-022*Tempw(i,j)^6-1.1884e-
017*Tempw(i,j)^5+7.3959e-014*Tempw(i,j)^4-2.4534e-010*Tempw(i,j)^3 ...
                +4.628e-007*Tempw(i,j)^2-4.758e-
004*Tempw(i,j)+.85935;%Prandtl number as fxn of tempw, R^2=.99282
            cfl(i,j)=.664/Re(i,j)^(.5);


F(i,j)=1/(J*G)*(1/(2*Pr(i,j)^(2/3))*((Pr(i,j)^.5*dTemp(i)+Temp(i)-
Tempw(i,j)/dTemp(i))*(cfl(i,j)*rho(i)*acubed(i)*M(i)^3)/2))- ...
                (em*bol*Tempw(i,j)^4)/G;
            q1=F(i,j)*dt;
            TTempw(i,j)=Tempw(i,j)+q1/2;
            Pr(i,j)=7.882e-022*Tempw(i,j)^6-1.1884e-
017*Tempw(i,j)^5+7.3959e-014*Tempw(i,j)^4-2.4534e-010*Tempw(i,j)^3 ...
                +4.628e-007*Tempw(i,j)^2-4.758e-
004*Tempw(i,j)+.85935;%Prandtl number as fxn of tempw, R^2=.99282
            cfl(i,j)=.664/Re(i,j)^(.5);

F(i,j)=1/(J*G)*(1/(2*Pr(i,j)^(2/3))*((Pr(i,j)^.5*dTemp(i)+Temp(i)-
TTempw(i,j)/dTemp(i))*(cfl(i,j)*rho(i)*acubed(i)*M(i)^3)/2))- ...
                (em*bol*TTempw(i,j)^4)/G;
            q2=F(i,j)*dt;
```

```
            TTempw(i,j)=Tempw(i,j)+q2/2;
             Pr(i,j)=7.882e-022*Tempw(i,j)^6-1.1884e-
017*Tempw(i,j)^5+7.3959e-014*Tempw(i,j)^4-2.4534e-010*Tempw(i,j)^3 ...
                 +4.628e-007*Tempw(i,j)^2-4.758e-
004*Tempw(i,j)+.85935;%Prandtl number as fxn of tempw, R^2=.99282
            cfl(i,j)=.664/Re(i,j)^(.5);

F(i,j)=1/(J*G)*(1/(2*Pr(i,j)^(2/3))*((Pr(i,j)^.5*dTemp(i)+Temp(i)-
TTempw(i,j)/dTemp(i))*(cfl(i,j)*rho(i)*acubed(i)*M(i)^3)/2))- ...
                 (em*bol*TTempw(i,j)^4)/G;
            q3=F(i,j)*dt;

            TTempw(i,j)=Tempw(i,j)+q3;
             Pr(i,j)=7.882e-022*Tempw(i,j)^6-1.1884e-
017*Tempw(i,j)^5+7.3959e-014*Tempw(i,j)^4-2.4534e-010*Tempw(i,j)^3 ...
                 +4.628e-007*Tempw(i,j)^2-4.758e-
004*Tempw(i,j)+.85935;%Prandtl number as fxn of tempw, R^2=.99282
            cfl(i,j)=.664/Re(i,j)^(.5);

F(i,j)=1/(J*G)*(1/(2*Pr(i,j)^(2/3))*((Pr(i,j)^.5*dTemp(i)+Temp(i)-
TTempw(i,j)/dTemp(i))*(cfl(i,j)*rho(i)*acubed(i)*M(i)^3)/2))- ...
                 (em*bol*TTempw(i,j)^4)/G;
            q4=F(i,j)*dt;

            k1=.5*(q1+q4);
            k2=q2+q3;
            q=1/3*(k1+k2);
            Tempw(i+1,j)=Tempw(i,j)+q;
            lam=lam+1;
        elseif Re(i,j)>=10^6%Turbulent
             Pr(i,j)=7.882e-022*Tempw(i,j)^6-1.1884e-
017*Tempw(i,j)^5+7.3959e-014*Tempw(i,j)^4-2.4534e-010*Tempw(i,j)^3 ...
                 +4.628e-007*Tempw(i,j)^2-4.758e-
004*Tempw(i,j)+.85935;%Prandtl number as fxn of tempw, R^2=.99282
            cft(i,j)=.0592/Re(i,j)^(1/5);

F(i,j)=1/(J*G)*(.6*((Pr(i,j)^1/3*dTemp(i)+Temp(i)+Tempw(i,j))/dTemp(i))*(cft
(i,j)*rho(i)*acubed(i)*M(i)^3)/2)- ...
                  (em*bol*Tempw(i,j)^4)/G;
              q1=F(i,j)*dt;

            TTempw(i,j)=Tempw(i,j)+q1/2;
             Pr(i,j)=7.882e-022*Tempw(i,j)^6-1.1884e-
017*Tempw(i,j)^5+7.3959e-014*Tempw(i,j)^4-2.4534e-010*Tempw(i,j)^3 ...
                 +4.628e-007*Tempw(i,j)^2-4.758e-
004*Tempw(i,j)+.85935;%Prandtl number as fxn of tempw, R^2=.99282
             cft(i,j)=.0592/Re(i,j)^(1/5);

F(i,j)=1/(J*G)*(.6*((Pr(i,j)^1/3*dTemp(i)+Temp(i)+TTempw(i,j))/dTemp(i))*(cf
t(i,j)*rho(i)*acubed(i)*M(i)^3)/2)- ...
                  (em*bol*TTempw(i,j)^4)/G;
            q2=F(i,j)*dt;

            TTempw(i,j)=Tempw(i,j)+q2/2;
```

```
            Pr(i,j)=7.882e-022*Tempw(i,j)^6-1.1884e-
017*Tempw(i,j)^5+7.3959e-014*Tempw(i,j)^4-2.4534e-010*Tempw(i,j)^3 ...
               +4.628e-007*Tempw(i,j)^2-4.758e-
004*Tempw(i,j)+.85935;%Prandtl number as fxn of tempw, R^2=.99282
            cft(i,j)=.0592/Re(i,j)^(1/5);

F(i,j)=1/(J*G)*(.6*((Pr(i,j)^1/3*dTemp(i)+Temp(i)+TTempw(i,j))/dTemp(i))*(cf
t(i,j)*rho(i)*acubed(i)*M(i)^3)/2)- ...
                (em*bol*Tempw(i,j)^4)/G;
           q3=F(i,j)*dt;

           TTempw(i,j)=Tempw(i,j)+q3;
            Pr(i,j)=7.882e-022*Tempw(i,j)^6-1.1884e-
017*Tempw(i,j)^5+7.3959e-014*Tempw(i,j)^4-2.4534e-010*Tempw(i,j)^3 ...
               +4.628e-007*Tempw(i,j)^2-4.758e-
004*Tempw(i,j)+.85935;%Prandtl number as fxn of tempw, R^2=.99282
            cft(i,j)=.0592/Re(i,j)^(1/5);

F(i,j)=1/(J*G)*(.6*((Pr(i,j)^1/3*dTemp(i)+Temp(i)+TTempw(i,j))/dTemp(i))*(cf
t(i,j)*rho(i)*acubed(i)*M(i)^3)/2)- ...
                (em*bol*Tempw(i,j)^4)/G;
           q4=F(i,j)*dt;

           k1=.5*(q1+q4);
           k2=q2+q3;
           q=1/3*(k1+k2);
           Tempw(i+1,j)=Tempw(i,j)+q;
           turb=turb+1;
        end
    end
    i
end
figure(10);
plot(dTemp);
title('DTemp');
figure(11);
surf(Re);
xlabel length;ylabel time;zlabel Re;
title('Reynolds');

figure(1);
plot(M);
title('Mach Number');
xlabel time;ylabel Mach;

figure(2);
plot(rho);
title('Density');
xlabel time;ylabel Density;

figure(3);
plot(h);
title('Altitude');
xlabel time;ylabel Altitude;
```

```
figure(4);
plot(Temp);
title('Temperature');
xlabel time;ylabel Temperature;

figure(5);
plot(acubed);
title('Speed of Sound cubed');
xlabel time;ylabel SpeedofSoundcubed;

figure(6);
plot(P);
title('Pressure');
xlabel time;ylabel Pressure;

figure(7);
plot(v);
title('Velocity');
xlabel time;ylabel Velocity;

figure(12);
surf(Tempw);
title('Wall Temperature');
shading interp
xlabel distance;ylabel time;zlabel Temperature;
```

REFERENCES

[1] Lunar Rocket and Rover Inc., 3597 Sausalito Street, Los Alamitos, CA, 90720 www.lunar-rocket.com

[2] Krasnov, N. F., "Aerodynamics of Bodies of Revolution", American Elsevier Publishing Company, Inc., New York, 1970

[3] Florida Space Authority Test Directive, 6-Dec-03, Test Number G4129, Program Manager: Pete Gunn

[4] Research Staff, NACA Report 1135, "Equations, Tables, and Charts for Compressible Flow", Ames Research Facility 1953

[5] Martin, J.J., "Atmospheric Reentry an Introduction to its Science and Engineering", Prentice Hall, Inc., Englewood Cliffs, N.J., 1966

[6] Truitt, R.W., "Fundamental of Aerodynamic Heating", The Ronald Press Company, N.Y., 1960

[7] www.matweb.com

[8] Orbital Sciences Corp. Space Data Corp., Technical Manual T.O. 11a11-28-1, Published under authority of the secretary of the Air Force

[9] www.digitaldutch.com (Standard Atmosphere Property Calculator)

[10] Milton D. Van Dyke, NACA TN-2744, "Practical Calculation of Second-Order Supersonic Flow Past Non-lifting Bodies of Revolution", 1952

[11] Ehret, Dorris M., NACA TN-2764, "Accuracy of approximate methods for predicting pressures on pointed non-lifting bodies of revolution in supersonic flow", 1952

[12] Eggers, A. H., Jr., and Savin Raymond C., NACA TN 2579, “Approximate methods for calculating the flow about non-lifting bodies of revolution at high supersonic airspeeds”, 1951

[13] Schlichting, H, “Boundary Layer Theory", Pergamon Press, New York, 1955

[14] Luther, M. L., “Comments on ‘Turbulent Boundary-Layer Characteristics…,’” , J. Aero. Sci., Vol. 19, No. 5, May 1952, p. 357.

[15] Gantmakher F. R. and Levin L. M., “ The Flight of Uncontrolled Rockets”, Pergamon Press, NY, NY

[16] Rao Makund, “Experimental Determination of Mechanical and Thermal Environment for Microstar Payloads “, UCF Press, 1996

[17] Cole, J. D., Cook, L. P., “Transonic Aerodynamics”, North Holland, Amsterdam, 1986

[18] http://www.asdl.gatech.edu

[19] Bonney, E.A., et al, *Aerodynamics, Propulsion, Structures, and Design Practice*, “Principles of Guided Missile Design”, D. Van Nostrand Company, Inc., Princeton, New Jersey, 1956

[20] Emanual, George, “Gasdynamics: Theory and Applications”, AIAA Educational Series, AIAA, New York, NY, 1986

[21] Chin, S. S., “Missile Configuration Design”, Mcgraw Hill, New York, 1961

[22] Panton Ronald, L., “Incompressible Flow”, John Wiley & Sons, New York, 1996

[23] Syverston C. A., and Dennis D. H., NACA TR 1328 “A Second Order Shock Expansion Method Applicable to Bodies of Revolution Near Zero Lift”, 1957