



5-2016

Improved Modeling of Atmospheric Entry for Meteors with Nose Radii Between 5cm and 10m

Jakob Dale Brisby

University of Tennessee - Knoxville, jbrisby@vols.utk.edu

Follow this and additional works at: https://trace.tennessee.edu/utk_gradthes



Part of the [Aerodynamics and Fluid Mechanics Commons](#), [Aeronautical Vehicles Commons](#), [Astrodynamics Commons](#), [Heat Transfer, Combustion Commons](#), and the [Space Vehicles Commons](#)

Recommended Citation

Brisby, Jakob Dale, "Improved Modeling of Atmospheric Entry for Meteors with Nose Radii Between 5cm and 10m. " Master's Thesis, University of Tennessee, 2016.
https://trace.tennessee.edu/utk_gradthes/3755

This Thesis is brought to you for free and open access by the Graduate School at TRACE: Tennessee Research and Creative Exchange. It has been accepted for inclusion in Masters Theses by an authorized administrator of TRACE: Tennessee Research and Creative Exchange. For more information, please contact trace@utk.edu.

To the Graduate Council:

I am submitting herewith a thesis written by Jakob Dale Brisby entitled "Improved Modeling of Atmospheric Entry for Meteors with Nose Radii Between 5cm and 10m." I have examined the final electronic copy of this thesis for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Master of Science, with a major in Aerospace Engineering.

James E. Lyne, Major Professor

We have read this thesis and recommend its acceptance:

Trevor M. Moeller, Zhili Zhang

Accepted for the Council:

Carolyn R. Hodges

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

Improved Modeling of Atmospheric Entry for Meteors with Nose Radii Between 5cm and 10m

A Thesis Presented for the

Master of Science

Degree

The University of Tennessee, Knoxville

Jakob Dale Brisby

May 2016

© by Jakob Dale Brisby, 2016
All Rights Reserved.

Dedication

This research is dedicated to my parents, Julie and Guy Beeckman.

Acknowledgements

I would like to thank my parents for pushing me while I was young to understand the importance of hard work, determination, and perserverance. Without these useful skills my career as a student would have long ago been cut short. Also, I'd like to thank Dr. Evans Lyne for convincing me to enter into graduate school as a student of his to expand my knowledge of Aerospace Engineering. His expert knowledge and guidance along with his assurance in me were crucial in allowing me to reach this goal of attaining a Master's Degree. I would also like to thank Dr. Trevor Moeller and Dr. Zhili Zhang for being on my thesis committee. Next, I'd like to thank the Engineering Fundamentals Department at The University of Tennessee, namely, Dr. Richard Bennett, Professor Will Schleter, and Dr. Amy Biegalski for allowing me to grow as a teacher and professional while working as a GTA. Without their understanding and assistance it would not have been possible for me to finish my studies. Lastly, I'd like to thank Samantha Murrell for being by my side throughout my seemingly infinite schooling. She was always there to lift my spirits when I would hit roadblocks in my research and convince me that giving up is never the answer.

Abstract

Atmospheric entry studies typically look closely at the peak heating rate that a body encounters during its trajectory. This is an extremely important phenomenon to study because it allows engineers to determine if a trajectory is possible with given materials and craft design specifications. It also allows designers to choose what type of method will be used for mitigating the enormous heat fluxes during entry. In general, it is accepted that during the supersonic flight regime the body will continue to be heated and an ablative heat shield often is used to deal with these heating processes. The theory outlined in this research is that with a certain set of parameters: entry velocity, nose radius, entry angle, and body characteristics, there should exist some trajectories where the entry body will begin to transfer its heat into the gas that lies between the body and the shock wave formed. To study this, a code that has been used for previous studies was adapted to look at bodies with a range of nose radii of 5cm to 10m. These trajectories were vetted against POST, a NASA developed trajectory calculator, and were determined to produce accurate flight path and velocity calculations. The heating models were also updated with curve fitted data for radiative heating and the addition of convective heating. Modifications were made to allow calculation of stagnation temperature during the late phases of the trajectory. This will allow future researchers to determine more accurately, the phase of meteor entry trajectories during which convective cooling may an important phenomenon.

Table of Contents

1	Introduction	1
1.1	Objective	1
1.2	Aerodynamic Heating	3
1.3	Implications for Man-made Spaceflight	4
1.4	Implications for Panspermia	5
2	Literature Review	6
2.1	Background	6
2.2	Fought	8
2.3	Thames	13
3	Methodology	16
3.1	Meteor Parameters	16
3.2	Limitations of Previous Model	16
3.3	Density Corrections	17
3.4	Model Validation	25
3.5	Radiative Heating Corrections	26
3.6	Convective Heating	34
4	Results	36
4.1	Discussion	36
4.2	5 cm Nose Radius	37

4.3	10 cm Nose Radius	46
4.4	50 cm Nose Radius	64
4.5	1 m Nose Radius	77
5	Conclusions	88
5.1	Conclusions	88
5.2	Future Work	89
	Bibliography	91
	Appendix	96
A	Radiative Tables	97
B	Matlab Automation	107
C	Fortran Code	111
	Vita	675

List of Tables

3.1	Common material properties of for different impactors (Chyba et al., 1993)	17
3.2	Range of applicability of the <code>qfit</code> subroutine.	33
5.1	Model range of applicability.	89
A.1	Radiative Heating Prediction at 30 km Altitude (Sutton and Hartung, 1990)	97
A.2	Radiative Heating Prediction at 36 km Altitude (Sutton and Hartung, 1990)	98
A.3	Radiative Heating Prediction at 42 km Altitude (Sutton and Hartung, 1990)	99
A.4	Radiative Heating Prediction at 48 km Altitude (Sutton and Hartung, 1990)	100
A.5	Radiative Heating Prediction at 54 km Altitude (Sutton and Hartung, 1990)	101
A.6	Radiative Heating Prediction at 60 km Altitude (Sutton and Hartung, 1990)	102
A.7	Radiative Heating Prediction at 66 km Altitude (Sutton and Hartung, 1990)	103
A.8	Radiative Heating Prediction at 72 km Altitude (Sutton and Hartung, 1990)	104

A.9 Radiative Heating Prediction at 78 km Altitude (Sutton and Hartung, 1990)	105
A.10 Radiative Heating Prediction at 84 km Altitude (Sutton and Hartung, 1990)	106

List of Figures

1.1	Convective prediction for shuttle ascent (Greenwood and Lee, 1983).	2
3.1	Discontinuity of density ratio from the ideal gas assumption.	18
3.2	Density ratios up to 22,000 ft/s (Huber, 1963).	19
3.3	Density ratios up to 46,000 ft/s (Huber, 1963).	20
3.4	Digitized density ratios up to 14,000 ft/s for various altitudes	21
3.5	Density ratios caclulated from curve fits.	23
3.6	Density ratios vs. velocity profile after updated density curve fits	24
3.7	Comparison of Altitude vs Velocity profile for a body with 1 meter nose radius entering at -15 degrees, traveling at 14 km/s with a drag coefficient of 1.2.	25
3.8	Raditative heating as a function of velocity for various nose radii at $\rho_{\infty} = 1.840E - 2 \frac{kg}{m^3}$	27
3.9	Raditative heating as a function of velocity for various nose radii at $\rho_{\infty} = 7.2579E - 3 \frac{kg}{m^3}$	27
3.10	Raditative heating as a function of velocity for various nose radii at $\rho_{\infty} = 2.9948E - 3 \frac{kg}{m^3}$	28
3.11	Raditative heating as a function of velocity for various nose radii at $\rho_{\infty} = 1.3167E - 3 \frac{kg}{m^3}$	28
3.12	Raditative heating as a function of velocity for various nose radii at $\rho_{\infty} = 6.3137E - 4 \frac{kg}{m^3}$	29

3.13	Raditative heating as a function of velocity for various nose radii at $\rho_{\infty} = 3.0592E - 4 \frac{kg}{m^3}$	29
3.14	Raditative heating as a function of velocity for various nose radii at $\rho_{\infty} = 1.4713E - 4 \frac{kg}{m^3}$	30
3.15	Raditative heating as a function of velocity for various nose radii at $\rho_{\infty} = 6.6593E - 5 \frac{kg}{m^3}$	30
3.16	Raditative heating as a function of velocity for various nose radii at $\rho_{\infty} = 2.5239E - 5 \frac{kg}{m^3}$	31
3.17	Raditative heating as a function of velocity for various nose radii at $\rho_{\infty} = 9.6940E - 6 \frac{kg}{m^3}$	31
3.18	A comparison of radiative and convective heating for a 0.1 meter body entering at -15 degrees with a velocity of 14 km/s.	35
4.1	Heating-time profile for 5 cm body entering at -8 degrees and 12 km/s.	38
4.2	Heating-time profile for 5 cm body entering at -8 degrees and 14 km/s.	38
4.3	Heating-time profile for 5 cm body entering at -8 degrees and 16 km/s.	39
4.4	Heating-time profile for 5 cm body entering at -8 degrees and 18 km/s.	39
4.5	Heating-time profile for 5 cm body entering at -15 degrees and 12 km/s.	40
4.6	Heating-time profile for 5 cm body entering at -15 degrees and 14 km/s.	40
4.7	Heating-time profile for 5 cm body entering at -15 degrees and 16 km/s.	41
4.8	Heating-time profile for 5 cm body entering at -15 degrees and 18 km/s.	41
4.9	Heating-time profile for 5 cm body entering at -30 degrees and 12 km/s.	42
4.10	Heating-time profile for 5 cm body entering at -30 degrees and 14 km/s.	42
4.11	Heating-time profile for 5 cm body entering at -30 degrees and 16 km/s.	43
4.12	Heating-time profile for 5 cm body entering at -30 degrees and 18 km/s.	43
4.13	Heating-time profile for 5 cm body entering at -45 degrees and 12 km/s.	44
4.14	Heating-time profile for 5 cm body entering at -45 degrees and 14 km/s.	44
4.15	Heating-time profile for 5 cm body entering at -45 degrees and 16 km/s.	45
4.16	Heating-time profile for 5 cm body entering at -45 degrees and 18 km/s.	45

4.17	Heating-time profile for 10 cm body entering at -8 degrees and 12 km/s.	47
4.18	Heating-time profile for 10 cm body entering at -8 degrees and 14 km/s.	47
4.19	Heating-time profile for 10 cm body entering at -8 degrees and 16 km/s.	48
4.20	Heating-time profile for 10 cm body entering at -8 degrees and 18 km/s.	48
4.21	Heating-time profile for 10 cm body entering at -15 degrees and 12 km/s.	49
4.22	Heating-time profile for 10 cm body entering at -15 degrees and 14 km/s.	49
4.23	Heating-time profile for 10 cm body entering at -15 degrees and 16 km/s.	50
4.24	Heating-time profile for 10 cm body entering at -15 degrees and 18 km/s.	50
4.25	Heating-time profile for 10 cm body entering at -30 degrees and 12 km/s.	51
4.26	Heating-time profile for 10 cm body entering at -30 degrees and 14 km/s.	51
4.27	Heating-time profile for 10 cm body entering at -30 degrees and 16 km/s.	52
4.28	Heating-time profile for 10 cm body entering at -30 degrees and 18 km/s.	52
4.29	Heating-time profile for 10 cm body entering at -45 degrees and 12 km/s.	53
4.30	Heating-time profile for 10 cm body entering at -45 degrees and 14 km/s.	53
4.31	Heating-time profile for 10 cm body entering at -45 degrees and 16 km/s.	54
4.32	Heating-time profile for 10 cm body entering at -45 degrees and 18 km/s.	54
4.33	Altitude-velocity profile for 10 cm body entering at -8 degrees.	56
4.34	Altitude-velocity profile for 10 cm body entering at -15 degrees.	56
4.35	Altitude-velocity profile for 10 cm body entering at -30 degrees.	57
4.36	Altitude-velocity profile for 10 cm body entering at -45 degrees.	57
4.37	Altitude-stagnation temperature profile for 10 cm body entering at -8 degrees.	58
4.38	Altitude-stagnation temperature profile for 10 cm body entering at -15 degrees.	58
4.39	Altitude-stagnation temperature profile for 10 cm body entering at -30 degrees.	59
4.40	Altitude-stagnation temperature profile for 10 cm body entering at -45 degrees.	59

4.41 Stagnation temperature-time profile for 10 cm body entering at -8 degrees.	60
4.42 Stagnation temperature-time profile for 10 cm body entering at -15 degrees.	60
4.43 Stagnation temperature-time profile for 10 cm body entering at -30 degrees.	61
4.44 Stagnation temperature-time profile for 10 cm body entering at -45 degrees.	61
4.45 Mass-velocity profile for 10 cm body entering at -8 degrees.	62
4.46 Mass-velocity profile for 10 cm body entering at -15 degrees.	62
4.47 Mass-velocity profile for 10 cm body entering at -30 degrees.	63
4.48 Mass-velocity profile for 10 cm body entering at -45 degrees.	63
4.49 Heating-time profile for 50 cm body entering at -8 degrees and 12 km/s. 65	
4.50 Heating-time profile for 50 cm body entering at -8 degrees and 14 km/s. 65	
4.51 Heating-time profile for 50 cm body entering at -8 degrees and 16 km/s. 66	
4.52 Heating-time profile for 50 cm body entering at -8 degrees and 18 km/s. 66	
4.53 Heating-time profile for 50 cm body entering at -15 degrees and 12 km/s. 67	
4.54 Heating-time profile for 50 cm body entering at -15 degrees and 14 km/s. 67	
4.55 Heating-time profile for 50 cm body entering at -15 degrees and 16 km/s. 68	
4.56 Heating-time profile for 50 cm body entering at -15 degrees and 18 km/s. 68	
4.57 Heating-time profile for 50 cm body entering at -30 degrees and 12 km/s. 69	
4.58 Heating-time profile for 50 cm body entering at -30 degrees and 14 km/s. 69	
4.59 Heating-time profile for 50 cm body entering at -30 degrees and 16 km/s. 70	
4.60 Heating-time profile for 50 cm body entering at -30 degrees and 18 km/s. 70	
4.61 Heating-time profile for 50 cm body entering at -45 degrees and 12 km/s. 71	
4.62 Heating-time profile for 50 cm body entering at -45 degrees and 14 km/s. 71	
4.63 Heating-time profile for 50 cm body entering at -45 degrees and 16 km/s. 72	
4.64 Heating-time profile for 50 cm body entering at -45 degrees and 18 km/s. 72	
4.65 Altitude-velocity profile for 50 cm body entering at -8 degrees.	73

4.66	Altitude-velocity profile for 50 cm body entering at -15 degrees. . . .	73
4.67	Altitude-stagnation temperature profile for 50 cm body entering at -8 degrees.	74
4.68	Altitude-stagnation temperature profile for 50 cm body entering at -15 degrees.	74
4.69	Stagnation temperature-time profile for 50 cm body entering at -8 degrees.	75
4.70	Stagnation temperature-time profile for 50 cm body entering at -15 degrees.	75
4.71	Mass-velocity profile for 50 cm body entering at -8 degrees.	76
4.72	Mass-velocity profile for 50 cm body entering at -15 degrees.	76
4.73	Heating-time profile for 1 m body entering at -8 degrees and 12 km/s.	77
4.74	Heating-time profile for 1 m body entering at -8 degrees and 12 km/s.	78
4.75	Heating-time profile for 1 m body entering at -8 degrees and 14 km/s.	78
4.76	Heating-time profile for 1 m body entering at -8 degrees and 16 km/s.	79
4.77	Heating-time profile for 1 m body entering at -8 degrees and 18 km/s.	79
4.78	Heating-time profile for 1 m body entering at -15 degrees and 12 km/s.	80
4.79	Heating-time profile for 1 m body entering at -15 degrees and 14 km/s.	80
4.80	Heating-time profile for 1 m body entering at -15 degrees and 16 km/s.	81
4.81	Heating-time profile for 1 m body entering at -15 degrees and 18 km/s.	81
4.82	Heating-time profile for 1 m body entering at -30 degrees and 12 km/s.	82
4.83	Heating-time profile for 1 m body entering at -30 degrees and 14 km/s.	82
4.84	Heating-time profile for 1 m body entering at -30 degrees and 16 km/s.	83
4.85	Heating-time profile for 1 m body entering at -30 degrees and 18 km/s.	83
4.86	Heating-time profile for 1 m body entering at -45 degrees and 12 km/s.	84
4.87	Heating-time profile for 1 m body entering at -45 degrees and 14 km/s.	84
4.88	Heating-time profile for 1 m body entering at -45 degrees and 16 km/s.	85
4.89	Heating-time profile for 1 m body entering at -45 degrees and 18 km/s.	85
4.90	Altitude-velocity profile for 1 m body entering at -8 degrees.	86

4.91	Altitude-stagnation temperature profile for 1 m body entering at -8 degrees.	86
4.92	Stagnation temperature-time profile for 1 m body entering at -8 degrees.	87
4.93	Mass-velocity profile for 1 m body entering at -8 degrees.	87

Chapter 1

Introduction

1.1 Objective

The objective of this study was to improve upon existing models of meteoroid entry to allow a better examination of the late phases of the trajectory where we suspect that high speed convective cooling is of interest because it impacts internal temperature, with implications for both man made entry bodies and panspermia. Discussion with numerous experts in the field and literature search showed there is currently very little published work on the effect or plausibility of this high speed convective cooling process. Previous studies of meteor entry trajectories have lead to the realization that under certain conditions, the hypersonic flow about a body can actually cool the surface. The mechanism behind this phenomenon is that for a certain range of bodies and entry parameters, there may be portions of the high-speed trajectory where the gas in the shock layer is at a lower temperature than the surface of the entry body. Plots of stagnation point heat transfer vs time for natural or man made entry bodies typically show a rapid growth, brief peak and rapid decline, with the calculated surface heating rate never dropping below zero. Clearly, under certain circumstances, this is incorrect and an example of this cooling phenomenon can be seen in Fig. 1.1. The solid line in Fig. 1.1 represents actual data from a sensor

located on the aft skirt of the space shuttle solid rocket boosters. During the first 90 seconds of the ascent the sensors report convective cooling and then a transition occurs to convective heating. The other lines in the figure show the predicted values for the convective heating.

Man made entry bodies could benefit from this work because it would allow scientists and engineers to plan entry trajectories to exploit this cooling so that less thermal insulation would be needed or to plan for thermal soak of electronics. The plausibility of Panspermia, the theory that life spawned somewhere else in space and traveled here by means of meteoroids or other celestial bodies, would also be impacted by this research as well. If high speed convective cooling is applied to the entry body after the peak heating, the range of body sizes and entry conditions that could transport intact organisms through Earth's atmosphere would be altered.

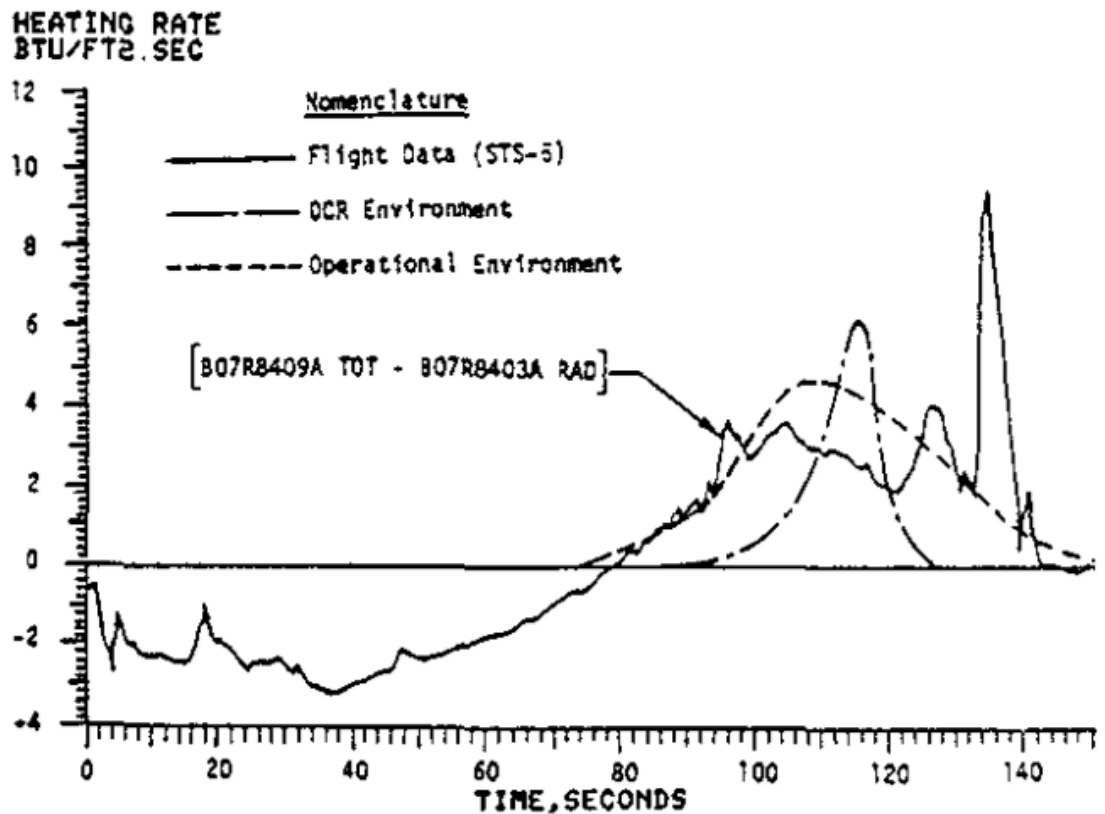


Figure 1.1: Convective prediction for shuttle ascent (Greenwood and Lee, 1983).

1.2 Aerodynamic Heating

There are two contributors to heating when dealing with atmospheric re-entry, convective and radiative heating. Convective heating is the process in which energy is transferred to an object by either advection or diffusion. Advection happens when the energy transfer occurs by the motion of a bulk fluid over the body, while diffusion arises from the random motion of the particles that make up the fluid (Bergman et al., 2011). The stagnation point convective heating is often calculated when analyzing re-entry trajectories because it can give some indication if the body can survive the estimated heat load it will encounter. This stagnation point convective heating relation is shown in Eq. 1.1 (Tauber, 213)

$$\dot{q}_0 = 10^8 \sqrt{\frac{\bar{\rho}}{r_n}} \left(\frac{V}{V_s} \right)^3 \left(1 - \frac{h_w}{h_t} \right) \quad (1.1)$$

where \dot{q}_0 is the stagnation point convective heating in $\frac{W}{m^2}$, $\bar{\rho}$ is ratio of local density to sea level density, r_n is the nose radius in meters, V is the velocity of the body in $\frac{m}{s}$, V_s is the satellite velocity of 7900 $\frac{m}{s}$, h_w is the wall enthalpy in $\frac{kJ}{kg}$, and h_t is the total enthalpy. An important relation that can be seen from the stagnation point convective heating equation is that the heat increase is inversely proportional to the square root of the nose radius, r_n . This means that as body nose radius decreases the convective heating will increase rapidly and can easily become the dominant heat source for entry bodies at relatively low hypersonic speeds.

Electromagnetic radiation is emitted by all bodies and is related to the temperature of the body. The present research is only concerned with radiation emitted as heat, or thermal radiation (Howell et al., 2015). During hypersonic flight radiative heating can easily become the largest contributor to heat by making the shock layer chemically reactive. Radiative heating is also known to increase as the velocity of the entry body increases and in many cases, become the dominating heat source for meteor entry. Air begins to dissociate between 2,000 K to 4,000 K and will be fully

dissociated when the gas temperature reaches 10,000 K. These temperatures are easily reached during the atmospheric entry (Anderson, 2000) (Thames and Lyne, 1997) (Fought, 1995) (Lyne et al., 1998). Not only do the molecules become dissociated but they can become at least singly ionized (Bauer, 1990). These ionized particles, along with an ablative heat shield, make it difficult to model exactly what is happening in the shock layer around the nose of a re-entry body.

1.3 Implications for Man-made Spaceflight

If hypersonic cooling is indeed a plausible phenomenon that occurs it could in some cases, have a significant impact on spacecraft design and mission planning. Currently, man-made vehicles employ a variety of Thermal Protection Systems (TPS). These are heat sinks, ablation, and radiative cooling. Using a heat sink is essentially placing a large amount of material in the front of the body to provide adequate storage of the heat. A major problem with this is that it can easily begin to constrain the amount of payload space and mass available. If the payload space and mass becomes too small to allow for scientific equipment then the mission would be useless. Using heat sinks along with ablation are important in the analysis of thermal soak for bodies entering an atmosphere. The material being used as the heat sink must have a low thermal conductivity so the diffusion of heat into the payload compartment is slow. Also, it must have an appropriate heat of ablation. Once the heat input to the outer layer of the body has become greater than the heat of ablation it will vaporize and shed from the body, taking the excess heat with it. If the material making up the TPS has a heat of ablation that is too low, the TPS will vaporize too quickly and not provide adequate protection to the payload. If the heat of ablation is too high, then it will not shed the excess heat, allowing the heat to diffuse through the body. To model this thermal soak, Agrawal et al. (2012) discusses an integrated tool called "Multi Mission System Analysis for Planetary Entry Descent and Landing." This model was developed to approximate the heating in the payload compartment

during an Earth entry. During a re-entry trajectory it can take long periods of time before enough heat has diffused through the body to show a rise in cabin temperature (Agrawal et al., 2012). Since it can take hours for the heat to penetrate the body, it may seem that the payload compartment heating wouldn't be an issue, but once the recovery times are factored in it becomes more apparent. It can take several hours before a re-entry body is retrieved depending on where it landed compared to its planned splash down. These long recovery times can allow for enough heat to reach the payload compartment to be damaging to scientific instruments on-board. Being able to account for the amount of high speed convective cooling that will occur during a flight could allow mission planners to take advantage of the cooling to decrease the predicted intensity of thermal soak on the entry body.

1.4 Implications for Panspermia

Panspermia is the theory that life, in the form of microbes, originated somewhere else near our celestial space and arrived here by some means. A likely scenario would be that terrain containing microbes was ejected due to some catastrophic collision with an asteroid or meteorite. The microbes would then need to survive the immense acceleration loads from the ejection, a travel through the hostile space environment, and re-entry to Earth. During its atmospheric entry to Earth, internal temperatures cannot exceed values where sterilization would occur. Therefore an understanding of the types of bodies which could transport microbes to the surface of Earth requires an ability to determine internal temperatures. According to Chyba et al. (1990) some organic material can survive temperatures around 850 K for approximately one second. If the entry body is sufficiently massive enough it may be possible to shield internal organic matter so that this sterilization temperature is not exceeded.

Chapter 2

Literature Review

2.1 Background

Many models have been created to study the physics of meteor entry. Inevitably, these models make assumptions that will allow the researchers to look at only specific characteristics or trajectories that have finely constrained parameters. One such model is that devised by Chyba et al. (1993) for the 1908 Tunguska explosion. Their model calculated the deceleration of the bolide by using Eq. 2.1

$$m \frac{dv}{dt} = \frac{1}{2} C_D \rho_a A v^2 + \frac{g}{m} \sin(\theta) \quad (2.1)$$

where ρ_a is the density of the freestream, A is the cross sectional area, g is the acceleration due to gravity, t is time, C_D is the drag coefficient, and θ is the inclination angle of the trajectory. Chyba et al. (1993) then calculated the ablated mass by using a method where he calculated the two values for the heat transfer and used the minimum. This equation can be seen as Eq. 2.2

$$Q \frac{dm}{dt} = -A \min \left(\frac{1}{2} C_H^0 \rho_a v^3, \sigma T^4 \right) \quad (2.2)$$

where Q is the heat of ablation in $\frac{\text{ergs}}{\text{g}}$, C_H^0 is heat transfer coefficient, ρ_a is the atmospheric density, v is the velocity in $\frac{\text{m}}{\text{s}}$, σ is the Stefan-Boltzmann constant, and T is the temperature. In this model, Chyba et al. (1993) mention that the heat transfer coefficient is effectively constant with a value of 0.1 above 30 km. Another assumption used is a temperature of 25,000 K everywhere in the shock layer. Which ever heat flux is the minimum is applied over the calculated area to estimate the mass lost due to ablation. The assumptions in this heating and ablative mass model are simplified and could be improved, however, Chyba et al. (1993) were able to achieve reasonable deformation and airburst altitudes consistent to those thought to occur with Meteor Crater, Tunguska, and Revelstoke objects.

There have been others models developed for looking at the ablation of meteors in planetary atmosphere. One of these models was by Ip (1990) and was developed for looking at ablation processes in Titan's atmosphere. This study uses a simple model for determining the amount of heat going into the meteor. The relation used can be seen in Eq. 2.3

$$\xi \frac{\pi}{2} R^2 \rho_a v^3 = 4\pi R^2 \epsilon \sigma (T_m^4 - T^4) + \frac{4\pi}{3} R^3 \rho_m C \frac{dT_m}{dt} + L \dot{m} \quad (2.3)$$

where ξ is the energy transfer coefficient relating the efficiency of conversion of the meteoroid kinetic energy to heating, radiation, and sublimation, ϵ is the meteoroid's emissivity, σ is the Stefan-Boltzmann constant, C is the heat capacity, T is the atmospheric temperature, T_m is the meteoroid temperature, ρ_a is the atmospheric density, ρ_m is the meteoroid density, R is the body radius, L is the heat of ablation, and \dot{m} is the mass flow rate of the the ablation products. In the above equation, the left hand side accounts for the heat being put into the body. It is a very simplistic assumption because Ip (1990) uses a constant value of 0.2 for ξ .

Another heating model that has been used to approximate the ablation of meteoroids in atmospheric entry was employed by Moses (1992) and McAuliffe and Apostolos (2005). They used Eq. 2.4 and Eq. 2.5

$$\begin{aligned} \frac{4}{3}\pi r^3 \rho_m c \frac{dT_m}{dt} &= \frac{\Lambda}{2}\pi r^2 \rho_a v^3 \\ +\pi r^2 S(1-A) - 4\pi r^2 \epsilon_{rad} \sigma T_m^4 + L \frac{dm}{dt} \end{aligned} \quad (2.4)$$

$$\Lambda = 1 - \frac{1}{2s_a^2} \left(\frac{\gamma + 1}{\gamma - 1} \right) \frac{T_m}{T_{atm}} \quad (2.5)$$

where r is the meteor nose radius, ρ_m is the meteoroid density, T_m is the temperature of the meteoroid, v is the velocity, m is the mass, A is the albedo, L is the latent heat of vaporization, c is the specific heat, ρ_a is the atmospheric density, S is the solar constant at 30 AU, ϵ_{rad} is the emissivity, σ is the Stefan-Boltzmann constant, Λ is the accommodation coefficient, s_a is equal to v/v_{th} , and γ is the ratio of specific heats. These relations account for the heat and energy being put into the meteor, as well as, the heat and energy leaving the body. The above equations account for thermal radiation, changes in internal energy, and convective heating.

2.2 Fought

The research by Fought (1995) was an initiative to devise a computer model to allow for a parametric study of Earth atmospheric entry of large meteors to determine their airburst altitude. A meteor's airburst is described as the point in the trajectory when the dynamic pressure in the shock layer surpasses a limit set by the body's material properties. This limit is accepted to be approximately twice the compressive yield strength of the material (Fought, 1995). Once this limit has been reached, the body begins to fracture and break apart. This causes the surface area to increase rapidly, which in-turn increases the drag on the debris cloud causing the conglomeration of pieces to slow down even faster. If the fracture happens violently enough it can result in a sudden loss of kinetic energy to the atmosphere. Such a

scenario is believed to have caused the Tunguska Event in Siberia in 1908 (Lyne and Tauber, 1995) (Chyba et al., 1993) (Fought et al., 1996).

The computer model developed by Fought was written in FORTRAN 77 and calculated the trajectory based on a trimmed up version of a NASA code that was used for determining re-entry profiles for space vehicles. The NASA code was a bit more robust by factoring in attitude control and body shape. These characteristics were ignored in the meteor adaptation by making simplifying assumptions about the entry body. Utilizing axi-symmetric bodies eliminated the need to calculate the body orientation and only focus on the trajectory profile. The NASA code was ideal because it took advantage of a subroutine in the IMSL library called IVPRK, which is a sixth-order Runge-Kutta algorithm. This solver was used to couple the change in velocity, flight path angle, and altitude all over time, and the equations of motion are shown in Eq. 2.6, Eq. 2.7, and Eq. 2.8, respectively. The local gravity with respect to altitude was also used and is Eq. 2.9 (Fought, 1995).

$$\frac{dV}{dt} = -g \sin \gamma - \frac{\frac{1}{2} \rho_{\infty} V^2}{C_B} \quad (2.6)$$

$$\frac{d\gamma}{dt} = - \left(\frac{g}{V} - \frac{V}{R_{earth} + H} \right) \cos \gamma \quad (2.7)$$

$$\frac{dH}{dt} = V \sin \gamma \quad (2.8)$$

$$g(H) = \frac{g_0}{\left(1 + \frac{H}{R_{earth}}\right)} \quad (2.9)$$

In the above equations, V represents the body velocity in $\frac{m}{s}$, t is time in seconds, g is local acceleration of gravity in $\frac{m}{s^2}$, γ is the flight path angle of the trajectory, ρ_{∞} is the density of the freestream in $\frac{kg}{m^3}$, C_B is the ballistic coefficient, R_{earth} is the radius of Earth in meters, H is the local altitude of the body in meters, and g_0 is the acceleration due to gravity at sea level in $\frac{m}{s^2}$. The ballistic coefficient is calculated using Eq. 2.10.

$$C_B = \frac{m}{C_D \pi r_n^2} \quad (2.10)$$

These equations are the drivers for the calculation of the entry trajectory. They are computed at each time step, based on the values of the previous time step. Some starting parameters were chosen to be invariant among the study and are the entry altitude of 100 km and a drag coefficient of 1.3. The drag coefficient of 1.3 was chosen to approximate the drag on a oblong spheroid. Parameters such as entry velocity, entry angle, nose radius, heat of ablation, and body density were varied to look at different trajectories and how they affect the entry. The purpose was to find cases which matched the known parameters of the Tunguska Event.

Fought then made calculations to determine the shape of the shock wave and the shock layer thermodynamics by curve fitting data from previous studies. Once the thermodynamics of the shock layer were known, it was possible estimate the adiabatic radiative flux using Eq. 2.11

$$Q'' = \sigma T^4 \quad (2.11)$$

where Q'' is the adiabatic radiative flux, σ is the Stefan Boltzman constant, and T is the temperature of the gas. The temperature of the gas in the shock will be variable as a function of shock angle. To account for this, the code steps around the body at 1 degree increments. At each degree the pressure is calculated using Eq. 2.12

$$P(\theta) = \rho_\infty V_\infty^2 \sin^2 \theta + P_\infty \quad (2.12)$$

where $P(\theta)$ is the pressure as a function of body angle in $\frac{N}{m^2}$, ρ_∞ is the local freestream density in $\frac{kg}{m^3}$, V_∞ is velocity of the freestream in $\frac{m}{s}$, P_∞ is the ambient pressure in $\frac{N}{m^2}$, and θ is the body angle in degrees measured from 0 to 90, with 0 being the nose of the body and 90 being the shoulder of the body.

Next, using oblique shock theory, Fought (1995) was able to derive Eq. 2.13 to calculate enthalpy as a function of body position

$$h = c_p T_\infty + \frac{1}{2} V_\infty^2 \sin^2 \beta \left[1 - \left(1 - \frac{\sin^2 \theta}{\sin^2 \beta} \right)^2 \right] \quad (2.13)$$

where c_p is the specific heat of air in $\frac{J}{kg \cdot K}$, T_∞ is temperature in K, V_∞ is the velocity of the body in $\frac{m}{s}$, θ is the body angle, and β is the angle of the shockwave. Using these two calculated thermodynamic properties, Fought (1995) used curve fits produced by Gupta et al. (1991) to calculate the temperature in the shock layer. The curves produced by Gupta et al. (1991) were only viable up to a temperature of 30,000 K for pressures between 10^{-4} and 10^2 atmospheres. This proved to be inadequate for Fought's model and required a scheme for extrapolating data for pressures up to 1,000 atm (Fought et al., 1997). After calculating the temperature in the shock layer it was important to add in correction factors to account for the cooling of the gas due to radiating its energy to the surrounding atmosphere. To account for this Goulard (1964) was able to devise Eq. 2.14. The Goulard number is then used in Eq. 2.15 to determine a new non-adiabatic flux devised by Tauber (1969)

$$\Gamma = \frac{Q''_{ad}}{4\rho_\infty V_\infty^3} \quad (2.14)$$

$$Q''_G = \frac{Q''_{ad}}{1 + 3.33\Gamma} \quad (2.15)$$

where Γ is the Goulard number, Q''_{ad} is the adiabatic radiative flux, and Q''_G is the corrected radiative flux neglecting radiative energy loss from the shock layer. By calculating the heat flux at one degree increments around the body at each time step it is possible to approximate the total heating to which the body is subjected. This was a significant improvement over the Chyba et al. (1993) assumption that the temperature everywhere in the shock layer is 25,000 K.

The radiation blockage was the next step in Fought's calculation of meteor entries. This was an important consideration because the blockage can have a significant effect on the amount of radiative heating absorbed by the body. This radiation blockage happens once the heat input to the body reaches and exceeds the heat of ablation for the material. Once this occurs the material will begin to vaporize and be stripped away. This will occur initially at the nose of the entry body and can cause deformation

and a change in the aerodynamic characteristics of the body. The body deformation is ignored to simplify the calculations and should be an area of focus for further studies. The material that gets vaporized and injected into the flow causes the blockage effect. The material is blown into the shock layer, and as it is transported along the body it acts as a shield to absorb the radiant heat before it reaches the entry body (Fought et al., 1996). The blockage is accounted for in the code by using Eq. 2.16 which was formulated from curve fits from a study done by Gupta et al. (1994).

$$\frac{\dot{m}_{ablation}}{\dot{m}_{freestream}} = \frac{Q''_{body}}{H_{ablation}\rho_{\infty}V_{\infty}} \quad (2.16)$$

The radiative heat flux into the body is limited to 15% based on observations by (Lyne and Tauber, 1994). Once the total heat flux for the time step is determined the change in mass is found assuming the body did not break, and a new radius is found for the next iteration in the time series assuming that the body remains sperical. Convective heating was neglected in Fought's research because he was looking for disruption which typically occurs with larger, faster bodies for which the trajectory is dominated by radiative heating. Using Eq. 2.17 it was determined that convection accounts for less than 2% of the total heating for a 67 meter stone (Fought, 1995).

$$Q''_{conv,stag} = .000183 \left(\frac{\rho_{\infty}}{r_{nose}} \right)^{\frac{1}{2}} V_{\infty}^3 \quad (2.17)$$

The disruption of the body is determined by the amount of dynamic pressure acting on the body. Chyba et al. (1993) developed a model that was used in Fought's research for disruption. The average pressure acting on the front face of the body was calculated using Eq. 2.18, while assuming there is no pressure acting on the rear side of the body. This allows for the assumption that the midpoint of the body has a pressure equal to half of the pressure acting on the face, giving rise to the limit of

twice the yield strength before disruption happens.

$$P_{avg} = \frac{1}{2}C_D\rho_\infty V_\infty^2 \quad (2.18)$$

If a disruption happens then Eq. 2.19 can be used to approximate the dispersion of the body. This model by Chyba et al. (1993) also points out that when disruption happens, the body will begin to flatten out, increasing its surface area and deposit its energy, resulting in an airburst.

$$r \frac{d^2r}{dt^2} = \frac{C_D\rho_\infty V_\infty^2}{2\rho_{mS}} \quad (2.19)$$

This study does not allow for reliable calculations for entries of greater than 19 km/s or those that generate shock layer temperatures in excess of 30,000 K. For trajectories in the envelope of the study, the model does produce reasonable results. Fought (1995) mentions that his model does produce airbursts at lower altitude than previous models, but this could be due to the more detailed estimate of the shock layer physics.

2.3 Thames

The research done by Thames (1997) set out to improve upon the model that has just been described. The thermodynamic data used by Fought (1995) limited the temperature range of air, and therefore the entry velocity that could be considered. To improve the temperature range for which the model was applicable, Thames applied the data from a Chance Vought report to allow for an extension of allowable temperatures, pressures, and velocities (Sewell, 1961). By incorporating the Chance Vought data, Thames was able to increase the allowable temperature from about 30,000 K to nearly 100,000 K for pressures between .005 atm and 1,000 atm. It will also allow for a more limited temperature range for pressures between .00002 atm and

100,000 atm (Thames, 1997). According to Chyba (1991) this allows for the model to be suitable for most Earth-crossing bodies by increasing the valid entry velocity to 50 km/s. Other improvements by Thames included updating the FORTRAN 77 code to FORTRAN 90, using more accurate calculation methods of determining the pressure and enthalpy across the shock, using a new differential equation solver, and improving the Green-Nicolet blowing factor (Green and Nicolet, 1978). The differential equation solver was replaced from the previous version in an attempt to move away from a machine dependent code. The differential equation solver that replaced the IMSL from Fought's research is the DVERK subroutine. The DVERK routine is a fifth and sixth order Runge-Kutta solver based on Vernor's method to approximate the solution to a system of first order differential equations.

The biggest gain from Thames' study was the incorporation of the Chance Vought report (Sewell, 1961). This report which has an unknown author was an internal report at the Chance Vought Research Center and was published in 1961. The group at the research center recognized the need to develop more accurate and robust calculations of high temperature thermodynamics and accomplished this by looking at certain thermodynamic properties of gas, such as temperature, pressure, density, enthalpy, entropy, molar concentration, speed of sound, specific heat at constant pressure, and specific heat at constant volume. Another reason the Chance Vought report is important, is they included argon into their calculations when aerodynamicists generally included nitrogen and oxygen. This led to a large number of differential equations to be used in calculating the data for the report. The result of this research was a set of graphs that could be used to determine the thermodynamic properties of air for a wide range of temperatures and pressures (Thames, 1997).

To make the Chance Vought curves useful, Thames devised a curve fitting solution by digitizing 1,500 data points from the graphs and using a bivariate interpolation (ITPLBV) routine developed by Akima (1974). To setup the model to use the ITPLBV routine, Thames developed two other subroutines in the FORTRAN model. The first is the `TEMP_FIT` model which interpolates the temperature from a given

pressure and enthalpy. The second was DENS_FIT which determined the density when given pressure and temperature. These subroutines were added to the existing code and meant that there were now two different subroutines to calculate the temperature at each times step. Since there were now multiple subroutines to estimate the temperature of the gas, it was necessary to add in conditional statments to the model so the correct subroutine could be chosen. If the enthalpy of the gas was above a limit of $7.44580512 \frac{MJ}{kg}$, then the model would use Thames' routine, and below that value it would use Fought's (Thames, 1997). The $7.44580512 \frac{MJ}{kg}$ limit corresponds the lower end of the Chance Vought graphs. Thames' routine does not allow for extrapolation of the Chance Vought graph because the relationships that are represented there are not linear and would given inaccurate results. Another limitation of Thames' fits is that it can only interpolate the temperature above 3,000 K. For temperatures below this, normal shock equations are used assuming an ideal gas.

The Green-Nicolet approximation was another physics parameter that needed adjusting to add fidelity to the model. Fought's model used set ratios of blockage based on certain conditions. To correct for this assumption Thames' was able to fit a fifth-order polynomial, Eq. 2.20 to the Green-Nicolet Approximation of the scale factor for blockage mass.

$$1.0095555 - .11552317 \left(\frac{\dot{m}}{\rho_1 V} \right) - 1.7206431 \left(\frac{\dot{m}}{\rho_1 V} \right)^2 - .53616033 \left(\frac{\dot{m}}{\rho_1 V} \right)^3 + 4.1602124 \left(\frac{\dot{m}}{\rho_1 V} \right)^4 - 2.2932833 \left(\frac{\dot{m}}{\rho_1 V} \right)^5 \quad (2.20)$$

Thames' code adpatations summarized here were critical for increasing the velocity range of applicability to the model. Although there are still areas of improvement for the code, he was able to achieve reasonable results to backup the calculations used.

Chapter 3

Methodology

3.1 Meteor Parameters

To properly develop the numerical model, it was important to determine an appropriate range of entry parameters. The entry altitude for every scenario was kept constant at 100 km while a range of entry angles, velocities, and body radii were used. The angles that were studied were -8, -15, -30, and -45 degrees. The entry angle of -45 degrees was used because it is the most common angle of impact (Kopal, 1971) and -8 degrees was chosen as a shallow ballistic entry. The range of velocities that were chosen were 12, 14, 16, and 18 km/s because these are within the range of the radiative heating tables calculated by Sutton and Hartung (1990). The drag coefficient for the entry bodies are set to a static value of 1.2. The drag coefficient for a sphere at high speed should be near to 1, but 1.2 was chosen to account for some irregularity in the shape. The material chosen for this study was stone, and the material properties can be seen in Table 3.1.

3.2 Limitations of Previous Model

The model developed by Fought (1995) and Thames (1997) was focused on peak radiative heating and the disruption event (airburst altitude). The trajectories that

Table 3.1: Common material properties of for different impactors (Chyba et al., 1993)

Meteor Material Properties			
Material	Density ($\frac{kg}{m^3}$)	Strength ($\frac{N}{m^2}$)	Heat of Ablation ($\frac{J}{kg}$)
Iron	7,900	1.0E8	8.0E6
Stone	3,500	1.0E7	8.0E6
Carbonaceous	2,200	1.0E5	2.5E6
Comet	1,000	1.0E5	2.5E6

were studied were characterized by high entry velocities and large body radii. Curve fits from Fought (1995) are used if the shock layer temperature is in excess of 3,000 K and if the enthalpy of the shock layer is above $7.44580512 \frac{MJ}{kg}$ then the curve fits from Thames (1997) are used. If the properties of the trajectory do not allow for the curve fits from Fought (1995) or Thames (1997), the code used normal shock equations and an assumption that the gas was ideal. Lastly, since the previous model was concerned with a parametric study of airbursts, none of the evaluated bodies slowed down enough to get out of the range where the curve fits were valid. This allowed for relatively low speed portions of the trajectory calculations to be ignored. In the current study, the code was modified to allow simulation of later phases of the entry, in order to determine if and when convective cooling is an important consideration.

3.3 Density Corrections

When simulating the trajectories of smaller bodies, the first short-coming to be fixed was the density calculation. Once the temperature or enthalpy reached values lower than allowed by the previous curve fits, the ideal gas assumption was used to determine density ratio across the shock. This caused noticeable inaccuracies in the data for the trajectories of small bodies. By looking at a plot of density ratio across the shock vs. velocity it was clear that the ideal gas assumption was being applied too early in the trajectory. This discontinuity can be seen in Fig. 3.1. As previously

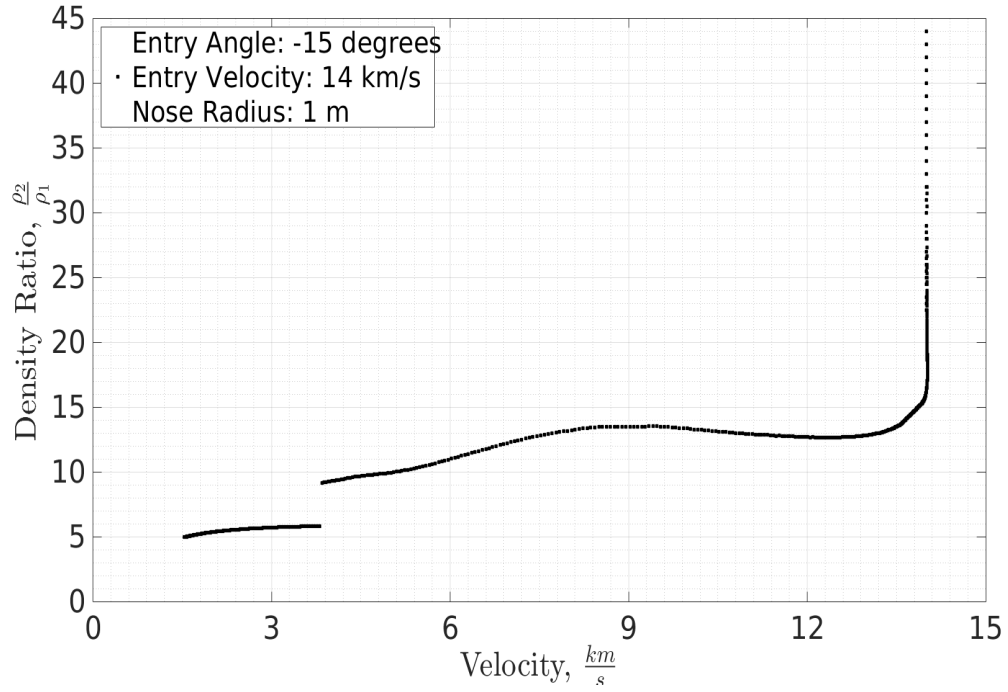


Figure 3.1: Discontinuity of density ratio from the ideal gas assumption.

mentioned, this was not a concern for the trajectories of the large bodies for which the code was developed, since they never reached this flight regime.

The trajectory in Fig. 3.1 is representative of all bodies that dipped below the bounds of the curve fits, so it was obvious this needed to be fixed in order to yield usable results for smaller meteors. A set of curve fits were constructed to more accurately calculate the density ratio across the shock wave for trajectories that would not allow the use of the routines developed by Fought (1995) and Thames (1997). The curve fits to more accurately calculate the density ratio across the shock were pulled from two data sources. The majority of this data came from Huber (1963) and the remaining from Hochstim (1963). The work done by Huber (1963) and Hochstim (1963) allowed for the density ratio to be calculated at multiple altitudes and then linearly interpolated to approximate the density in the shock layer. Figures 3.2 and 3.3 are plots produced by Huber (1963) showing the density ratio as plotted against velocity in feet per second. The lines of the graphs represent constant altitude for

which there are 12. The altitudes range from 35,900 feet, at the lowest, to 322,900, feet, with varying increments making up the intermediate altitudes. The domain of the velocity lies between 3,697 feet per second and 46,000 feet per second. They have been included to allow for quick reference.

Although some data points were given in the report by Huber (1963), it seemed advantageous to digitize the plot to get more data points. Using MATLAB it was possible to digitize more than 1,100 points to represent the curves. This process was not exact and did not result in smooth lines but allowed for best fit lines to be generated using MATLAB's Curve Fitting Toolbox. The first iteration of curve fits proved to give poor quality because the range of the fits were too large. By running various trajectory scenarios it was revealed that the routines from Fought (1995) and Thames (1997) worked for entry velocities down to around 10 km/s for bodies with a nose radius greater than 10 m, but for bodies in the 5 centimeters to 1 meter range, the velocity would slow enough to have stagnation temperatures that drop below the

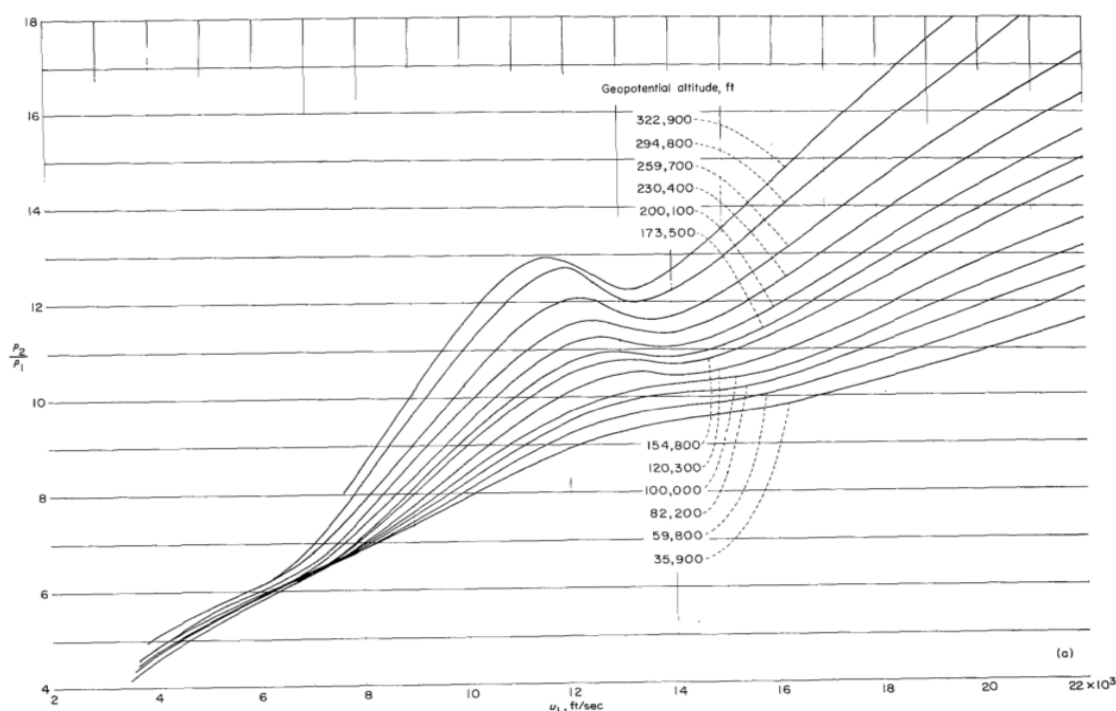


Figure 3.2: Density ratios up to 22,000 ft/s (Huber, 1963).

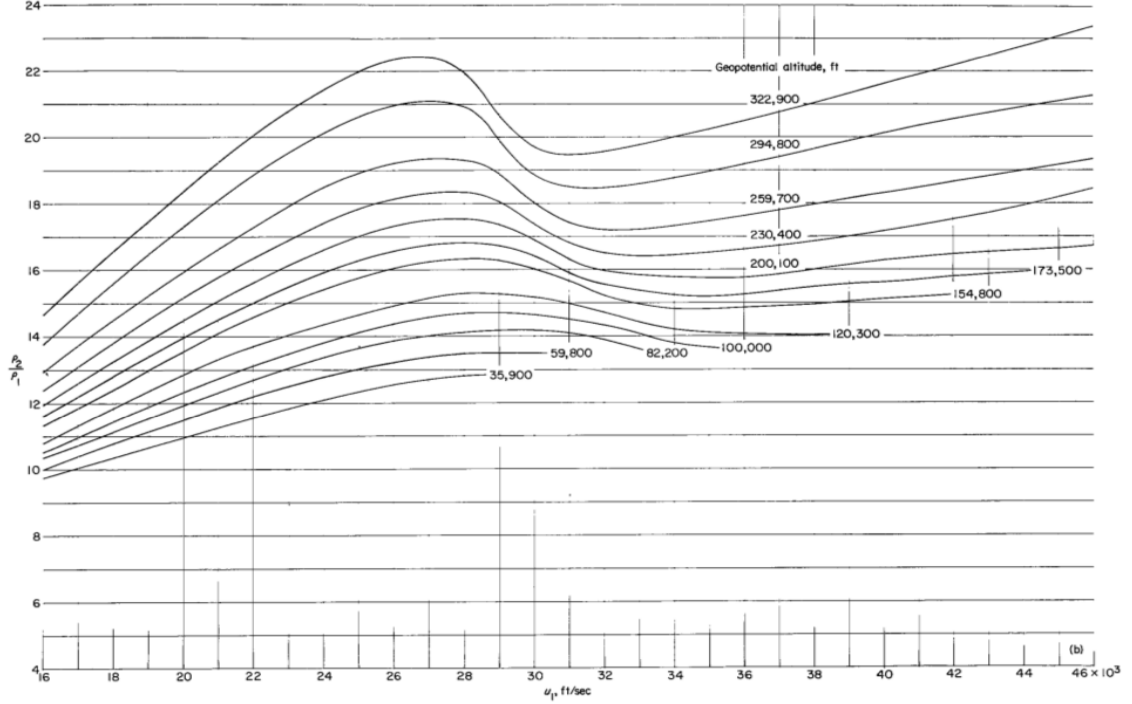


Figure 3.3: Density ratios up to 46,000 ft/s (Huber, 1963).

range for which they could be applied. Noticing this allowed the author to curve fit a much smaller range of the digitized data. The smaller data set required fewer equations and allowed for better fits of each data series. From looking at Fig. 3.2 and 3.3, an inflection point is observed for most data series at 14,000 ft/s or lower. This 14,000 ft/s velocity now became the top end of the data to be fit. The digitized data for this portion of the Huber (1963) curves can be seen in Fig. 3.4.

Each curve was broken up into two fitted equations. Different forms of equations were also explored in order to generate the best fit. The following equations were generated using MATLAB: 1st order Fourier Eq. 3.1, 2nd order Fourier Eq. 3.2, 3rd order Fourier Eq. 3.3, 4th order Fourier Eq. 3.4, 2nd order Gaussian Eq. 3.5, and 4th order Gaussian Eq. 3.6.

$$f(x) = a_0 + a_1 \cos(xw) + b_1 \sin(xw) \quad (3.1)$$

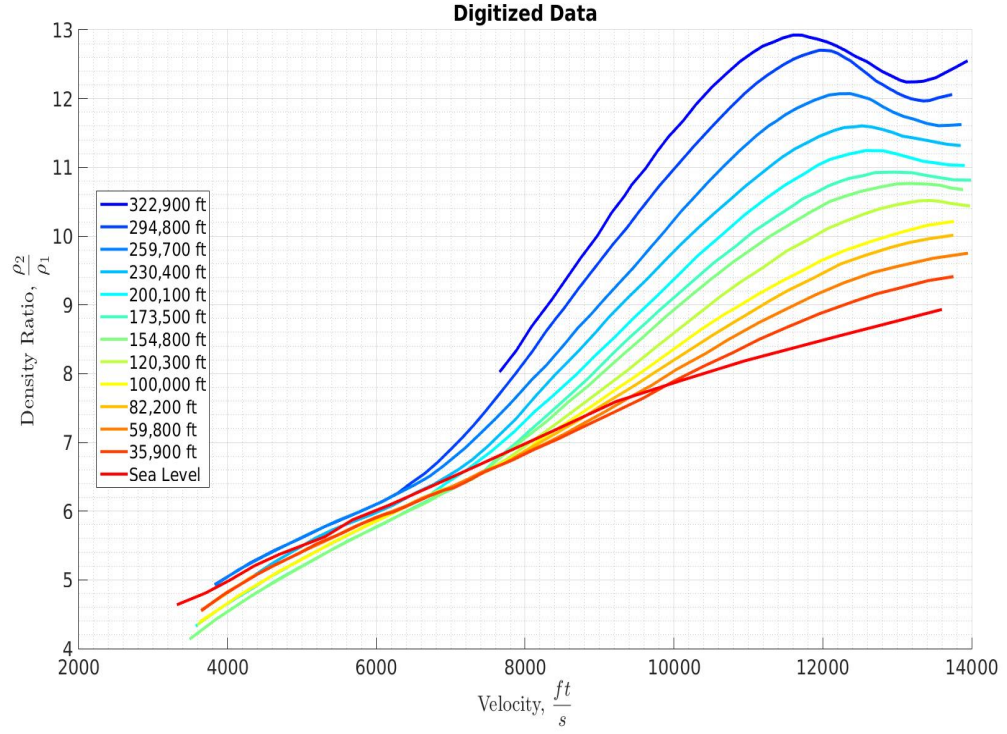


Figure 3.4: Digitized density ratios up to 14,000 ft/s for various altitudes

$$f(x) = a_0 + a_1 \cos(xw) + b_1 \sin(xw) + a_2 \cos(2xw) + b_2 \sin(2xw) \quad (3.2)$$

$$f(x) = a_0 + a_1 \cos(xw) + b_1 \sin(xw) + a_2 \cos(2xw) + b_2 \sin(2xw) + a_3 \cos(3xw) + b_3 \sin(3xw) \quad (3.3)$$

$$f(x) = a_0 + a_1 \cos(xw) + b_1 \sin(xw) + a_2 \cos(2xw) + b_2 \sin(2xw) + a_3 \cos(3xw) + b_3 \sin(3xw) + a_4 \cos(4xw) + b_4 \sin(4xw) \quad (3.4)$$

$$f(x) = a_1 e^{-\left(\frac{x-b_1}{c_1}\right)^2} + a_2 e^{-\left(\frac{x-b_2}{c_2}\right)^2} \quad (3.5)$$

$$f(x) = a_1 e^{-\left(\frac{x-b_1}{c_1}\right)^2} + a_2 e^{-\left(\frac{x-b_2}{c_2}\right)^2} + a_3 e^{-\left(\frac{x-b_3}{c_3}\right)^2} + a_4 e^{-\left(\frac{x-b_4}{c_4}\right)^2} \quad (3.6)$$

After creating the equations they were evaluated for velocities between 4,000 ft/s and 14,000 ft/s to see how well they follow the digitized data and are reported in Fig. 3.5. The first thing one may noticed is how the left end of the curve for 322,900 ft begins to rise again below 6,000 ft/s. This is due to the fact that the Huber (1963) paper has data down to only 7,600 ft/s so below that, the data is being extrapolated. This hasn't become a concern with the current study because the entry velocities are high enough to keep the body well within the range of applicability for high altitudes. Another item worth mentioning is that in almost all the curves, there appears to be a small discontinuity at 9,000 ft/s. This is the velocity value in which the curves were split into two equations, causing a small misalignment (above 9,000 ft/s the curve is represented by one equation, and below 9,000 ft/s a different equation is used). This discontinuity is small compared to the original discontinuity and should not yield wildly inaccurate density calculations.

After validating the curve fits for their accuracy, they were formatted and put into a subroutine `DENS_FIT3`. Not only does this subroutine contain the equations that were fitted for the data, it also converts the units from metric to English for calculation, then back to metric; it also contains the algorithm used for interpolating. A simple linear interpolation of density ratios is employed after calculating the density ratio for a given velocity at known altitudes for values above and below the given conditions. For instance, assume an entry body has an altitude and velocity of 60,000 ft and 8,000 ft/s, respectively. The `DENS_FIT3` routine will first determine if the density ratio will be calculated by the equation that represents the data below 9,000 ft/s. Then, using a series of conditional statements, the subroutine will calculate the density ratios at altitudes of 82,200 ft and 59,800 ft at the given velocity. Then a weight sum is computed to approximate the density ratio based on the difference in

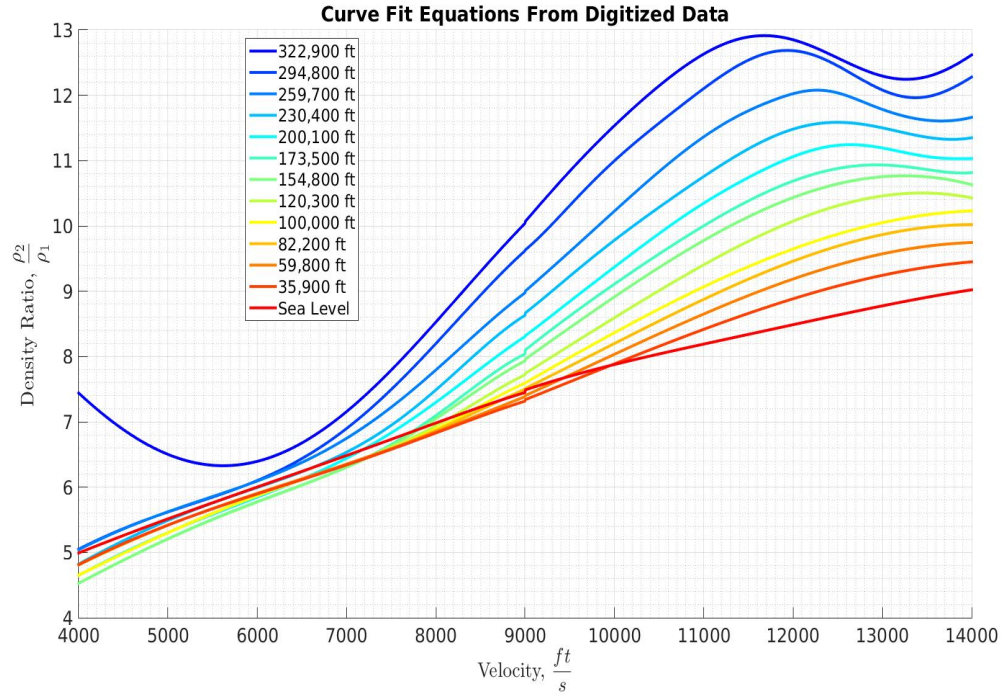


Figure 3.5: Density ratios calculated from curve fits.

altitude. The new density ratio is then returned to the main program where it can be used in subsequent calculations in the simulation.

The DENS_FIT3 subroutine is the last in the series of curve fits to be considered. The conditions for entering the subroutine are the velocity must be below 13,000 ft/s, must have a Mach number greater 1, and must have an enthalpy of less than 7.44580512 MJ/kg. Though the curve fits were computed for an upper limit of 14,000 ft/s, a slightly lower value of 13,000 ft/s was used in the routine to make sure the values were well within the range of applicability. Using this subroutine, a smoother transition to the normal shock relations was made to increase the range of applicability for the model. This has allowed the model to be extended to velocities near Mach 1 where temperatures in the shock layer may not be high enough to meet the conditions of the previous model.

Since the original model was not setup to account for slower moving bodies, the conditions for terminating the program had to be changed. The original ending

conditions for FORTRAN code were if the altitude was less than 1 meter or if the body had lost 99% of its original kinetic energy. There were two changes to the terminating conditions of the model. The first was that the kinetic energy was allowed to reach .0001% of its original before terminating, allowing for lower velocities to be considered. The second was an upper bound applied to the altitude to terminate if a body bounced back into orbit. Changing these terminating conditions also made it necessary to add in a calculation that was valid for the subsonic portion of the trajectory. To account for this the density ratio was just set to one so the model would use the freestream density to calculate the thermodynamic properties. Figure 3.6 shows a much smoother density ratio vs. velocity profile. The discontinuity that occurs just before ending is due to the velocity going subsonic just before the altitude goes to zero.

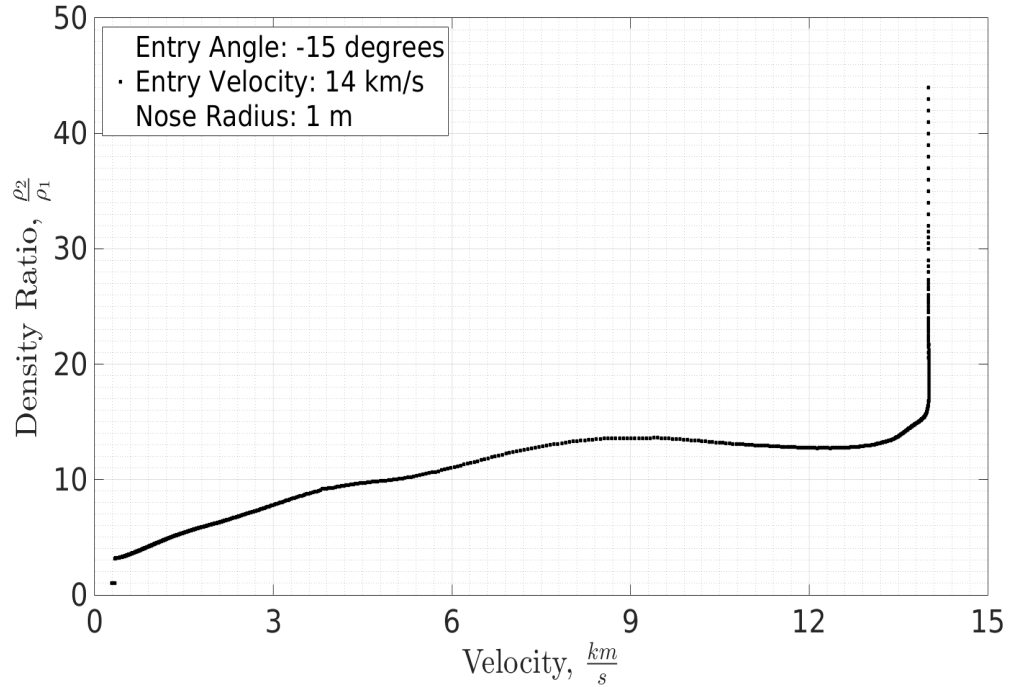


Figure 3.6: Density ratios vs. velocity profile after updated density curve fits

3.4 Model Validation

To validate our trajectory model, the results were compared to The Program to Optimize Simulated Trajectories(POST) (Brauer et al., 1989). POST was developed by NASA starting in 1970s as a way to produce accurate and reliable calculations for atmospheric entry trajectories. The input file used for the POST calculations included the entry velocity, nose radius, entry altitude, mass, drag coefficient, and entry angle, and the body was assumed to be a non-ablating, hard sphere. This required entering in parameters to the meteor code that would not allow the aerodynamic heating to surpass the heat of ablation for the entry body material and assigning a yield strength to the material that would not be exceeded by the pressures behind the shockwave. Doing this allowed for an accurate comparison of the two codes. It was determined that the calculated trajectory from this study fit reasonably well with the POST calculated data, and a comparison of trajectory data can be seen in Fig. 3.7.

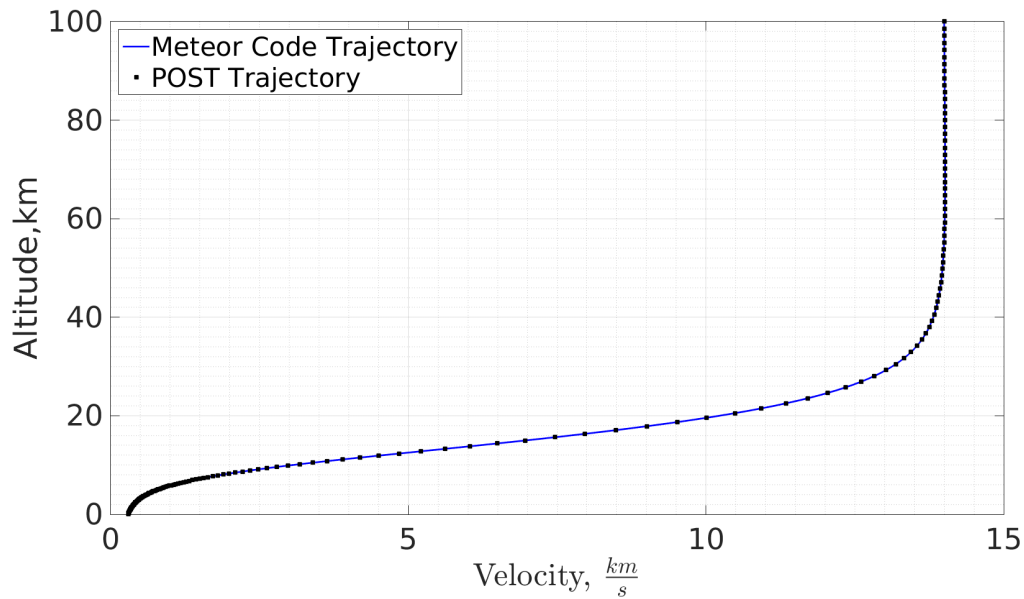


Figure 3.7: Comparison of Altitude vs Velocity profile for a body with 1 meter nose radius entering at -15 degrees, traveling at 14 km/s with a drag coefficient of 1.2.

3.5 Radiative Heating Corrections

The previous method for calculating radiative heating was appropriate for large bodies under the assumption that the shock layer is optically thick and the emissivity was assumed to be 1. This assumption breaks down for small bodies because the chord length of the body shrinks in relation to the mean free path of radiation (Pai, 2012). If this is not taken into account, the methodology of Fought (1995) and Thames (1997) grossly overpredicts radiative heating for small bodies.

To correct the radiative heating calculations for small bodies, another set of curve fits was completed using tables produced by Kenneth Sutton and Lin C. Hartung from NASA. The method used was a radiative, inviscid flow, stagnation point code which was developed by Sutton (1973). To test his code, a matrix of entry parameters was constructed and validated against ground based and flight data (Sutton and Hartung, 1990). The data that has been tabulated by Sutton and Hartung show the estimated radiative heating in $\frac{MW}{m^2}$ and shock stand-off distance for a number of entry bodies at differing velocities ranging from 8,000 m/s to 18,000 m/s and densities from $1.840E-2 \frac{kg}{m^3}$ to $9.6940E-6 \frac{kg}{m^3}$. These freestream densities correspond to altitudes of 30 km and 84 km respectively. The study looked at densities for altitudes at 6 km increments. These data can be seen in Fig. 3.8 through 3.17 and have been tabulated in Appendix A.

Again, the Curve Fitting Toolbox in MATLAB was used to produce a set of curve fits for these data. From Fig. 3.8 through 3.17 it can be seen that the radiative heating for each nose radius does not increase linearly with velocity. Exponential equations were first used to fit to the data but did not yield reasonable results across the densities. To improve upon the exponential equations the `spline` function in MATLAB was used. The `spline` function generates a smooth curve that passes through each of the data points in series. It does this by constructing a third order

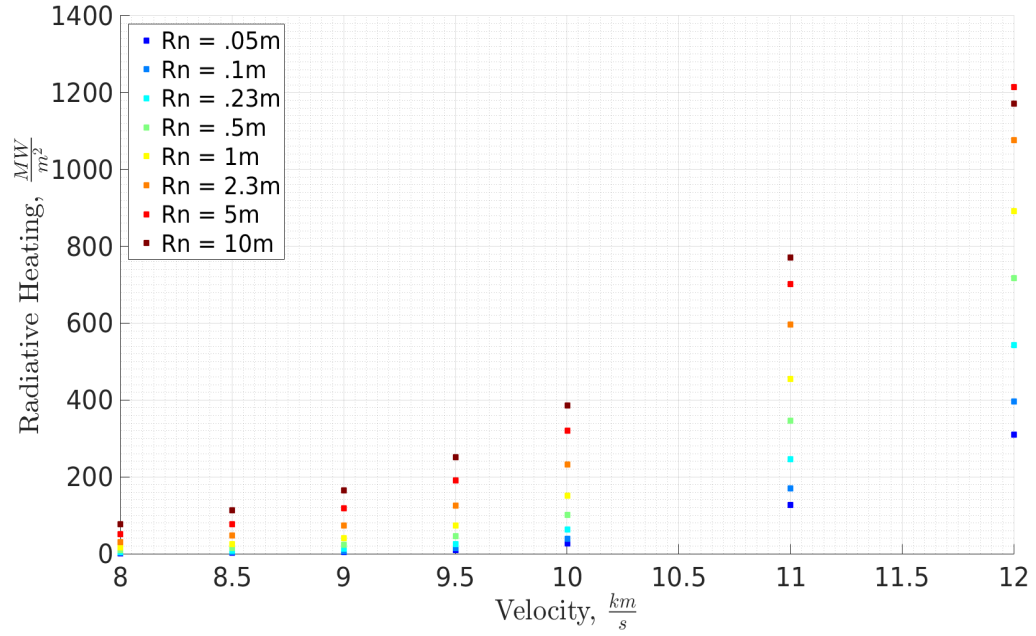


Figure 3.8: Raditative heating as a function of velocity for various nose radii at $\rho_{\infty} = 1.840E - 2 \frac{kg}{m^3}$.

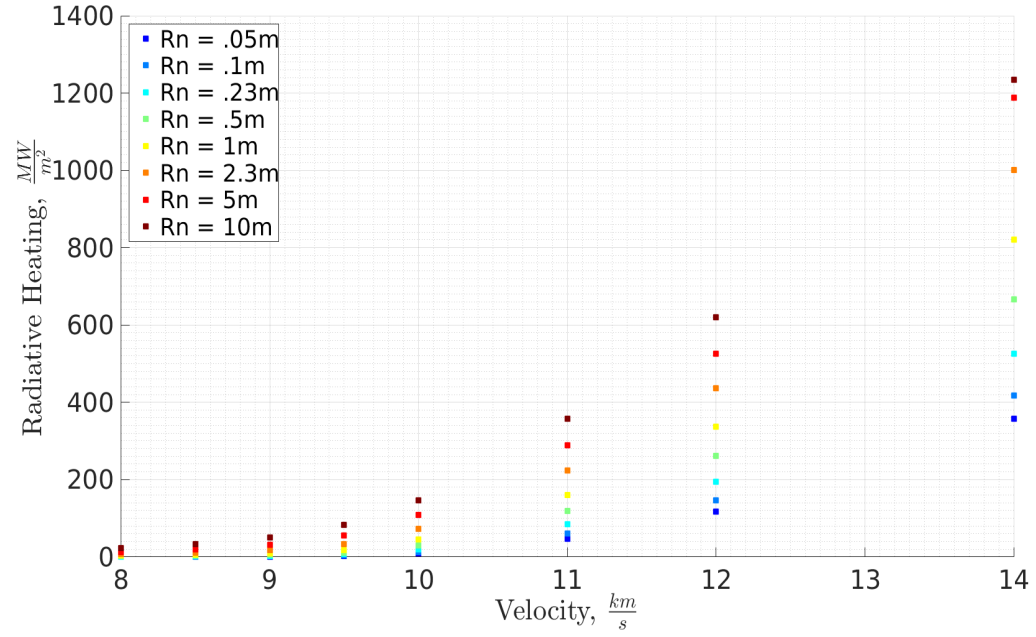


Figure 3.9: Raditative heating as a function of velocity for various nose radii at $\rho_{\infty} = 7.2579E - 3 \frac{kg}{m^3}$.

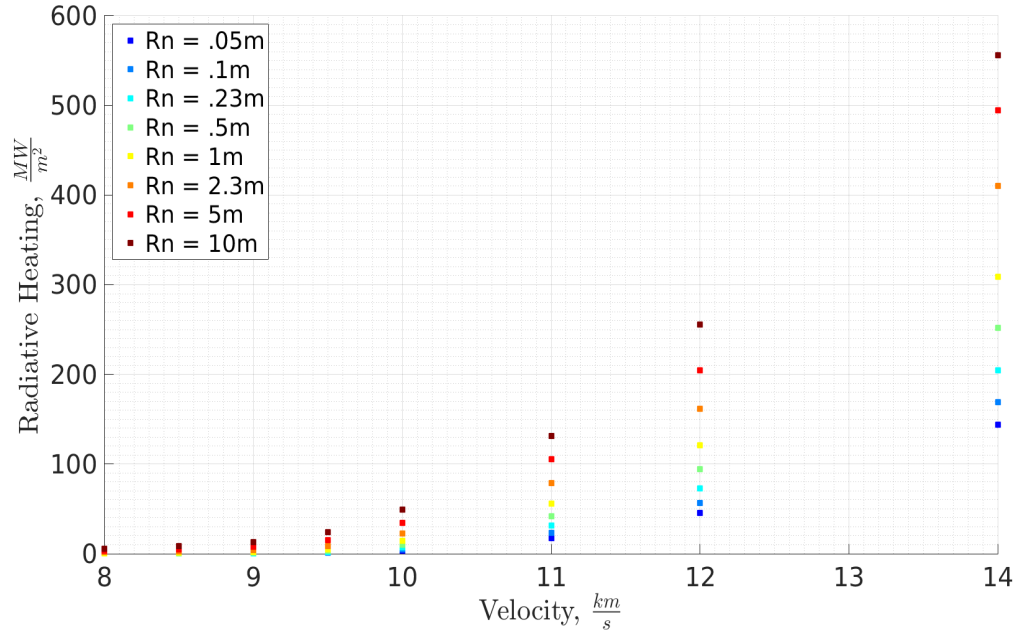


Figure 3.10: Raditive heating as a function of velocity for various nose radii at $\rho_{\infty} = 2.9948E - 3 \frac{kg}{m^3}$.

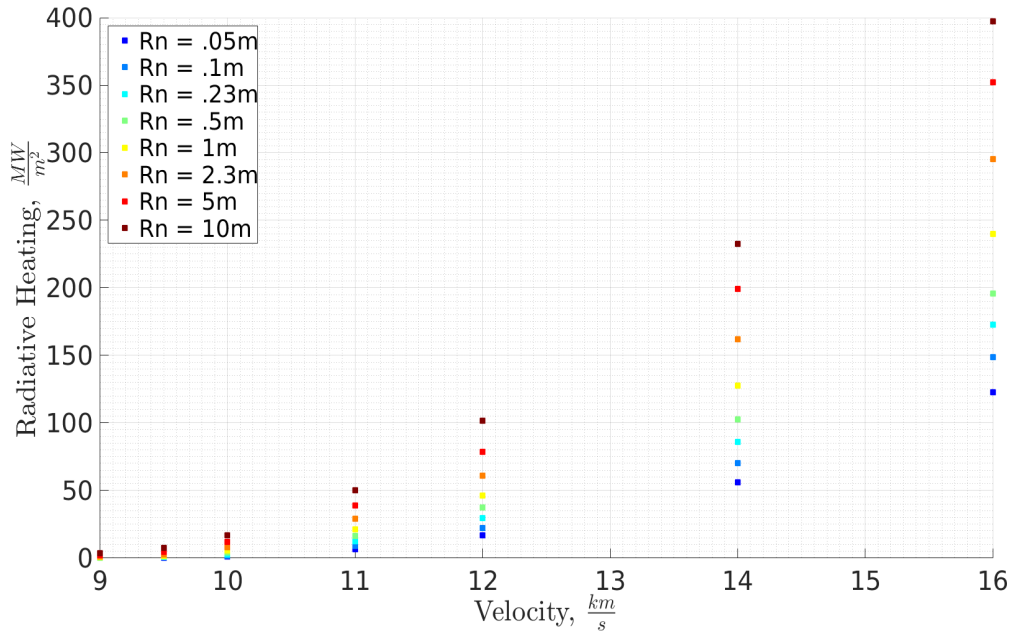


Figure 3.11: Raditive heating as a function of velocity for various nose radii at $\rho_{\infty} = 1.3167E - 3 \frac{kg}{m^3}$.

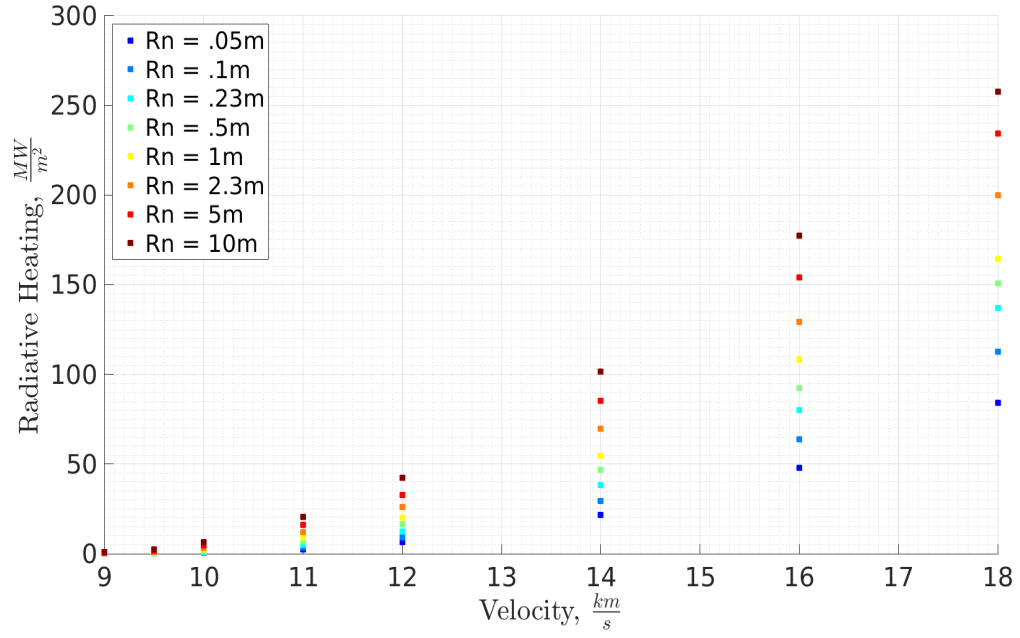


Figure 3.12: Raditive heating as a function of velocity for various nose radii at $\rho_{\infty} = 6.3137E - 4 \frac{kg}{m^3}$.

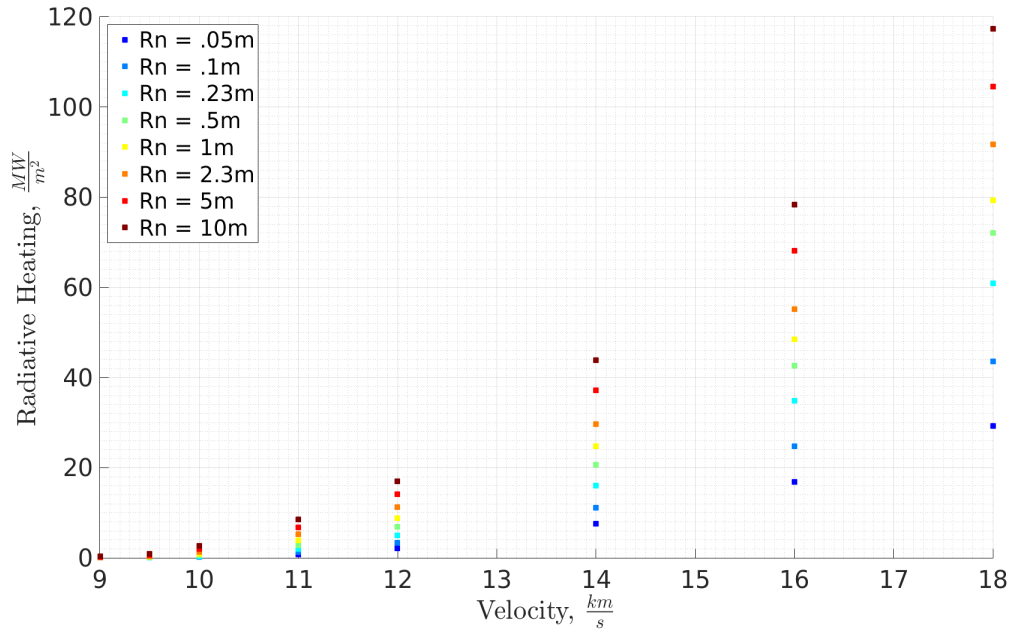


Figure 3.13: Raditive heating as a function of velocity for various nose radii at $\rho_{\infty} = 3.0592E - 4 \frac{kg}{m^3}$.

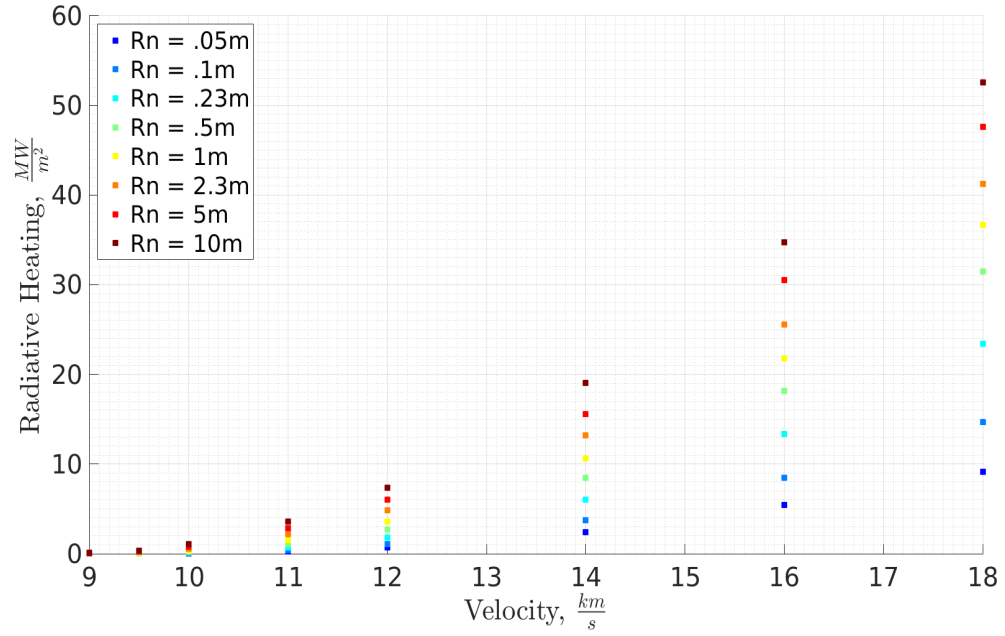


Figure 3.14: Raditive heating as a function of velocity for various nose radii at $\rho_{\infty} = 1.4713E - 4 \frac{kg}{m^3}$.

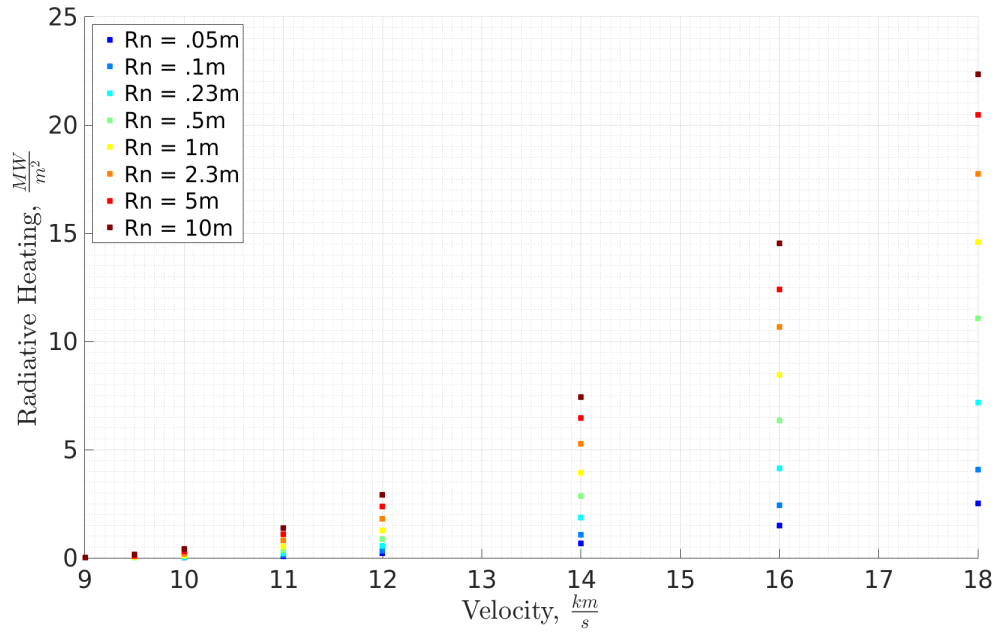


Figure 3.15: Raditive heating as a function of velocity for various nose radii at $\rho_{\infty} = 6.6593E - 5 \frac{kg}{m^3}$.

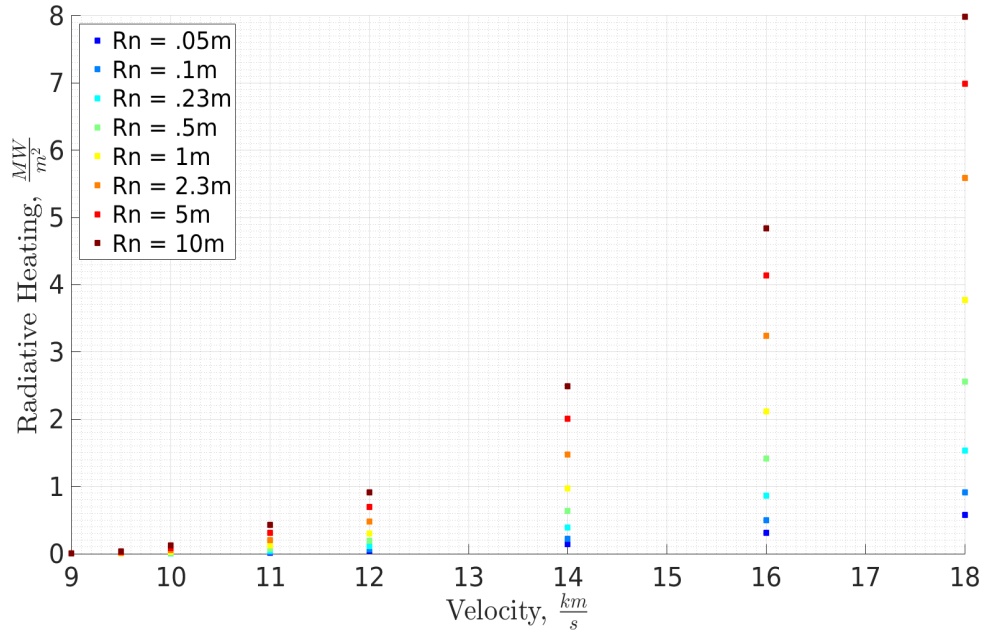


Figure 3.16: Raditative heating as a function of velocity for various nose radii at $\rho_{\infty} = 2.5239E - 5 \frac{kg}{m^3}$.

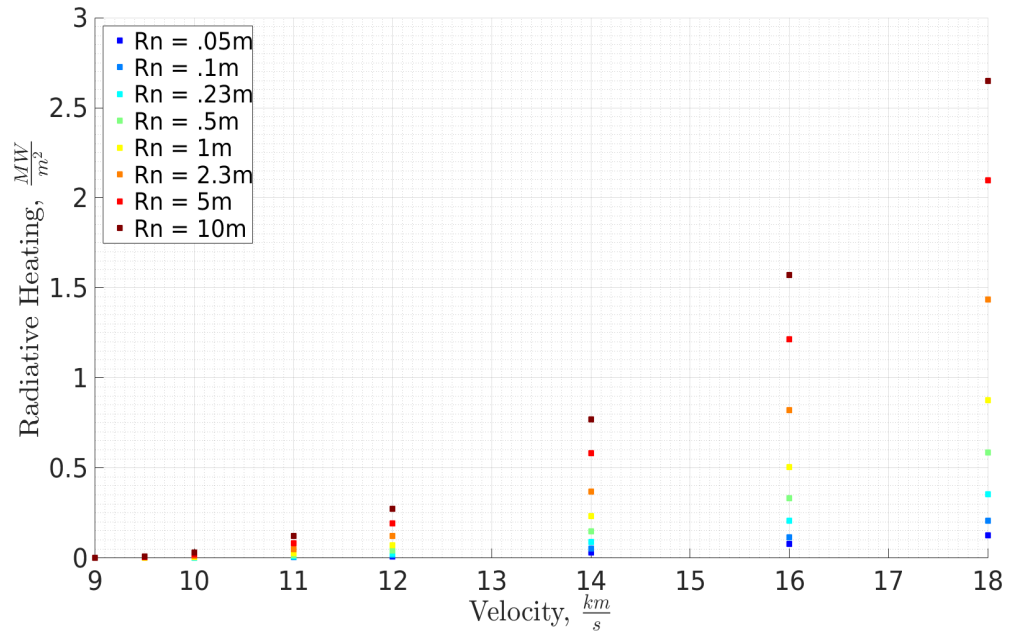


Figure 3.17: Raditative heating as a function of velocity for various nose radii at $\rho_{\infty} = 9.6940E - 6 \frac{kg}{m^3}$.

polynomial, see Eq. 3.7, between each of the data points.

$$y = a_1x^3 + a_2x^2 + a_3x + a_4 \quad (3.7)$$

The result from the curve fit was a set of 544 third order polynomials to represent the data tabulated by Sutton and Hartung (1990). Also, in an effort to have more full profiles of the radiative heating, linear equations were developed to extrapolate the radiative heating for velocities from the lowest tabulated value down to zero. This added 80 more equations for a total of 624. Using these equations it was possible to develop a FORTRAN subroutine to do multiple interpolations to approximate the radiative heating. This new subroutine, `qfit`, employed an interpolation method to that described for the density calculation. Once the `qfit` subroutine is called, the radiative heating is set to be zero in case the entry body is not within the range of applicability of the data. The subroutine then determines if the body is within range of the densities of the tables, and if it is, it uses the densities above and below the atmospheric density where the entry body is located. Using these two densities, a percent difference is calculated to be used to linearly interpolate in a later calculation. Then, the nose radius is checked in a similar way and to calculate a percent difference for the nose radius. Next, the subroutine will check the velocity to make sure that it is within the range of the table. The subroutine will then use the third order polynomials to approximate the radiative heating at body radii one size smaller and one size larger than itself at the lower and higher densities. This meant that 4 polynomials need to be checked at each density comparison which leads to more than 1,000 equations needed for the subroutine. Since each of these equations consisted of polynomials with coefficients consisting of double precision, floating point numbers, it was important to develop an algorithm to generate the code to eliminate human error. A MATLAB script was written to generate a FORTRAN subroutine that contained more than 7500 lines of code. After the subroutine was written, it was compiled and spot checked to make sure that the results matched up to the tabulated values

by Sutton and Hartung (1990), and once confirmed, it was added into the meteor program.

To add some clarification to the algorithm just described an example will be walked through. Assume at some time step, the entry body is at an altitude of 33 km ($\rho_\infty = 0.0112620 \frac{kg}{m^3}$), nose radius of .9 meters, and velocity of $9.2 \frac{km}{s}$. The subroutine will calculate the radiative heating for bodies with nose radii of .5 meters and 1 meter at freestream densities of $1.8410E-2 \frac{kg}{m^3}$ and $7.259E-3 \frac{kg}{m^3}$ at a velocity of $9.2 \frac{km}{s}$. For both densities, linear interpolation is used to approximate the correct heating for a .9 meter object. Then the heating is linearly interpolated between the densities to approximate the correct radiative heating for a .9 meter body at the specified freestream density.

Tauber and Sutton (1991) used the Sutton-Hartung data as well to develop radiative heating relations that were valid for velocities between 10 to 16 km/s, altitudes between 54 and 72 km, and body radii between .3 to 3 meters. They were able to achieve a maximum error of 23% and an average error of 8%. Because of the way the `spline` function generates the third order polynomials, an error comparison cannot be completed because these polynomials will reproduce the radiative heating for velocities, radii, and freestream densities, exactly. The range of applicability can be seen in Table 3.2.

Table 3.2: Range of applicability of the `qfit` subroutine.

Radiative Calculation Range			
Altitude Range	Lower Velocity	Higher Velocity	Body Radii
30 - 36 km	8 km/s	12 km/s	.05 - 10 m
36 - 42 km	8 km/s	14 km/s	.05 - 10 m
42 - 48 km	9 km/s	14 km/s	.05 - 10 m
48 - 54 km	9 km/s	16 km/s	.05 - 10 m
54 - 60 km	9 km/s	18 km/s	.05 - 10 m
60 - 66 km	9 km/s	18 km/s	.05 - 10 m
66 - 72 km	9 km/s	18 km/s	.05 - 10 m
72 - 78 km	9 km/s	18 km/s	.05 - 10 m
78 - 84 km	9 km/s	18 km/s	.05 - 10 m

3.6 Convective Heating

Due to the current study being applied to smaller, slower entry bodies, it was necessary to include a convective heating calculation, as it should make up a significant amount of the heat flux. This heating calculation was not as involved as the methods developed for curve fitting the density ratios and radiative heating. To determine the stagnation point convective heating, Eq. 3.8 was used

$$q_c = C_1 \left(\frac{\rho_1}{r_n} \right)^{.5} V^{C_2} \left(1 - \frac{h_w}{h_T} \right) \quad (3.8)$$

where C_1 and C_2 are constants that depend on the atmospheric composition and h_w and h_T are the enthalpy of the wall and enthalpy of the freestream, respectively (Marvin and Deiwert, 1965). To calculate the total freestream enthalpy Eq. 3.9 was used.

$$h_T = \frac{V^2}{2} + h_1 \quad (3.9)$$

Then using Eq. 3.10, Eq. 3.11, and Eq. 3.12 it was possible to find the enthalpy at wall (Lyne, 1992).

$$h_w = 940T_w + 0.1043 (T_w)^2 \quad (3.10)$$

$$T_w = \left(\frac{q_s}{\epsilon\sigma} \right)^{0.25} \quad (3.11)$$

$$q_c = C_1 \left(\frac{\rho_1}{r_n} \right)^{.5} V^{C_2} \quad (3.12)$$

After finding the convective heating, the total heat was calculated by adding the convective and radiative heating together. At each time step the heat flux was calculated to determine if any mass was going to be ablated. This was done by stepping around the body in 1 degree increments starting at the nose. At each step the heating was calculated, as well as the surface area of the body swept out at each increment. Multiplying the heating by the surface area allowed for a heat flux to be summed until the body angle reached 90 degrees. If the heat flux to the surface was

larger than the heat of ablation then the model determines how much mass is ablated based on the scale factors developed by Green and Nicolet (1978). After determining the amount of ablated mass, a new nose radius was found by solving the spherical volume equation for radius with the new mass and the original material density. By looking at Fig. 3.18, it is obvious that the convective heating plays an important role in the trajectory. Without accounting for convection, the ablation of the body would be incorrectly approximated and would yield unreliable results.

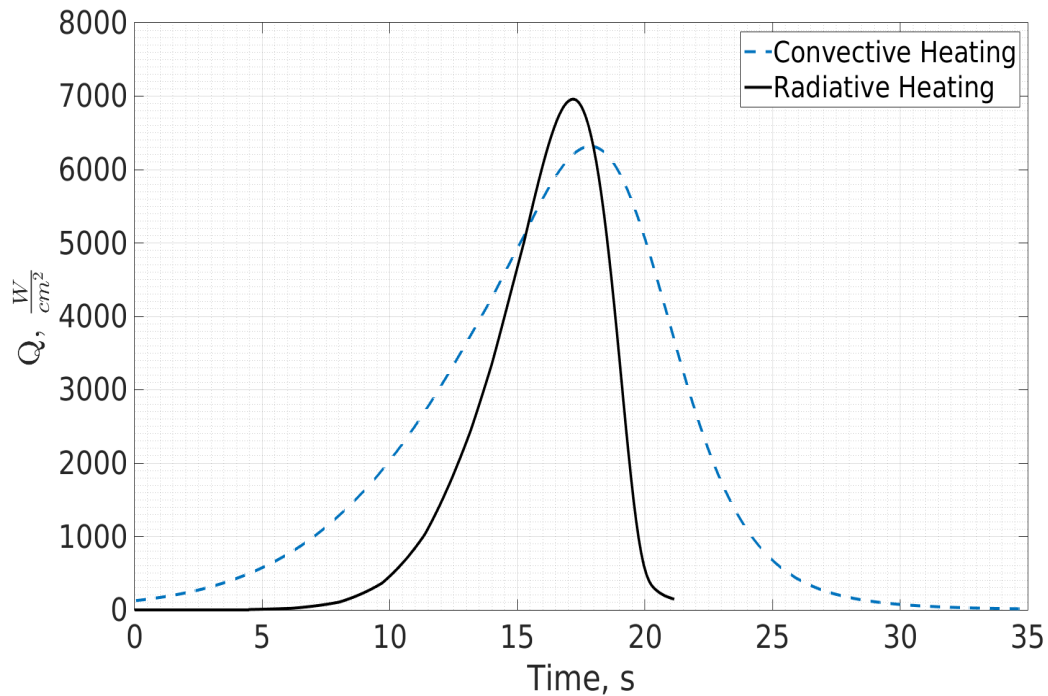


Figure 3.18: A comparison of radiative and convective heating for a 0.1 meter body entering at -15 degrees with a velocity of 14 km/s.

Chapter 4

Results

4.1 Discussion

The adaptations that were made to the model resulted in varying degrees of success. Some of the entry parameters chosen produced trajectories where the radiative heating is calculated through the full range of applicability for the curve fitting algorithm developed for the data from Sutton and Hartung (1990). Others produced trajectories that have fragmented data sets for radiative heating because of the constraints setup for the `qfit` subroutine and some never enter a valid range to approximate the radiative heating. The code for the model is listed in Appendix B.

The results of the trajectory plots are grouped by nose radius. For each nose radius, plots of convective and radiative heating as functions of time are plotted. These are used to determine whether a given trajectory has been accurately calculated. Because of the range of applicability for the `qfit` subroutine has staggered velocity limits, the radiative calculations can have gaps or be completely missing. Plots showing detailed information about the trajectory such as temperature, velocity, altitude, and mass will be omitted for cases where the model is unable to capture an accurate profile of the radiative heating.

For trajectories that allow for accurate radiative heating approximation, there are four other relationships that will be considered. The first is altitude as a function of velocity. This allows for a quick check to make sure the flight path makes sense. If the model had large errors in the calculation of the equations of motion they would show up in an altitude-velocity profile. Next, to get an idea of the thermal soaking involved in the trajectories it was necessary to also look at altitude as a function of stagnation temperature and stagnation temperature as a function of time. For both of these temperature plots there are black lines at 1,000 K, 2,000 K, and 3,000 K to represent different limits. According to Opik (1958), the vaporization temperature of meteoric stone is near 2,100 K. These plots help to visualize how long the meteors will be subject to high temperatures and when they may begin to experience high speed cooling. Another important relation to look at is mass as a function of velocity. High velocities mean higher stagnation temperature and intense heating. As the body is heated, it will ablate, shrinking the ballistic coefficient, and cause the body to slow down faster. It is also worth mentioning that plots of radius as a function of velocity have been omitted because the curves are similar to those of the mass as a function of velocity.

4.2 5 cm Nose Radius

The first trajectory analyzed consists of a body with a nose radius of 5 cm. As discussed previously, there are two sources of aerodynamic heating, convection and radiation. By looking at Fig. 4.1 through 4.16 it can be seen that the heating for the 5 cm body is only experiencing convective heating. This is not because radiative heating doesn't apply, but because as soon as the body ablates due to the convective heating, it will have a nose radius of less than 5 cm which pushes it outside the range of applicable nose radii for the curve fitted data from Sutton and Hartung (1990). Because of this oversight, further trajectory plots will not be shown because their profiles will be incorrect.

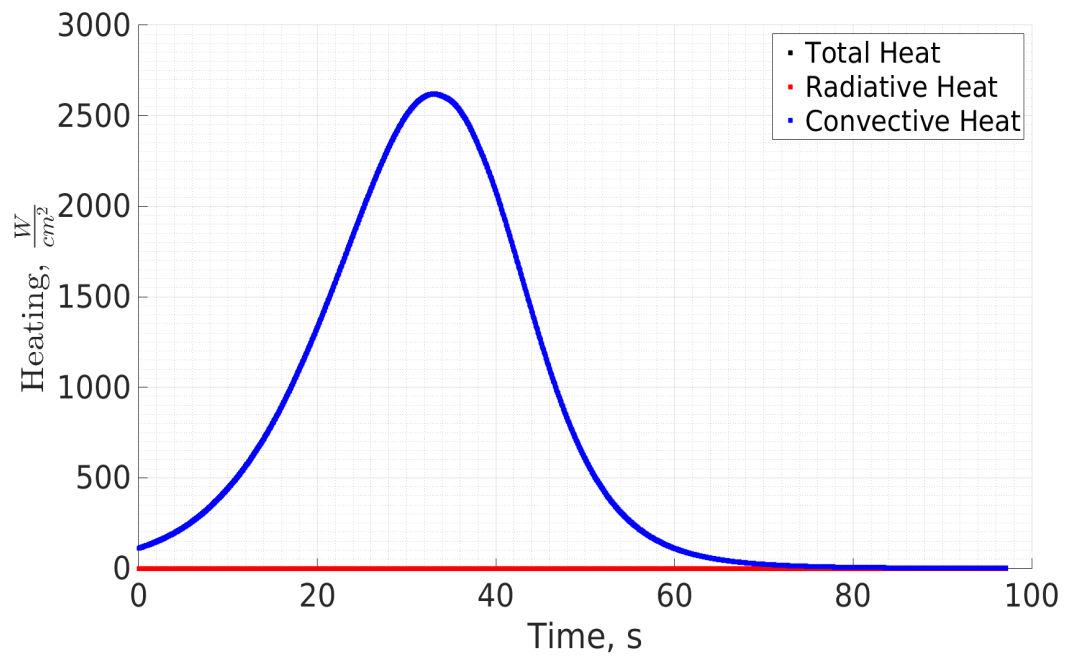


Figure 4.1: Heating-time profile for 5 cm body entering at -8 degrees and 12 km/s.

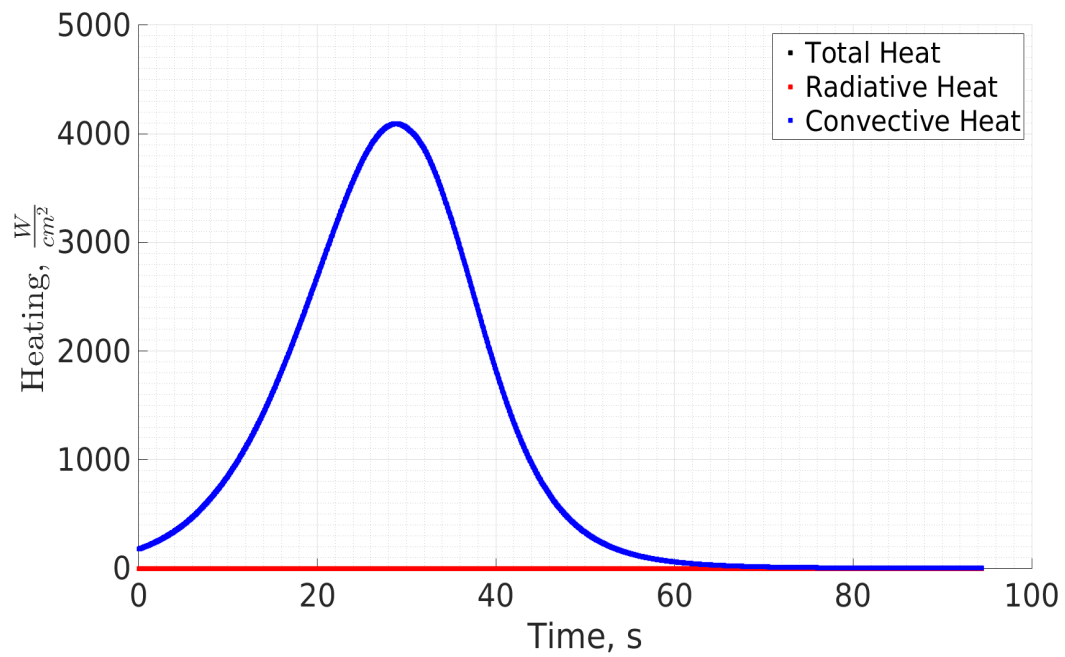


Figure 4.2: Heating-time profile for 5 cm body entering at -8 degrees and 14 km/s.

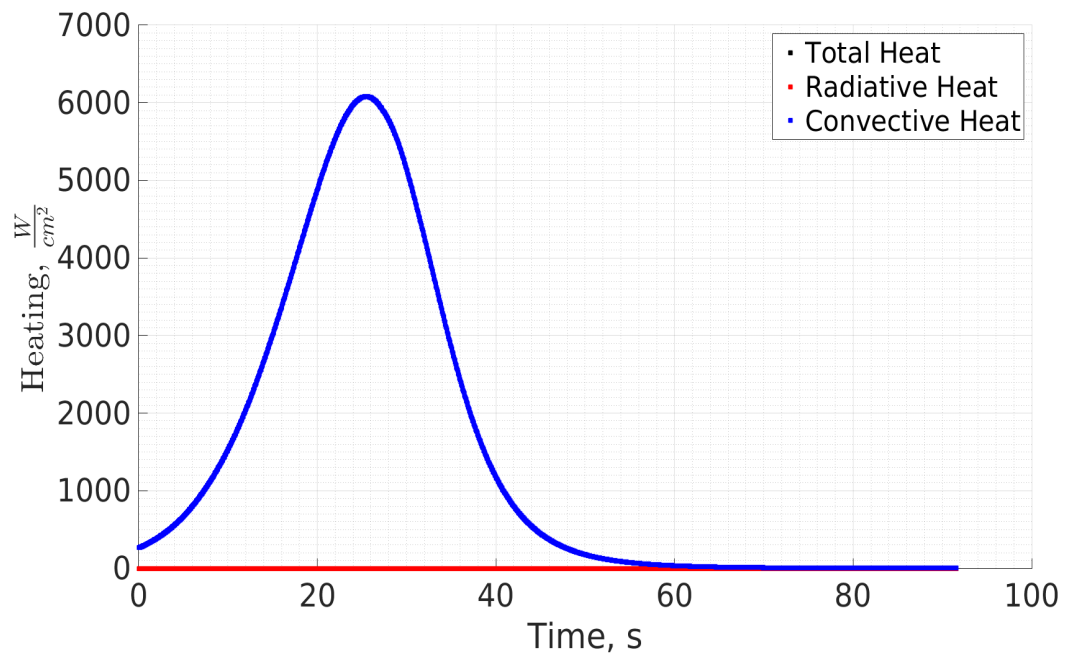


Figure 4.3: Heating-time profile for 5 cm body entering at -8 degrees and 16 km/s.

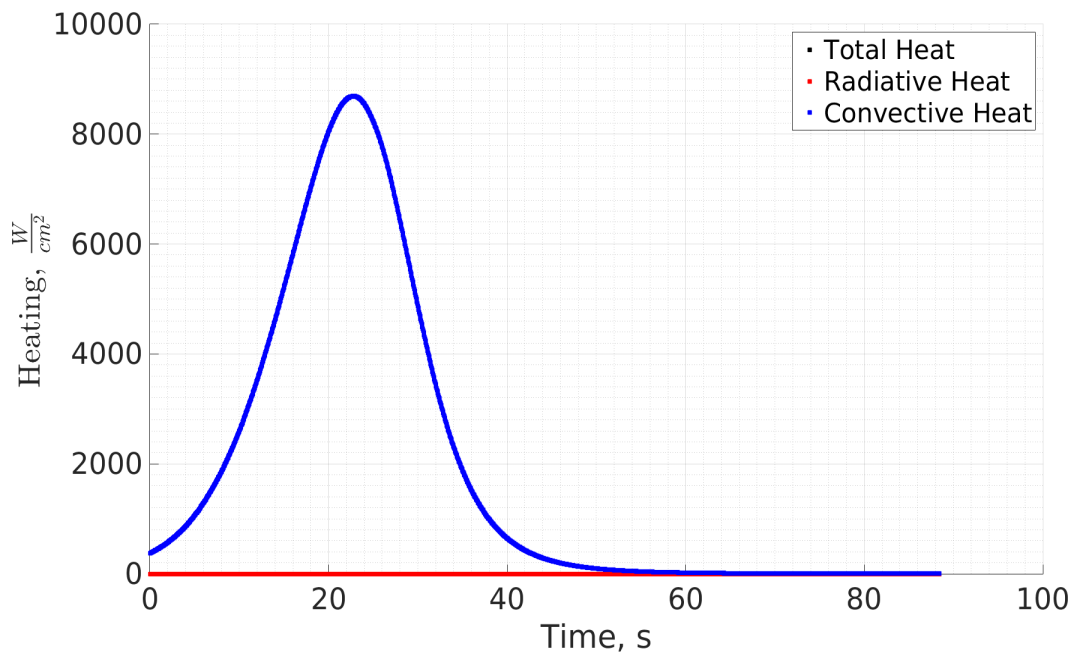


Figure 4.4: Heating-time profile for 5 cm body entering at -8 degrees and 18 km/s.

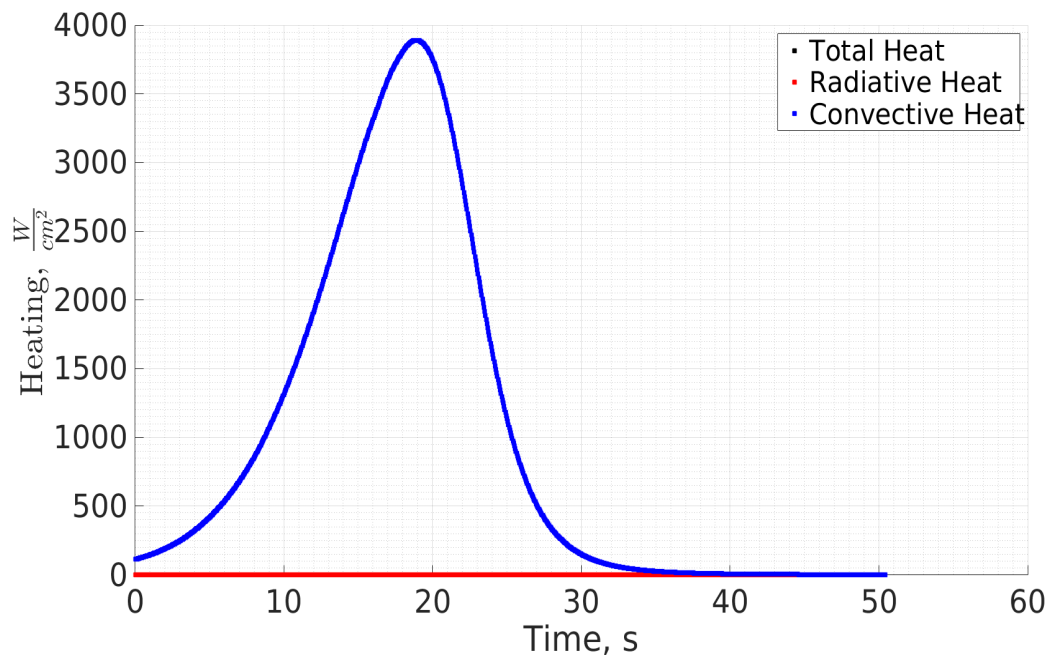


Figure 4.5: Heating-time profile for 5 cm body entering at -15 degrees and 12 km/s.

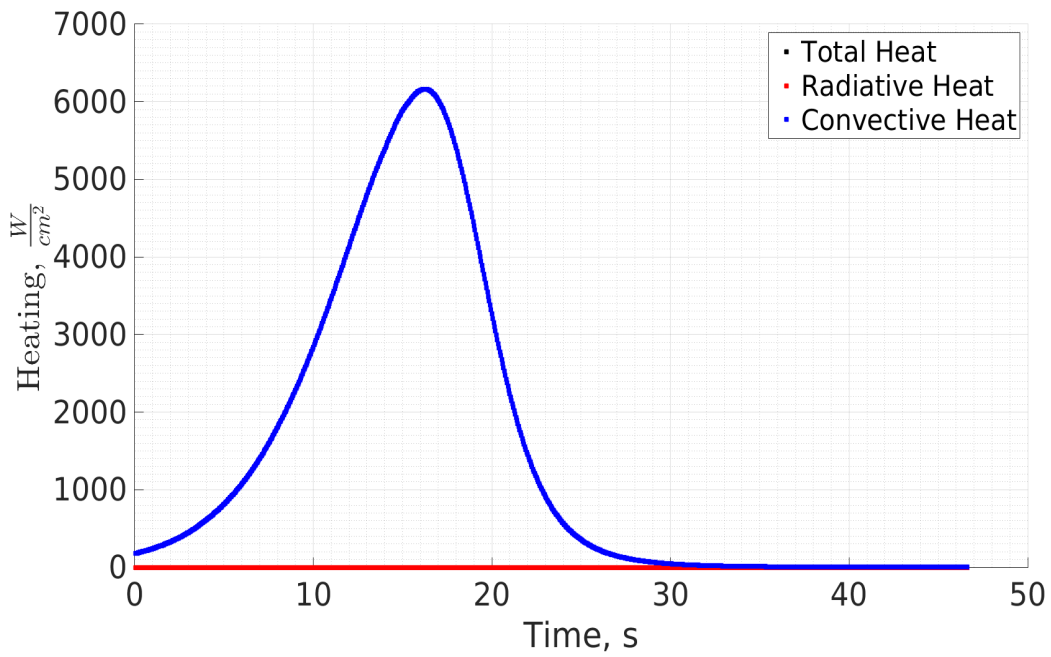


Figure 4.6: Heating-time profile for 5 cm body entering at -15 degrees and 14 km/s.

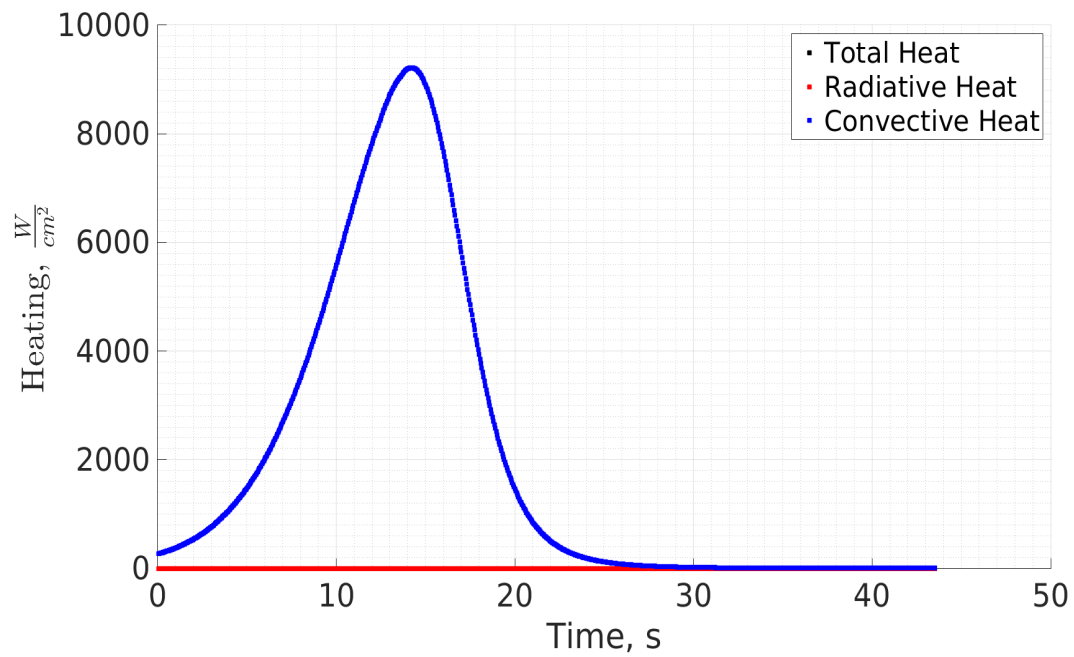


Figure 4.7: Heating-time profile for 5 cm body entering at -15 degrees and 16 km/s.

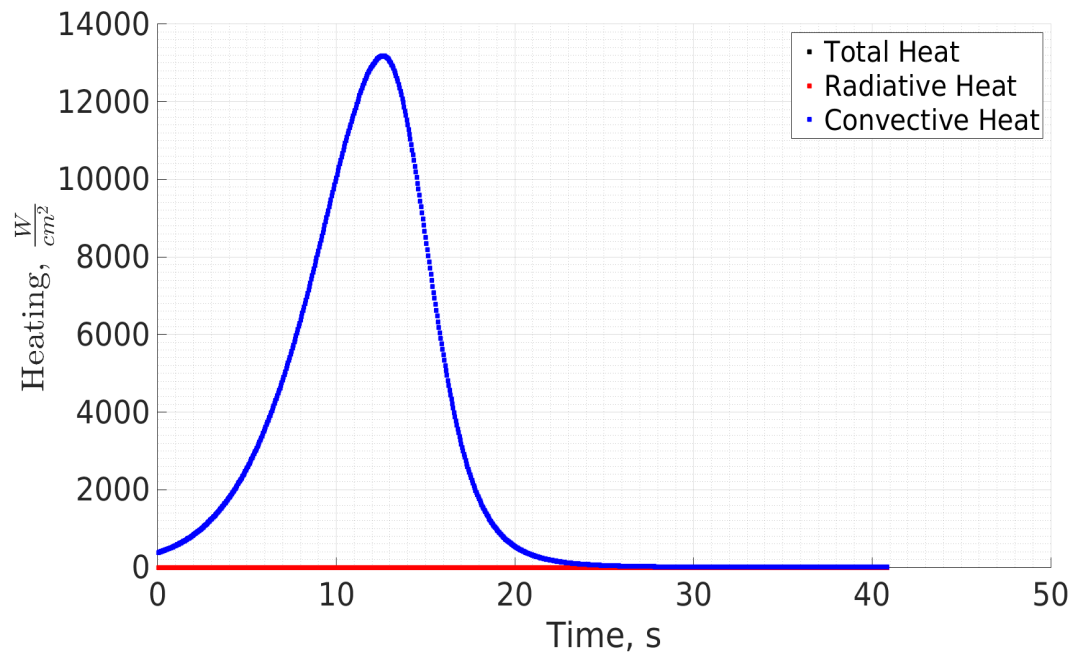


Figure 4.8: Heating-time profile for 5 cm body entering at -15 degrees and 18 km/s.

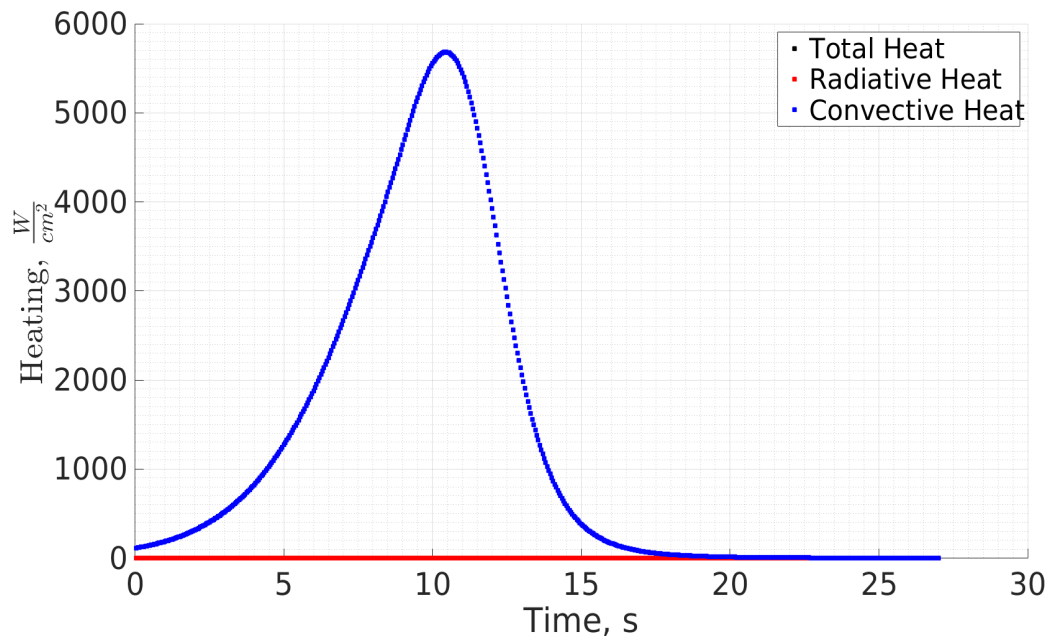


Figure 4.9: Heating-time profile for 5 cm body entering at -30 degrees and 12 km/s.

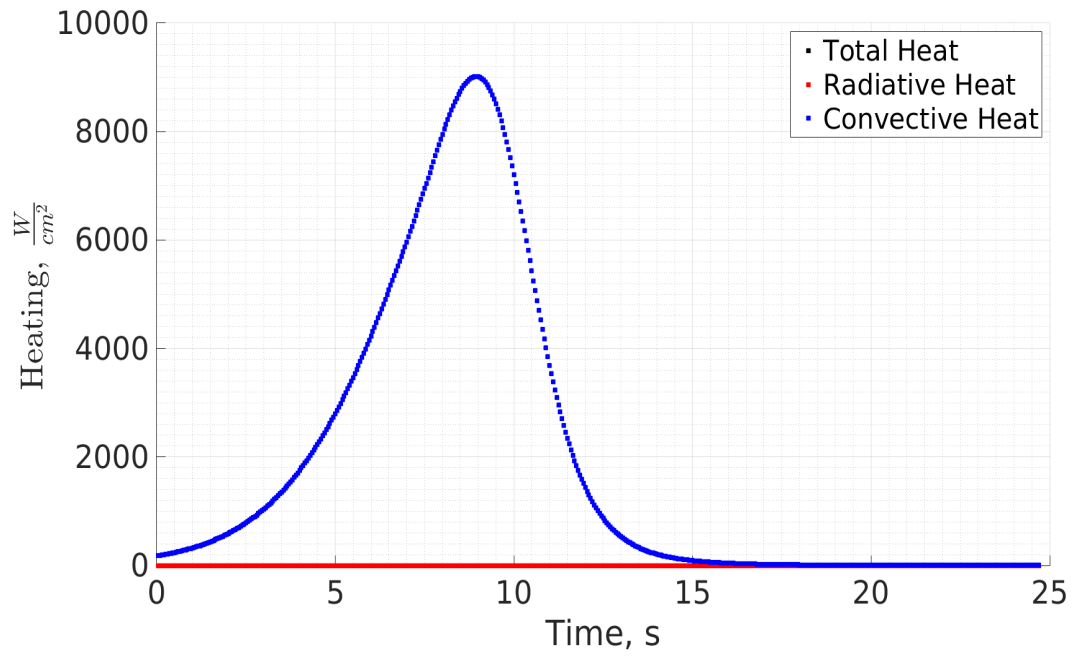


Figure 4.10: Heating-time profile for 5 cm body entering at -30 degrees and 14 km/s.

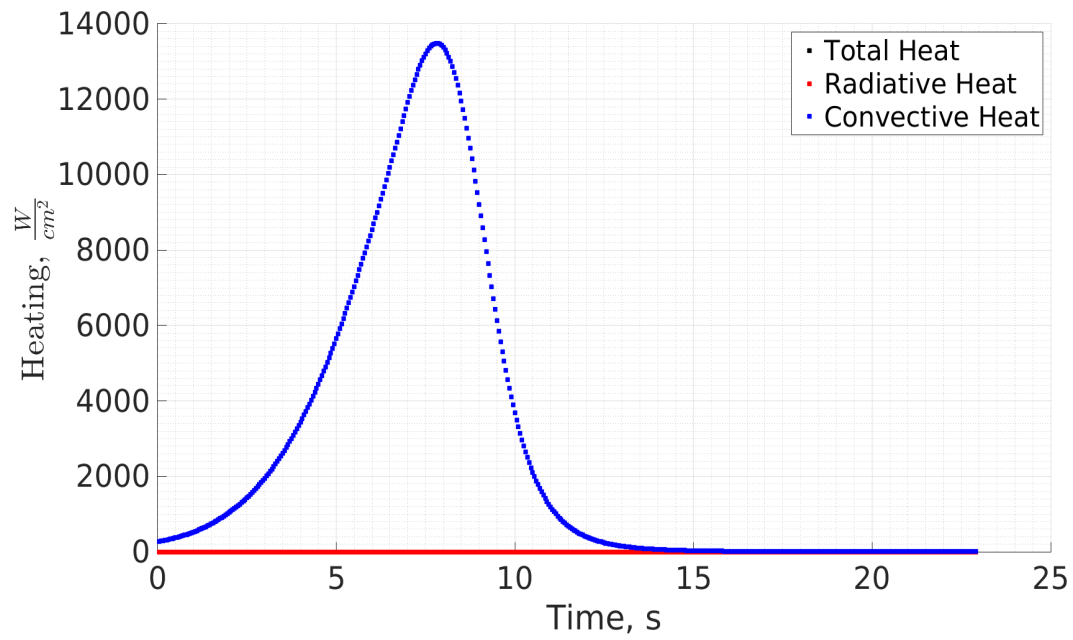


Figure 4.11: Heating-time profile for 5 cm body entering at -30 degrees and 16 km/s.

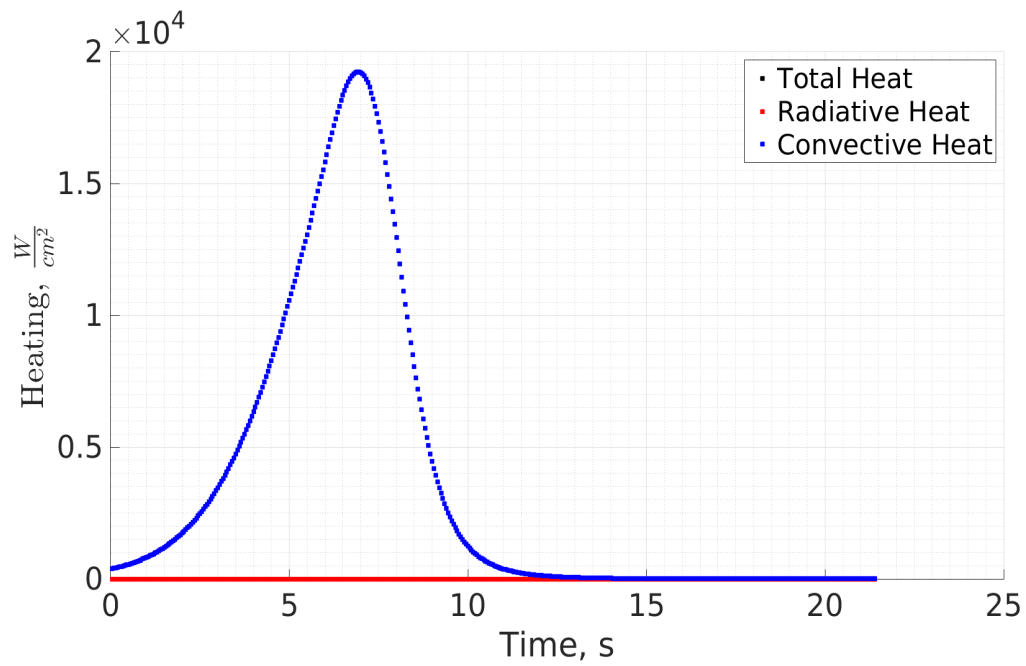


Figure 4.12: Heating-time profile for 5 cm body entering at -30 degrees and 18 km/s.

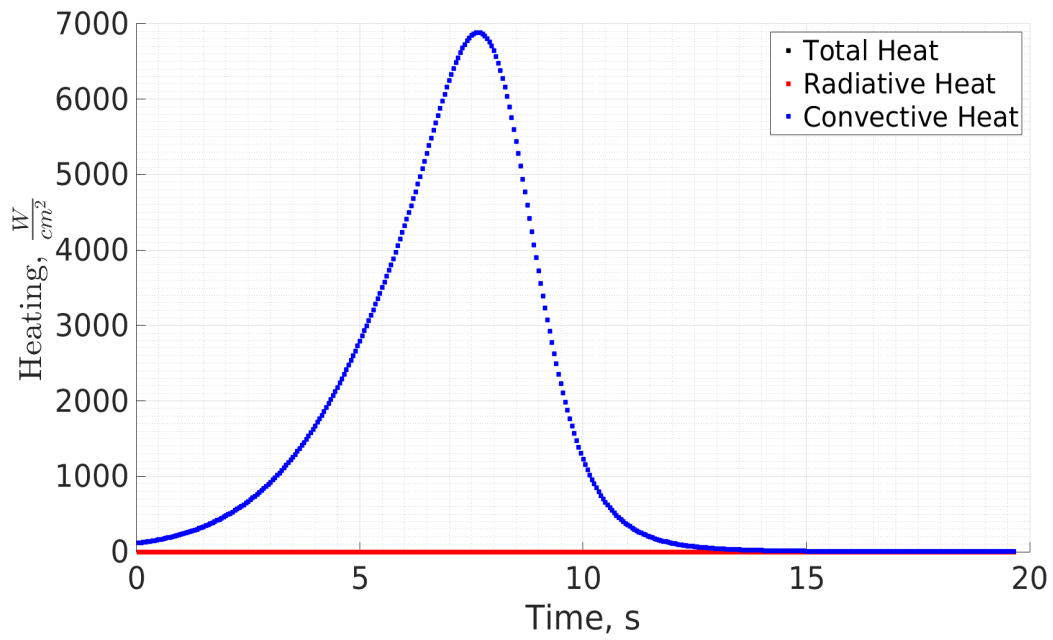


Figure 4.13: Heating-time profile for 5 cm body entering at -45 degrees and 12 km/s.

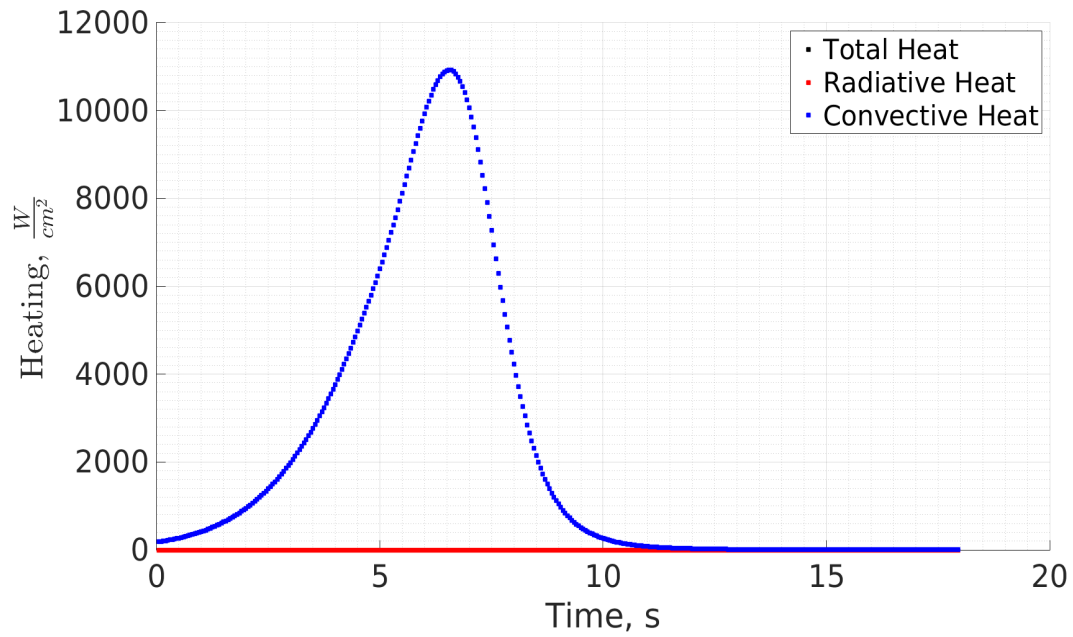


Figure 4.14: Heating-time profile for 5 cm body entering at -45 degrees and 14 km/s.

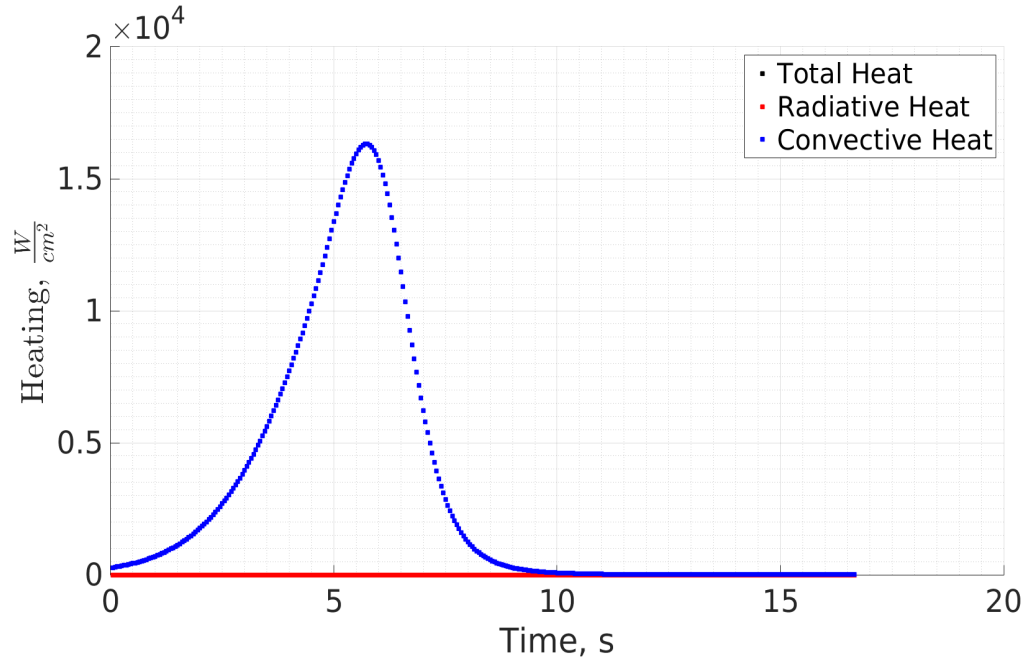


Figure 4.15: Heating-time profile for 5 cm body entering at -45 degrees and 16 km/s.

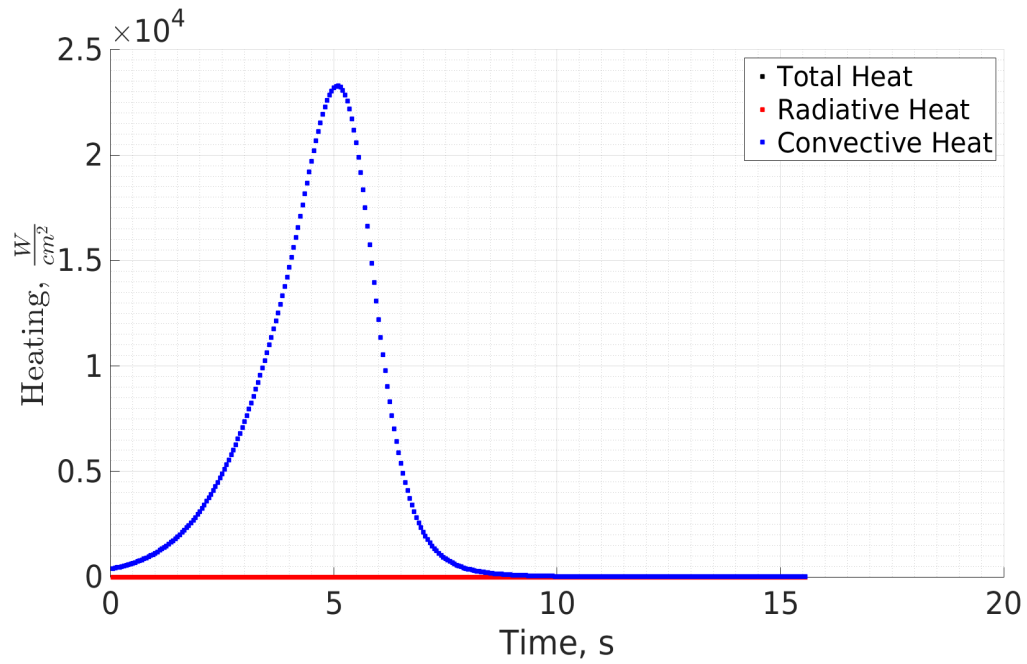


Figure 4.16: Heating-time profile for 5 cm body entering at -45 degrees and 18 km/s.

4.3 10 cm Nose Radius

The next set of trajectories have a nose radius of 10 cm. The heating predictions over the range of entry angles and velocities are shown in Figs. 4.17 through 4.32. Figures 4.17, 4.18, 4.19, 4.20, 4.21, 4.22, 4.23, 4.25, 4.26, and 4.29 have full or nearly full profiles for the radiative heating calculations while the others have too much missing data to provide reliable results. Using the radiative heating calculations as the determining factor in model validity, it can be seen that for an object with a 10 cm nose radius there are differing ranges for which the model can be applied. With an -8 degree entry angle, the max velocity is somewhere between 16 and 18 $\frac{km}{s}$, for -15 degree entry angle, the max velocity is about 16 $\frac{km}{s}$, for -30 degree entry angle, the max velocity is about 14 $\frac{km}{s}$, and for -45 degree entry angle, the max velocity is between 12 and 14 $\frac{km}{s}$. By looking at Fig. 4.17, 4.18, 4.19 and 4.20 it can be seen that convective and radiative heating play important roles when looking a range of trajectories. For the slowing moving bodies the convection is the dominant contributor to heating while radiation increase drastically as entry velocity increases. From these figures it is also noticeable that convective heating is more spread out over time and radiation rises to a peak and declines in a shorter time span. The rest of the heating plots, Fig. 4.21 through 4.32, show similar trends. As the entry angles become steeper, radiation becomes more important in the heat load calculation.

The altitude-velocity profiles, plotted in Fig. 4.33 through 4.36 show how the 10 cm object begins to slow down as it penetrates the atmosphere. The body maintains a large portion of its original kinetic energy until it reaches lower altitudes where the air density becomes thicker, causing it to slow down. As the entry angle becomes steeper, the body is able to keep more of its original kinetic energy at lower altitudes. All of the altitude-velocity profiles for the 10 cm bodies that are presented here begin to slow down rapidly and approach their terminal velocity. In Fig. 4.34 the plots show a near vertical drop below 20 km. This is where these trajectories approach their terminal velocity and enter free fall. Since the terminal velocity is a function of

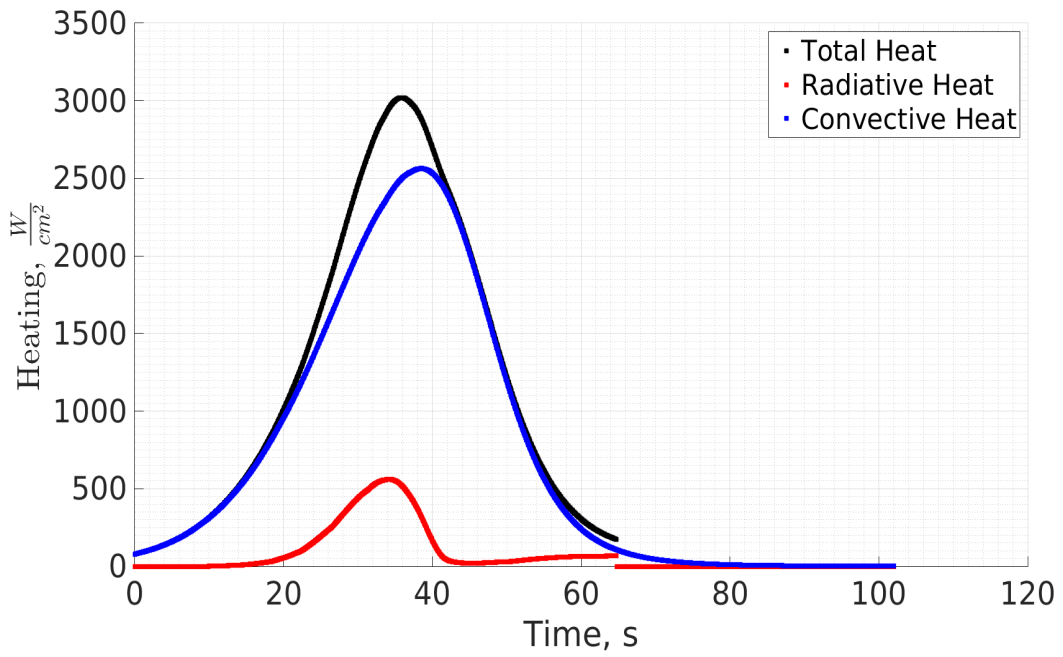


Figure 4.17: Heating-time profile for 10 cm body entering at -8 degrees and 12 km/s.

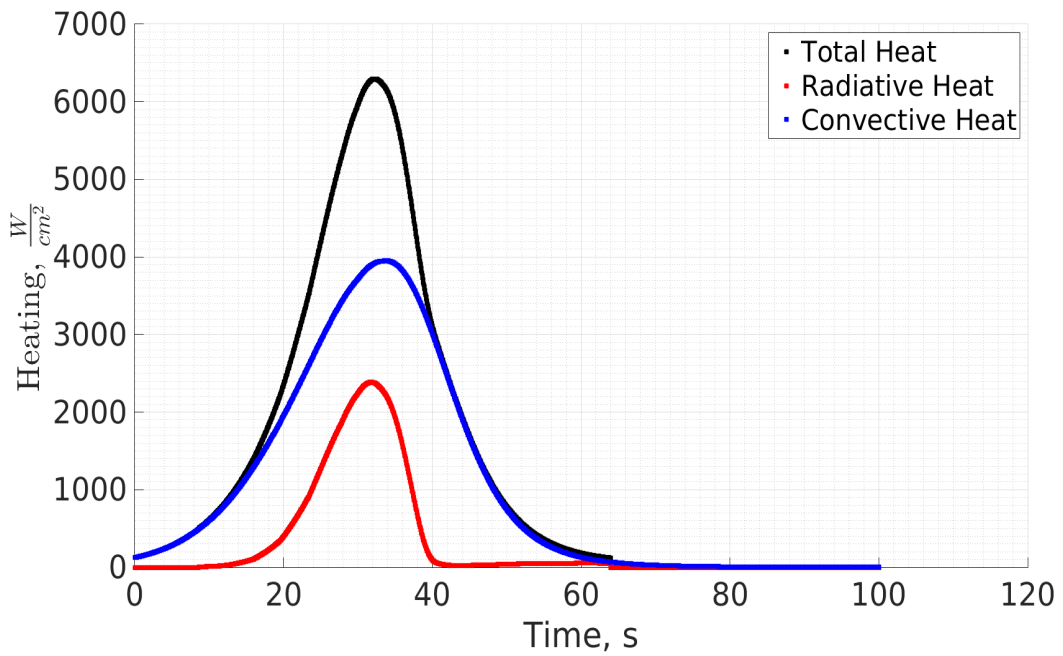


Figure 4.18: Heating-time profile for 10 cm body entering at -8 degrees and 14 km/s.

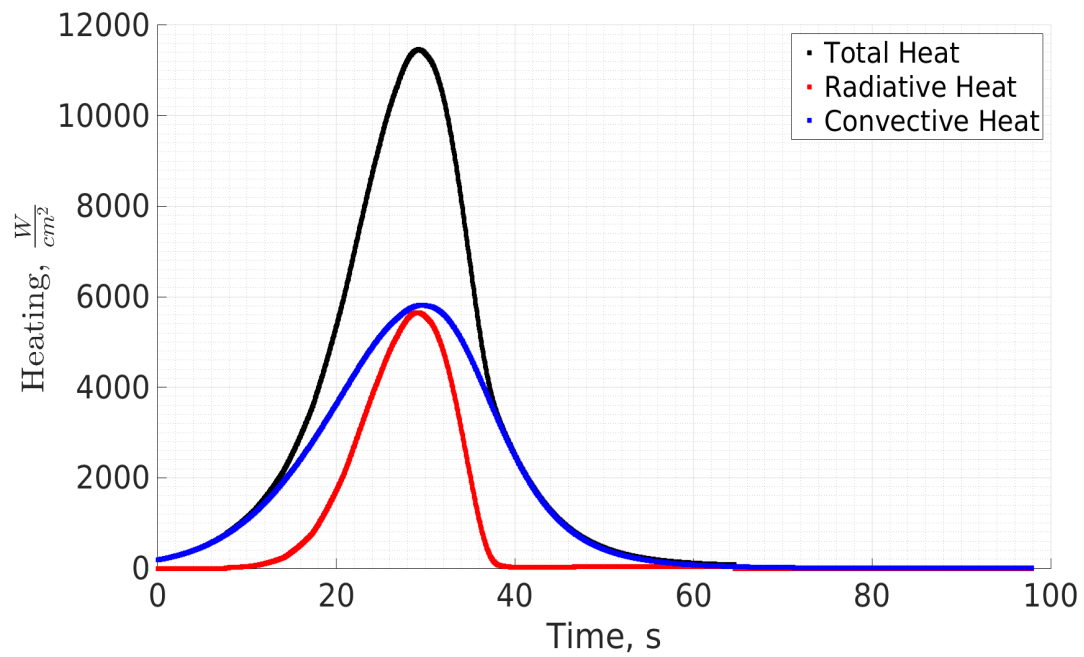


Figure 4.19: Heating-time profile for 10 cm body entering at -8 degrees and 16 km/s.

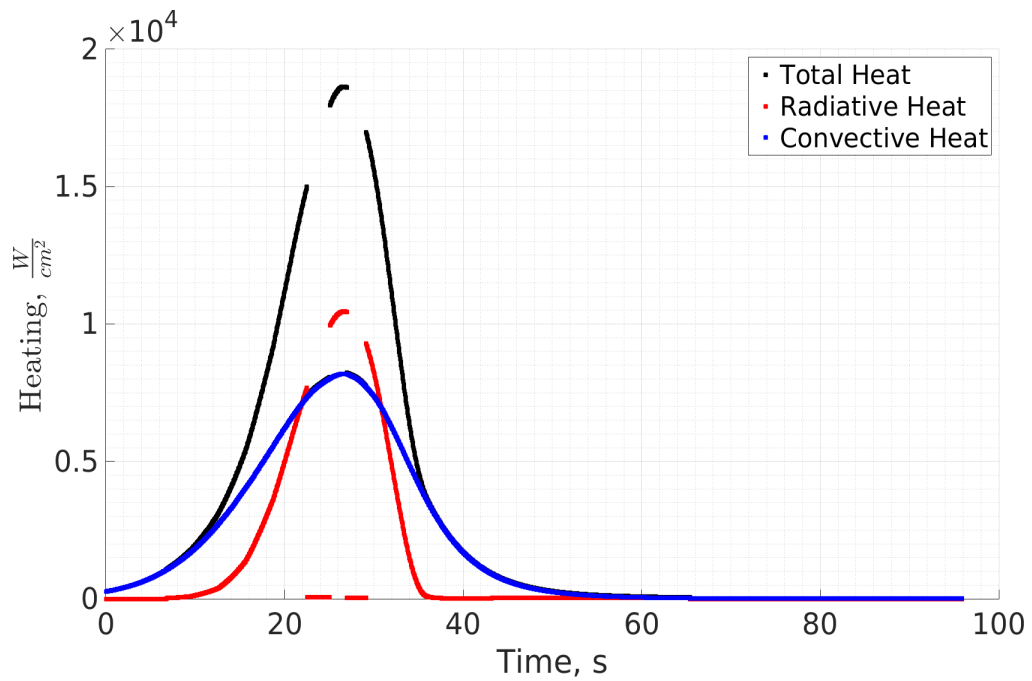


Figure 4.20: Heating-time profile for 10 cm body entering at -8 degrees and 18 km/s.

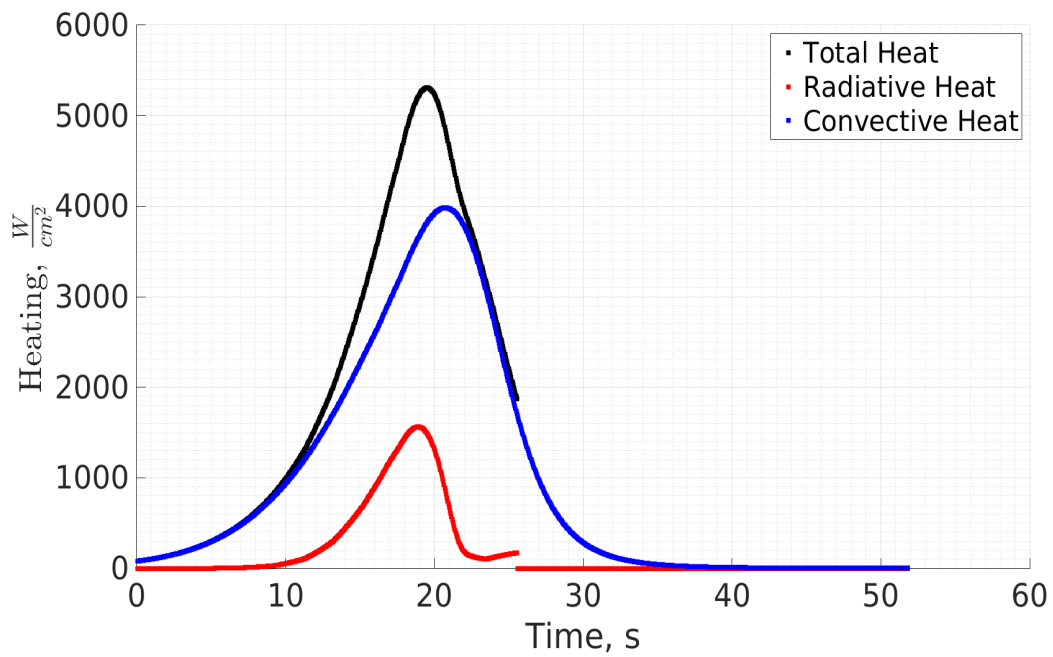


Figure 4.21: Heating-time profile for 10 cm body entering at -15 degrees and 12 km/s.

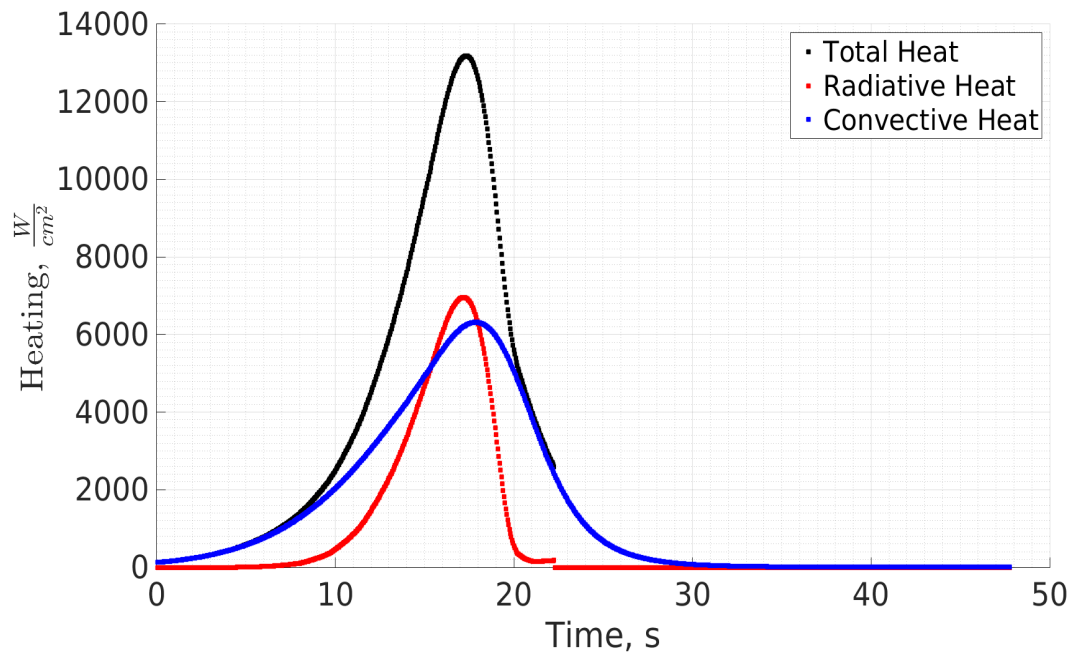


Figure 4.22: Heating-time profile for 10 cm body entering at -15 degrees and 14 km/s.

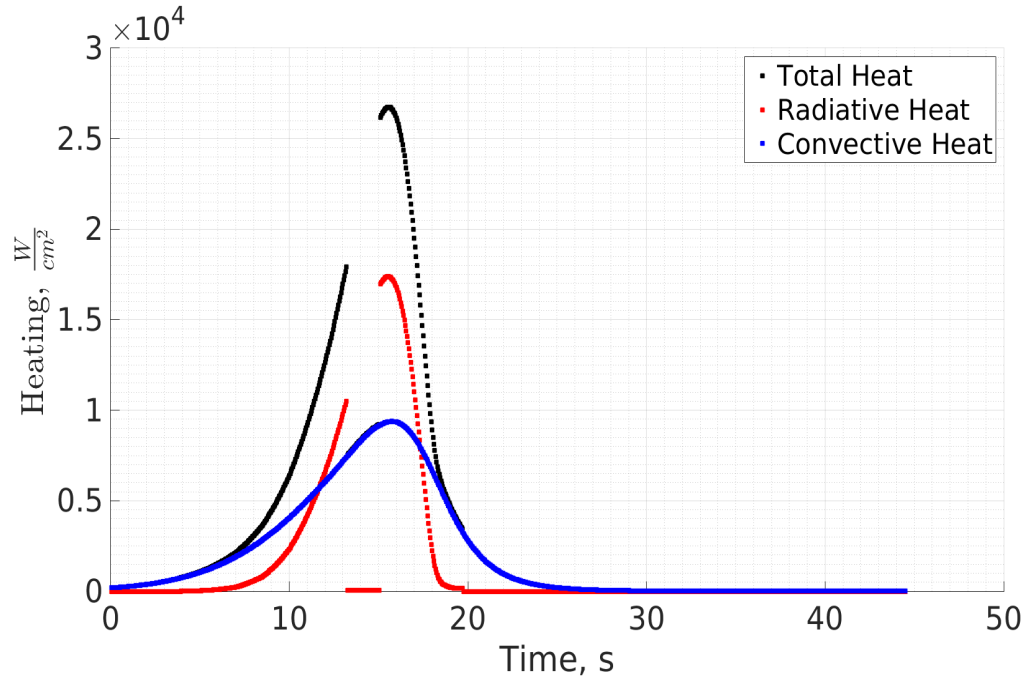


Figure 4.23: Heating-time profile for 10 cm body entering at -15 degrees and 16 km/s.

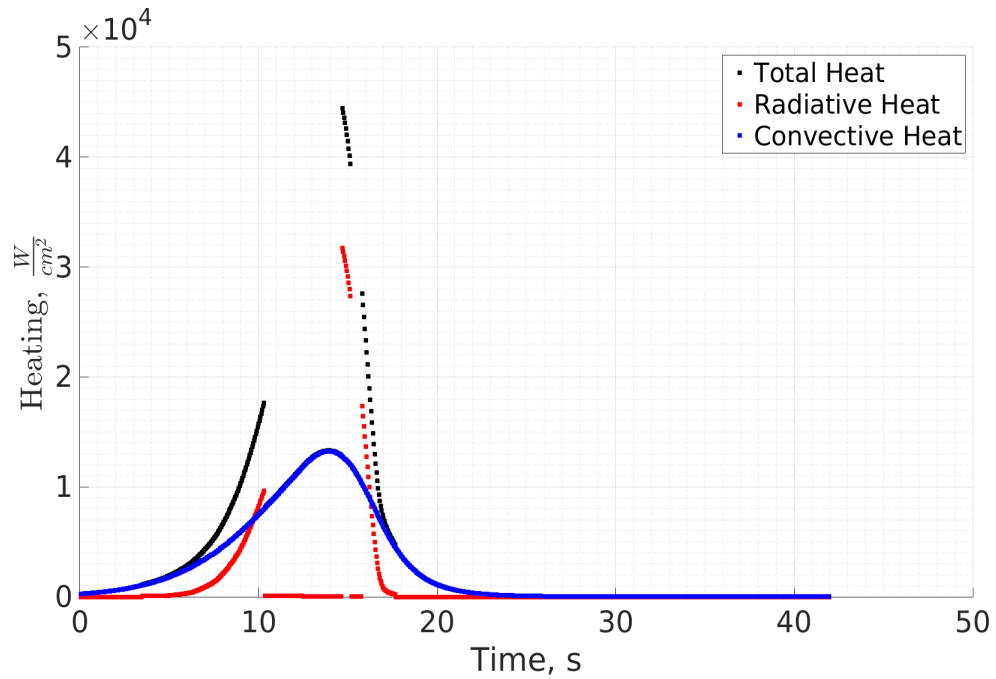


Figure 4.24: Heating-time profile for 10 cm body entering at -15 degrees and 18 km/s.

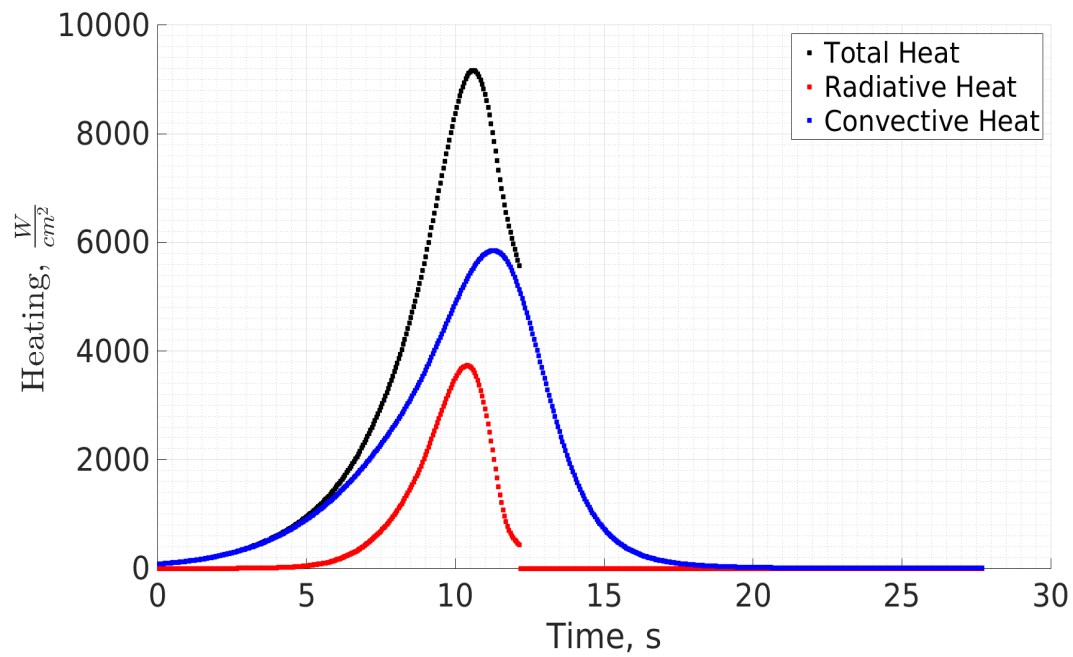


Figure 4.25: Heating-time profile for 10 cm body entering at -30 degrees and 12 km/s.

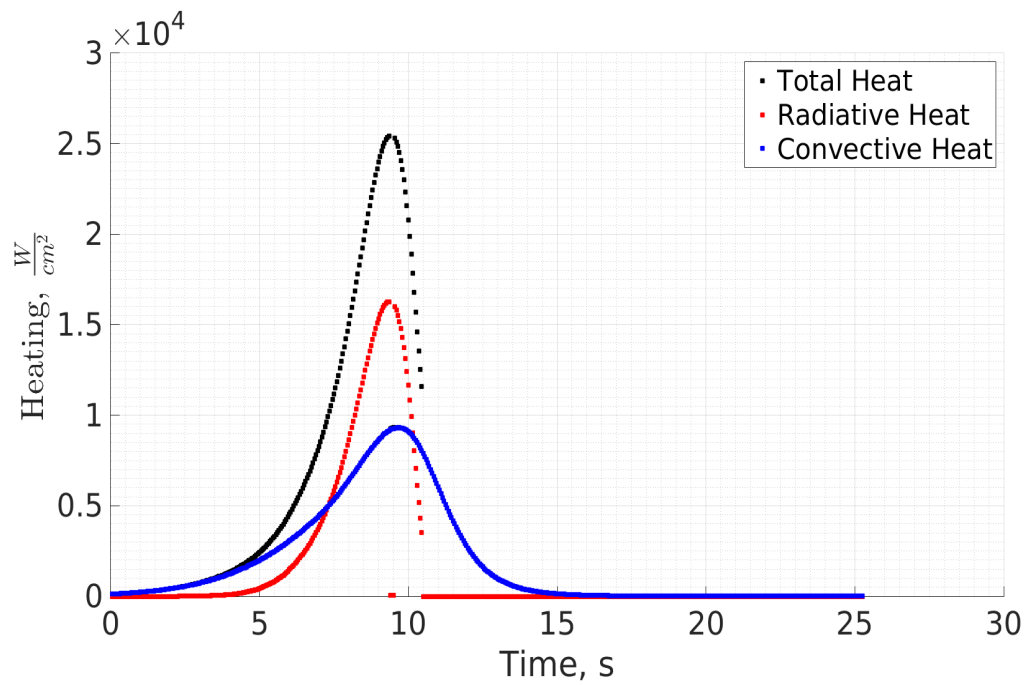


Figure 4.26: Heating-time profile for 10 cm body entering at -30 degrees and 14 km/s.

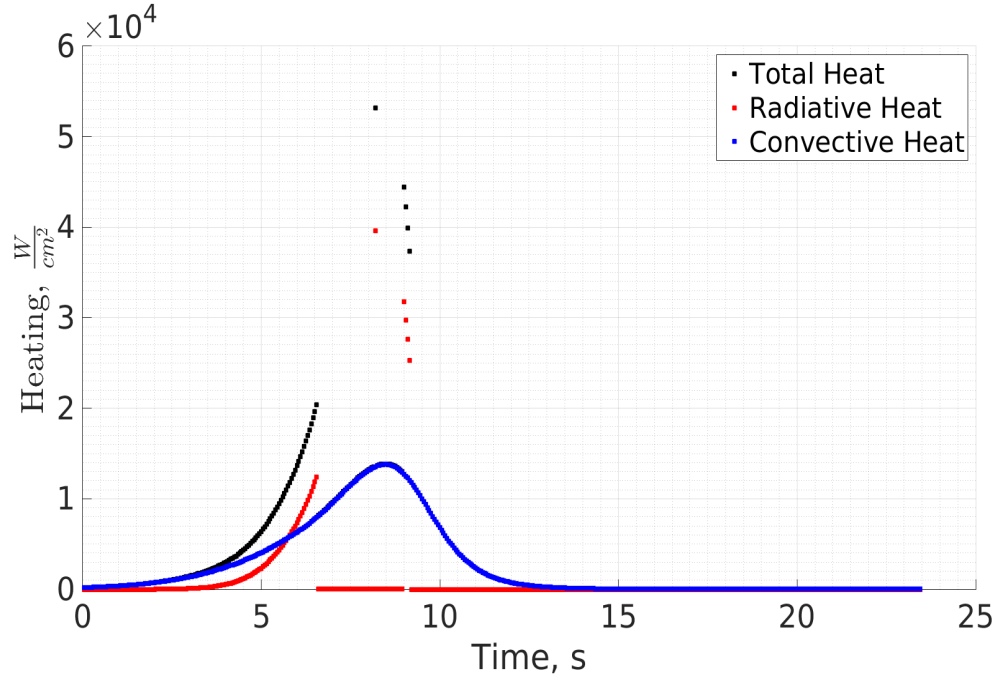


Figure 4.27: Heating-time profile for 10 cm body entering at -30 degrees and 16 km/s.

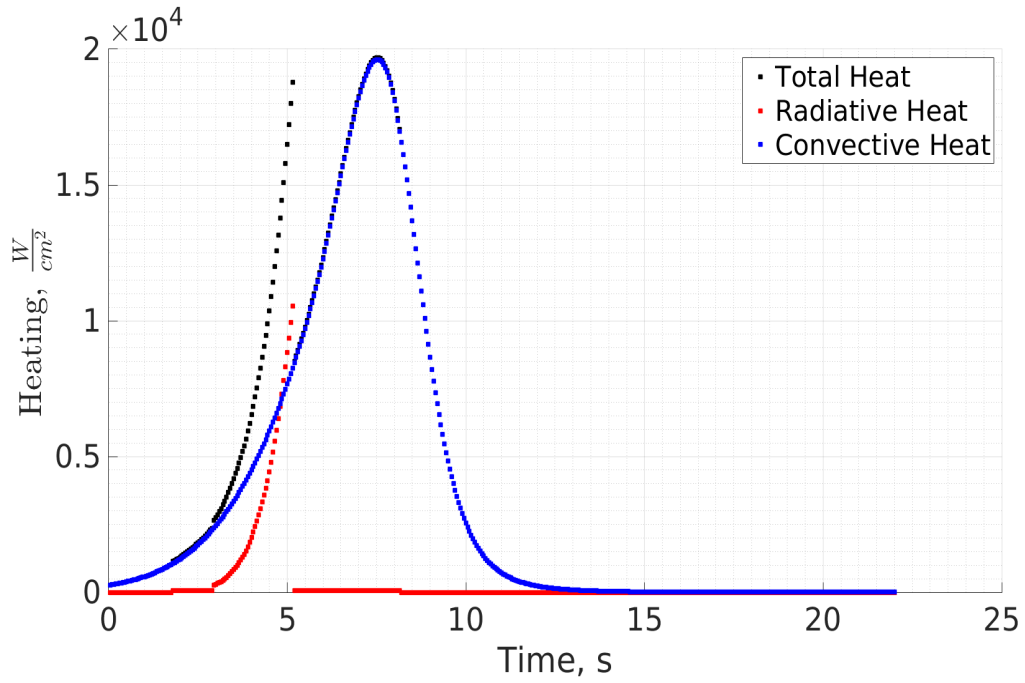


Figure 4.28: Heating-time profile for 10 cm body entering at -30 degrees and 18 km/s.

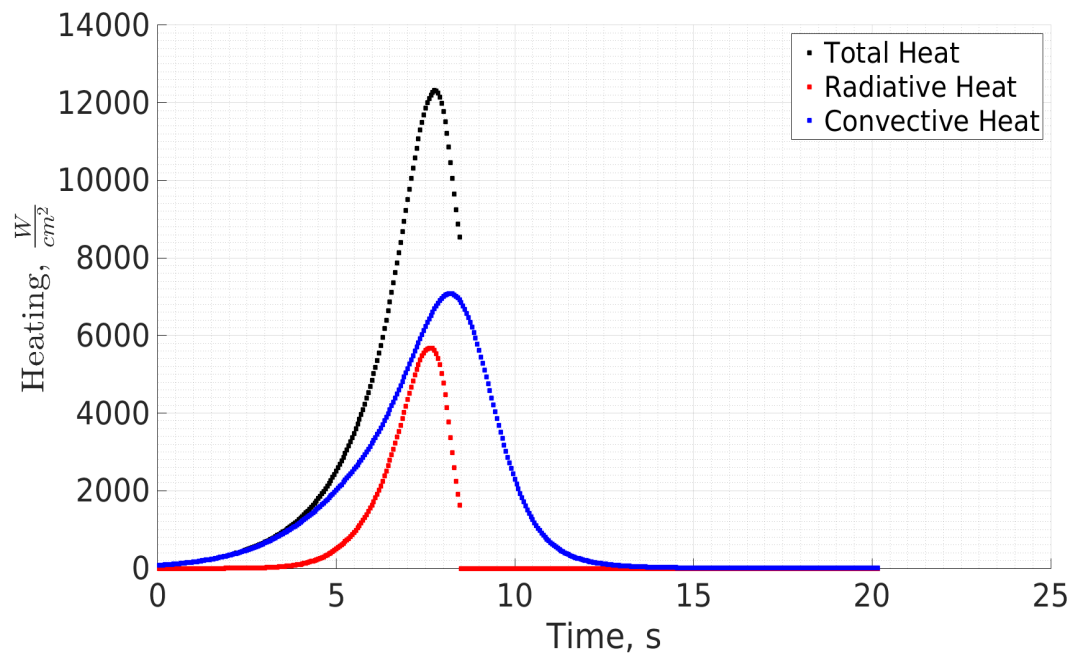


Figure 4.29: Heating-time profile for 10 cm body entering at -45 degrees and 12 km/s.

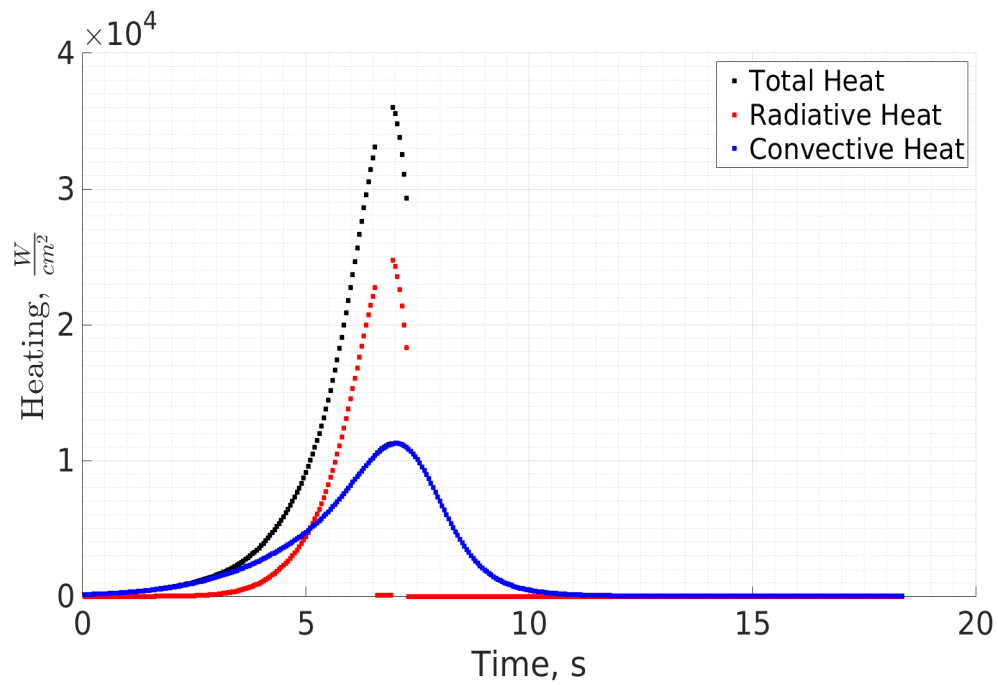


Figure 4.30: Heating-time profile for 10 cm body entering at -45 degrees and 14 km/s.

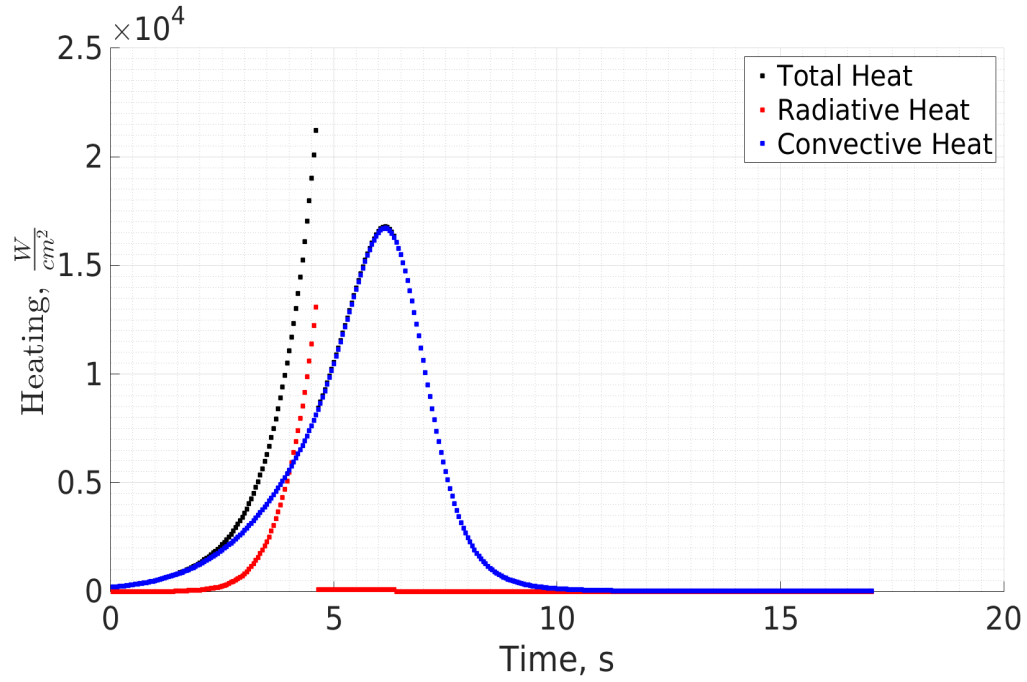


Figure 4.31: Heating-time profile for 10 cm body entering at -45 degrees and 16 km/s.

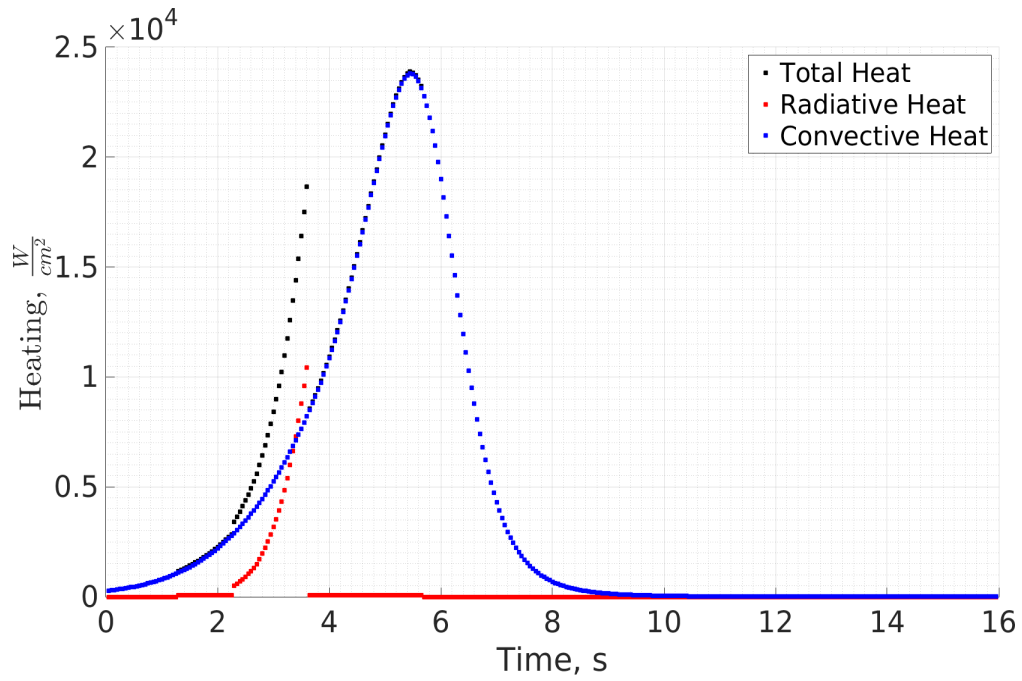


Figure 4.32: Heating-time profile for 10 cm body entering at -45 degrees and 18 km/s.

air density, body mass, and surface area, it will change as body mass is ablated and air density increases. The trajectories in Fig. 4.34 have terminal velocities of 191, 131, and $98 \frac{m}{s}$ at altitudes of 15, 10, and 5 km, respectively. The model reports the velocities for these bodies at 214, 140, and $101 \frac{m}{s}$, showing they are converging to terminal velocity.

Next, the plots that will be looked at are the altitude vs. stagnation temperature and the stagnation temperature vs. time for the 10 cm body. Figure 4.37 through 4.40 show that the gas in the shock layer is extremely hot during the early phase of the trajectory. While the bodies are high in the atmosphere they have very high velocities which result in strong shockwaves, and this causes intense heating in the shock layer. As the bodies reach lower altitudes, and lower velocities, the shock becomes weaker and produces lower stagnation temperatures. The trajectories in Fig. 4.37 and 4.38 have stagnation temperatures lower than the suggested meteoric vaporization temperature of 2,100 K starting at altitudes above 20 km. Comparing these plots to Fig. 4.41 through 4.44, it is clear that these trajectories spend a significant amount of time at temperatures lower than the 2,100 K limit. The 10 cm body trajectories shown here would be good candidates for the plausibility of late phase high-speed cooling.

The last trajectory characteristic of interest is the mass vs. velocity. This shows how much of the body ablates away during the trajectory and can be seen in Fig. 4.45 through 4.48. For most of the trajectories the figures reveal smooth curves but others show non-physical characteristics. In Fig. 4.45, the 10 cm body entering at -8 degrees with a entry velocity of $18 \frac{km}{s}$ shows some anomalies in the data. These anomalies occur when the trajectory is out of the range of the radiation curve fit. In these instances, the only heating applied to the body is convective heating. If radiation is a significant heat source for a given trajectory, the calculated ablated mass will be largely impacted when the trajectory moves out of range of the `qfit` subroutine. This supports the reasoning of omitting trajectories that have large gaps in the radiative heating data.

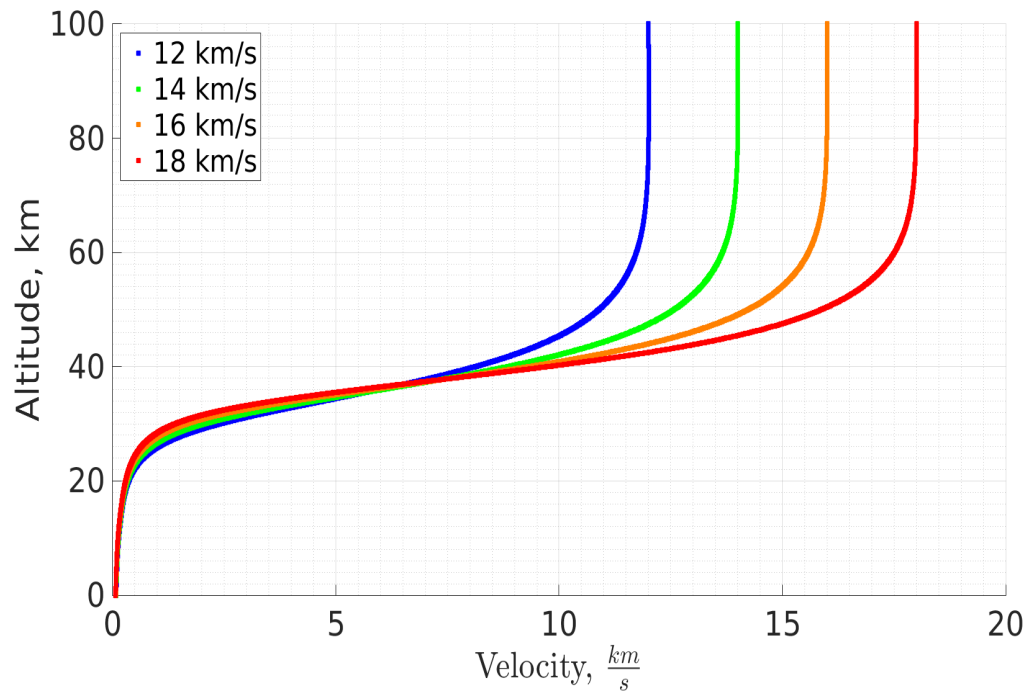


Figure 4.33: Altitude-velocity profile for 10 cm body entering at -8 degrees.

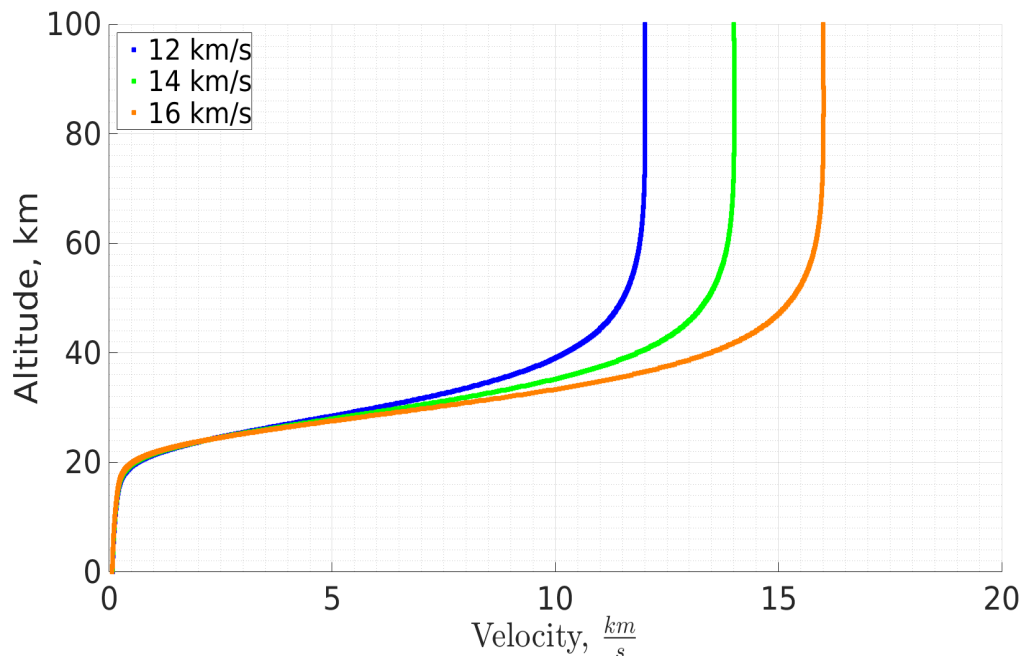


Figure 4.34: Altitude-velocity profile for 10 cm body entering at -15 degrees.

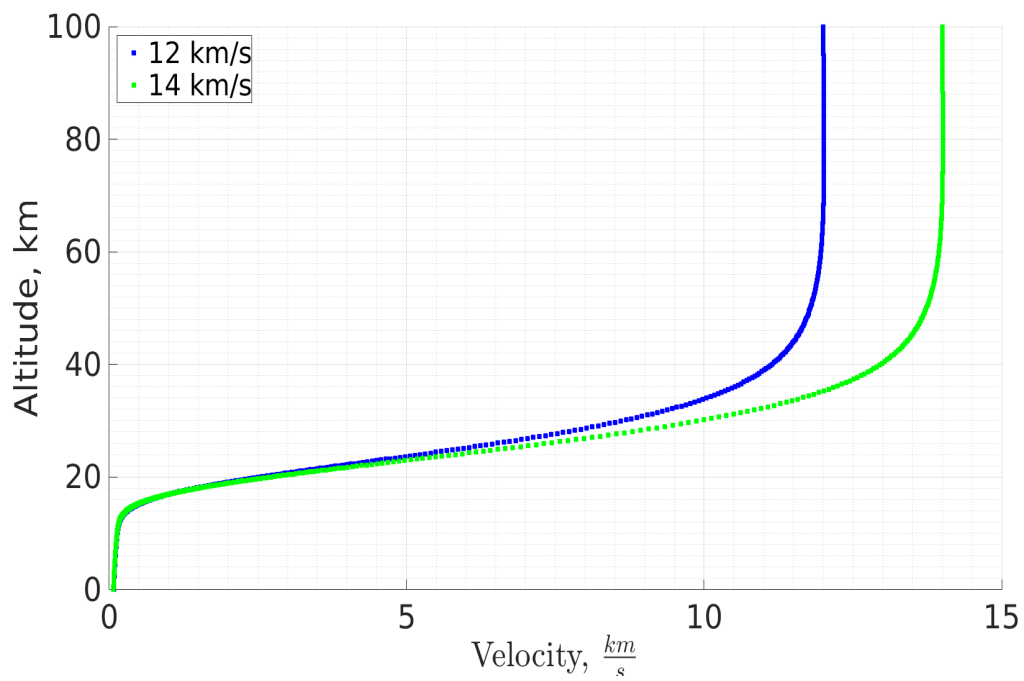


Figure 4.35: Altitude-velocity profile for 10 cm body entering at -30 degrees.

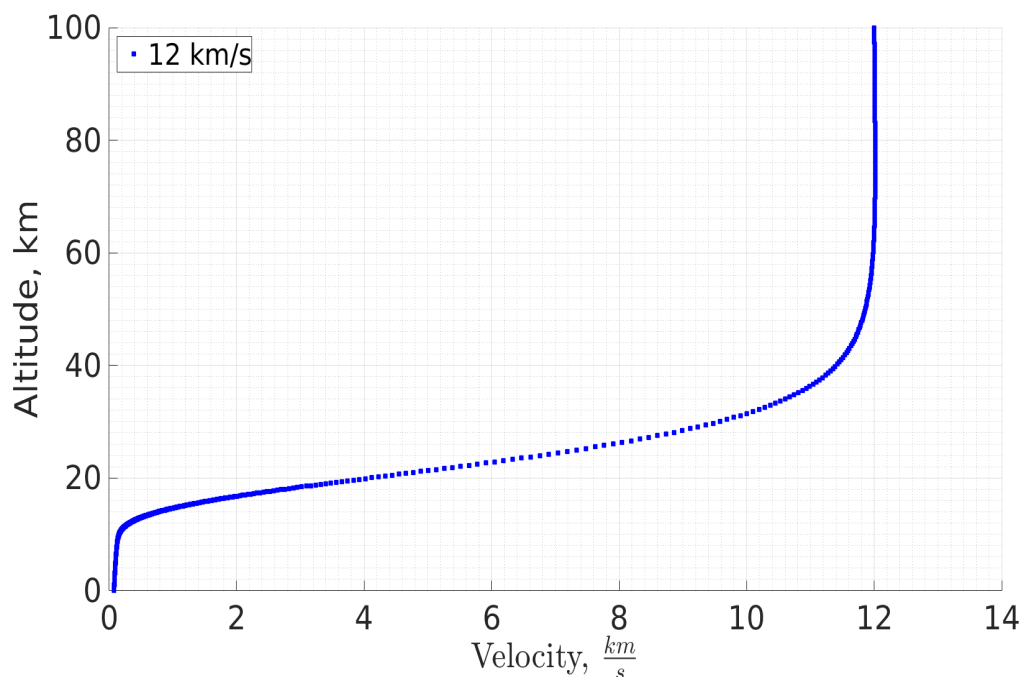


Figure 4.36: Altitude-velocity profile for 10 cm body entering at -45 degrees.

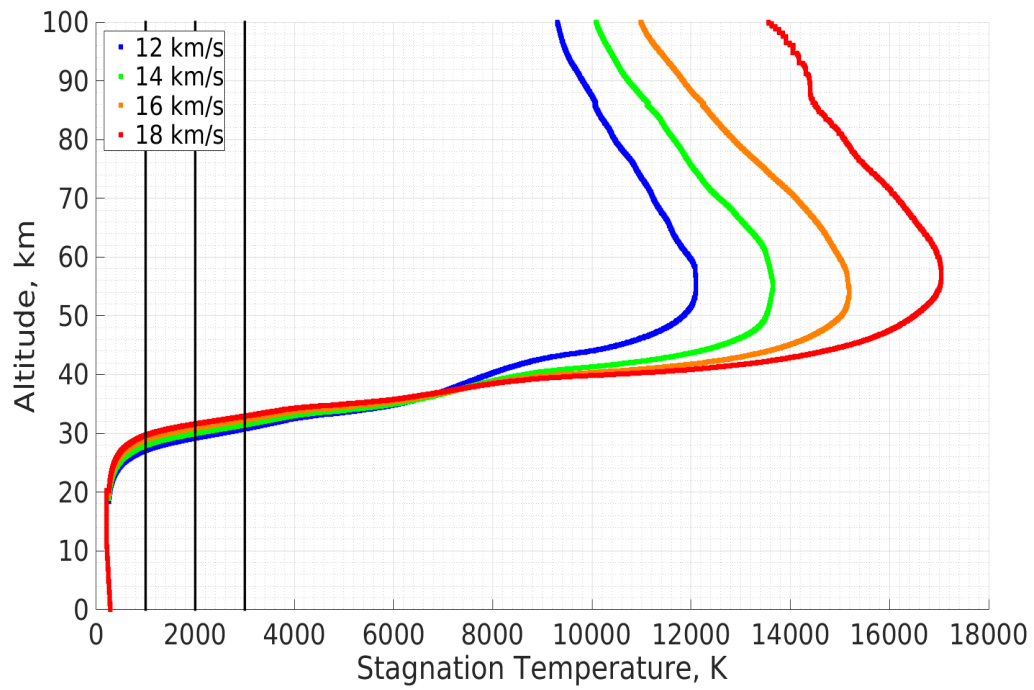


Figure 4.37: Altitude-stagnation temperature profile for 10 cm body entering at -8 degrees.

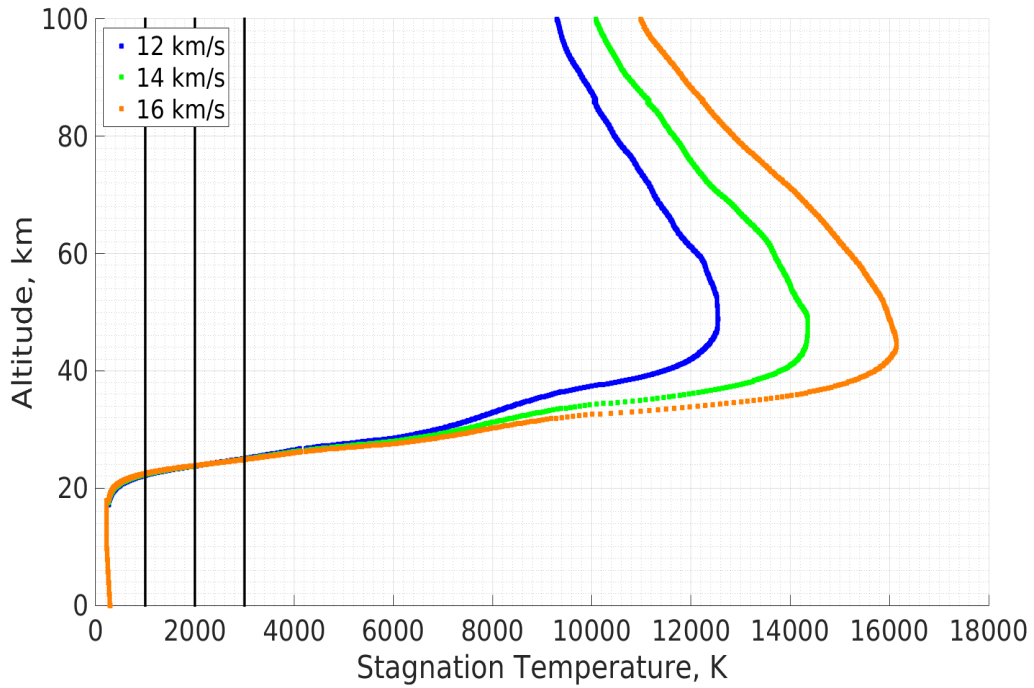


Figure 4.38: Altitude-stagnation temperature profile for 10 cm body entering at -15 degrees.

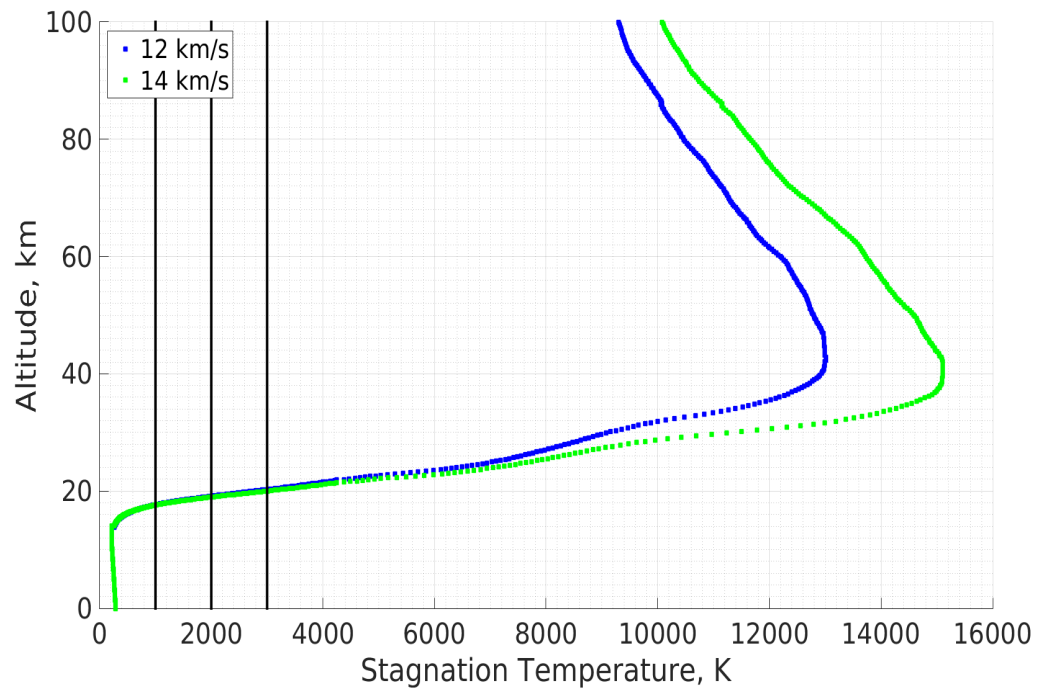


Figure 4.39: Altitude-stagnation temperature profile for 10 cm body entering at -30 degrees.

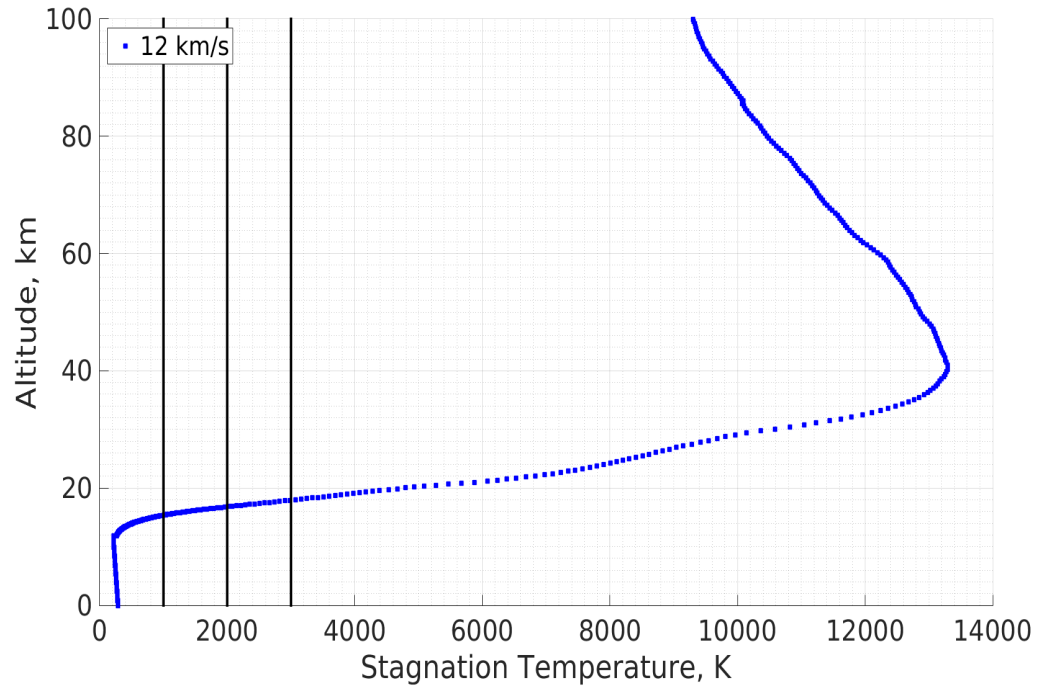


Figure 4.40: Altitude-stagnation temperature profile for 10 cm body entering at -45 degrees.

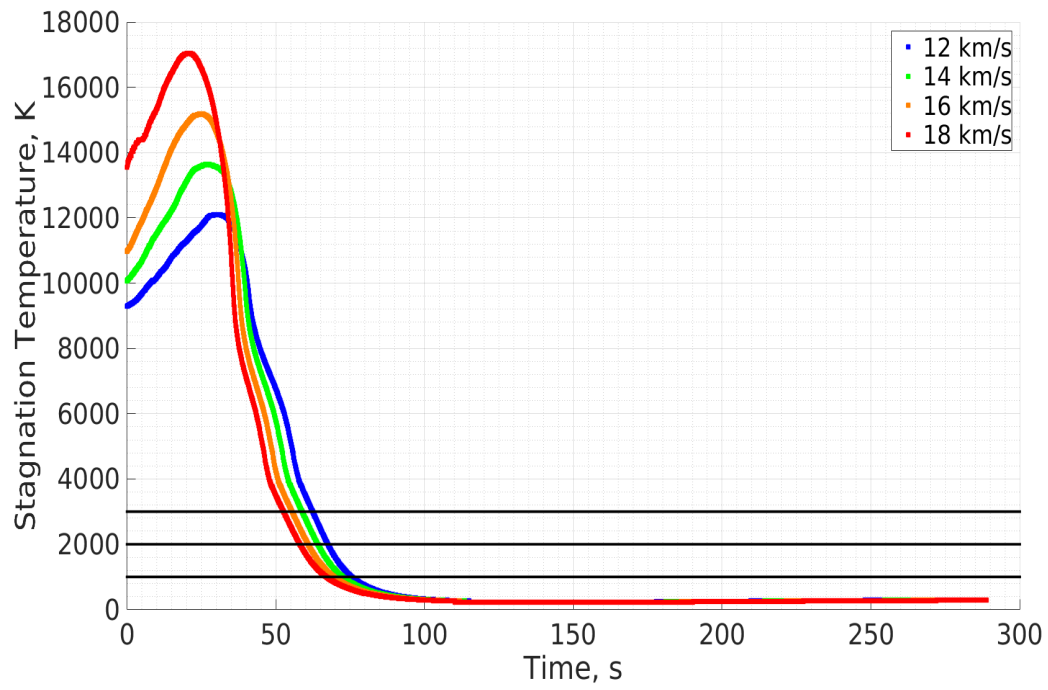


Figure 4.41: Stagnation temperature-time profile for 10 cm body entering at -8 degrees.

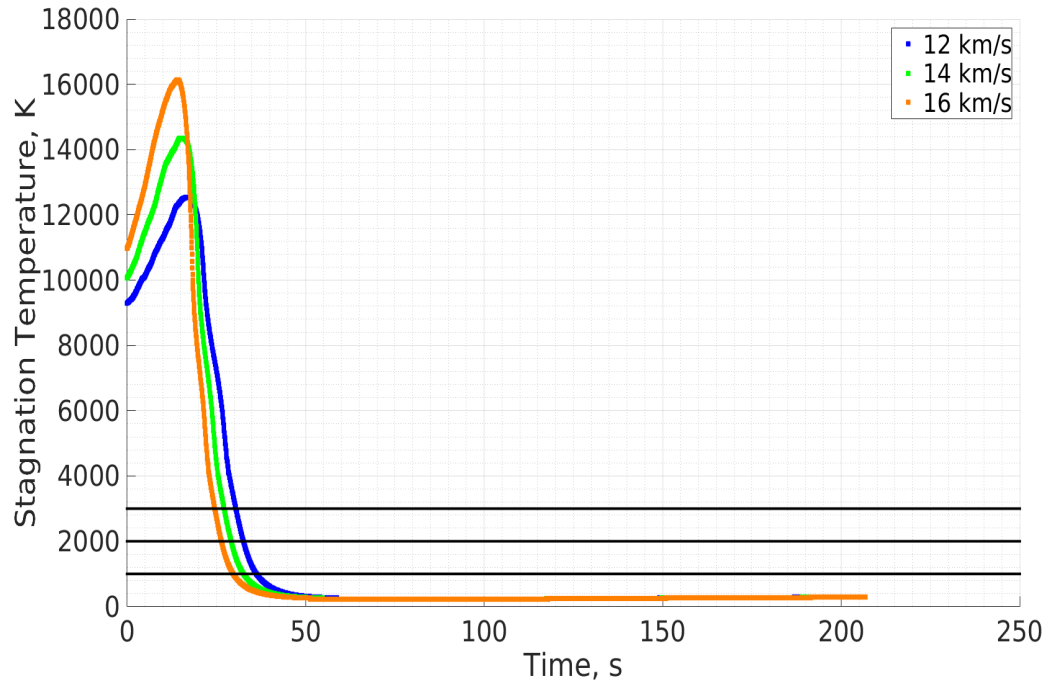


Figure 4.42: Stagnation temperature-time profile for 10 cm body entering at -15 degrees.

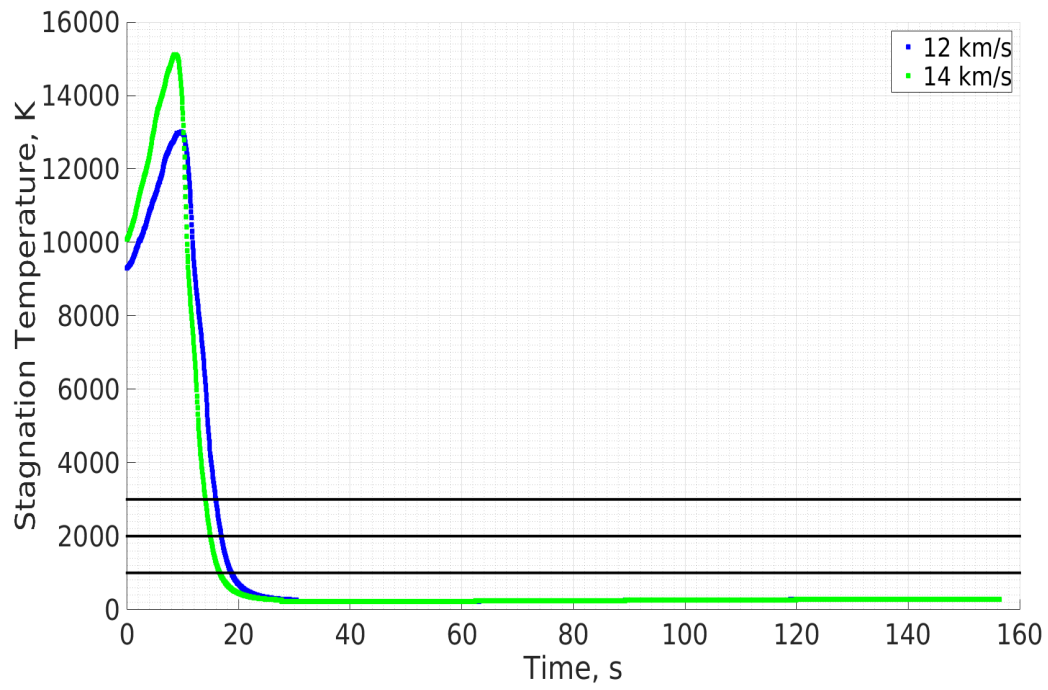


Figure 4.43: Stagnation temperature-time profile for 10 cm body entering at -30 degrees.

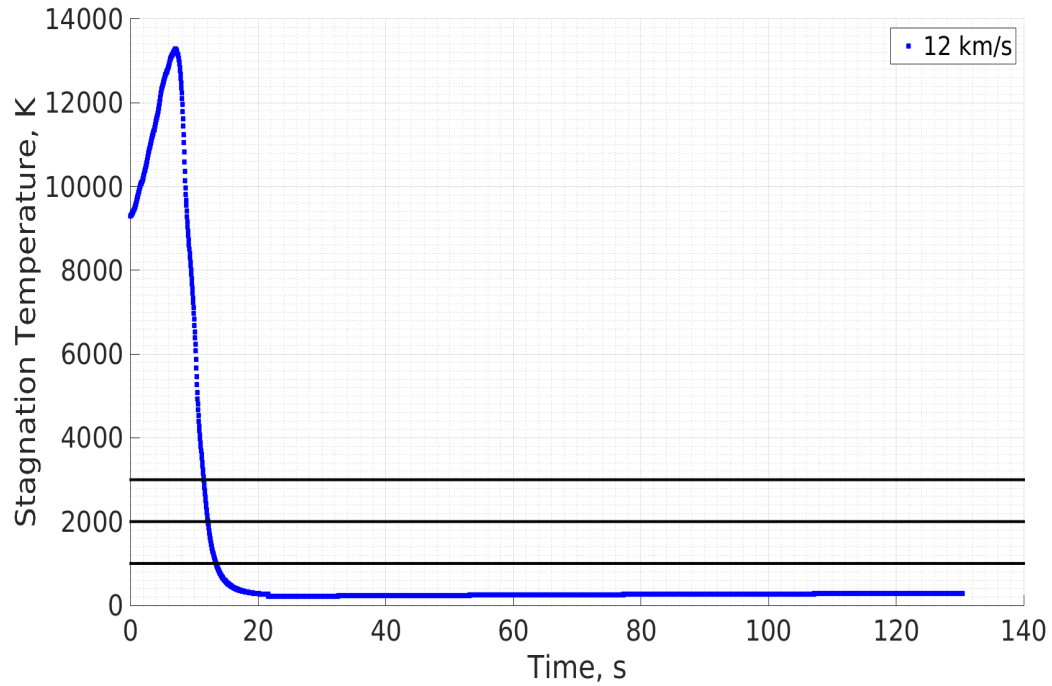


Figure 4.44: Stagnation temperature-time profile for 10 cm body entering at -45 degrees.

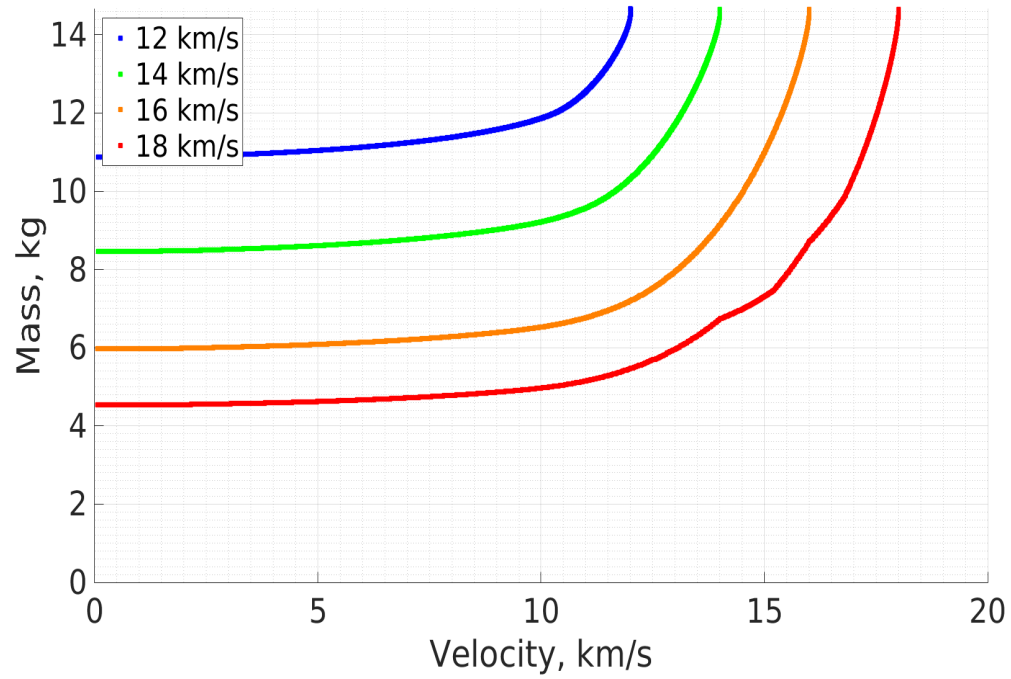


Figure 4.45: Mass-velocity profile for 10 cm body entering at -8 degrees.

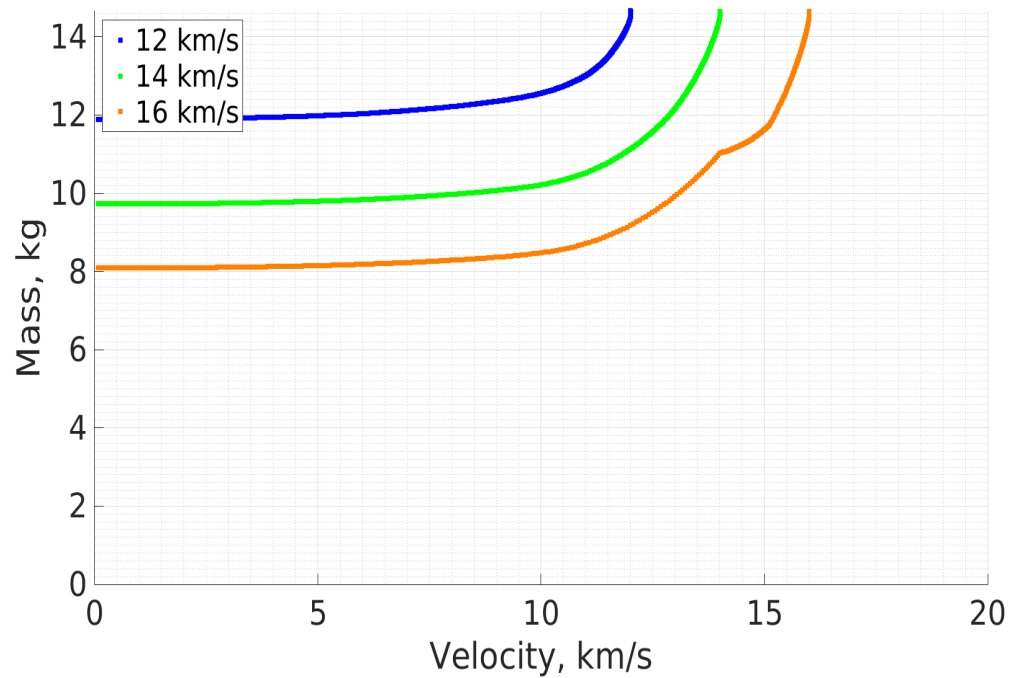


Figure 4.46: Mass-velocity profile for 10 cm body entering at -15 degrees.

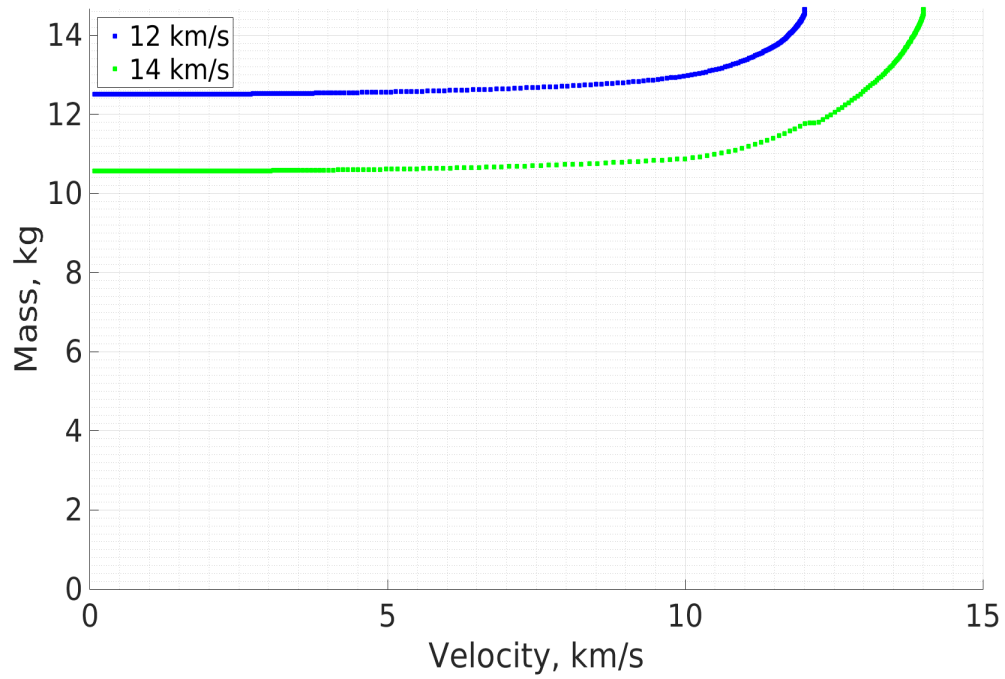


Figure 4.47: Mass-velocity profile for 10 cm body entering at -30 degrees.

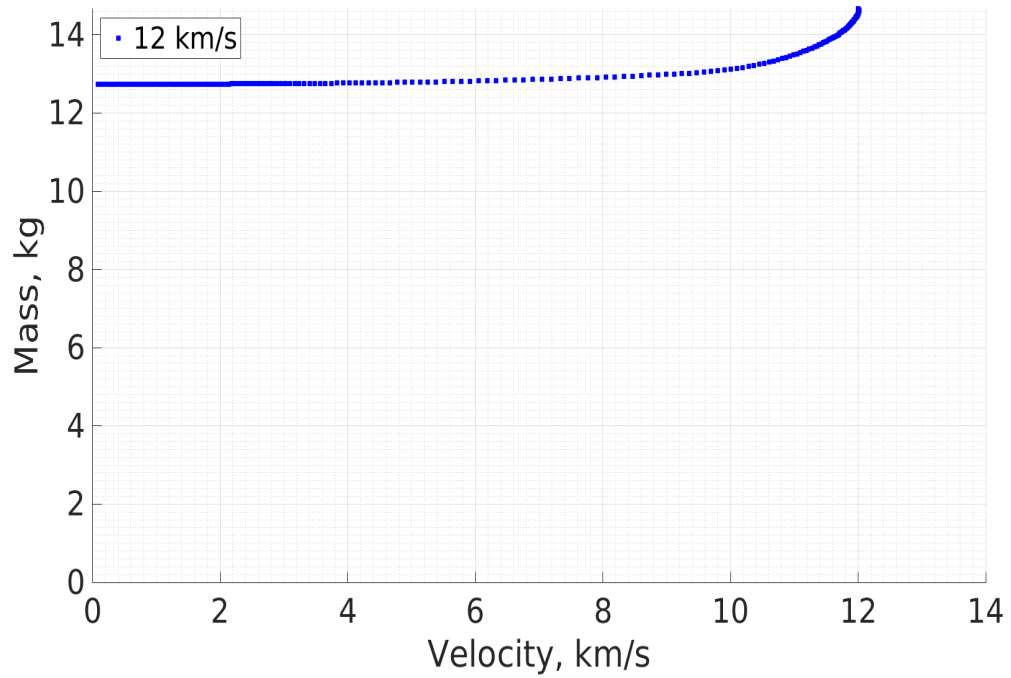


Figure 4.48: Mass-velocity profile for 10 cm body entering at -45 degrees.

4.4 50 cm Nose Radius

Moving up to the 50 cm body results in similar trends as the those discussed for the 10cm body. However, the range of entry conditions for which the model is applicable decreases. The heating plots for the 50 cm body are shown in Fig. 4.49 through 4.64. Figures 4.49 and 4.50 show that the model was able to capture the majority of the radiative heating and provides reasonable data. Figures 4.51 and 4.53 have heating plots with some gaps, but the model was able to catch the peak radiative heating for each trajectory and may be good candidates for developing an iterative technique to bridge the gaps in these data. The rest of the figures showing heating for the 50 cm body show too much missing data to have reliable trajectory information. This means that the model has a differing range of entry parameters for the 50 cm body. With an -8 degree entry angle, the max velocity is somewhere between 14 and 16 $\frac{km}{s}$, and for -15 degree entry angle, the max velocity is about 12 $\frac{km}{s}$. As the body size increases, the range of applicability of the model decreases because the bodies will be moving faster at lower altitudes.

The altitude vs. velocity plots, Fig. 4.65 and Fig. 4.66, show that the larger sized bodies keep more of their kinetic energy deeper in the atmosphere and this is a contributing factor in pushing the trajectories out of the range of the `qfit` subroutine. Also, because they are traveling faster at low altitudes, they do not spend as much time below the 2,100 K temperature threshold and may not experience as much of the hypothesized high-speed convective cooling. The stagnation temperature profiles are in Fig. 4.67 through Fig. 4.70. The preservation of kinetic energy also causes these larger bodies to experience larger heat loads and large amounts of mass being ablated and this is reflected when looking at Fig. 4.71 and Fig. 4.72.

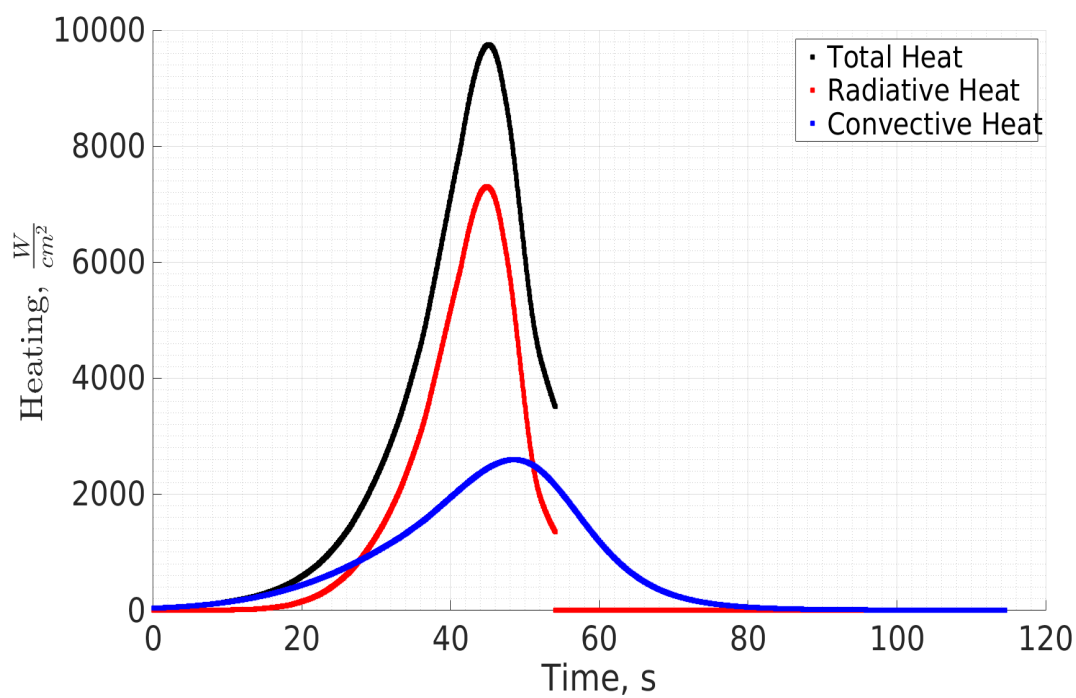


Figure 4.49: Heating-time profile for 50 cm body entering at -8 degrees and 12 km/s.

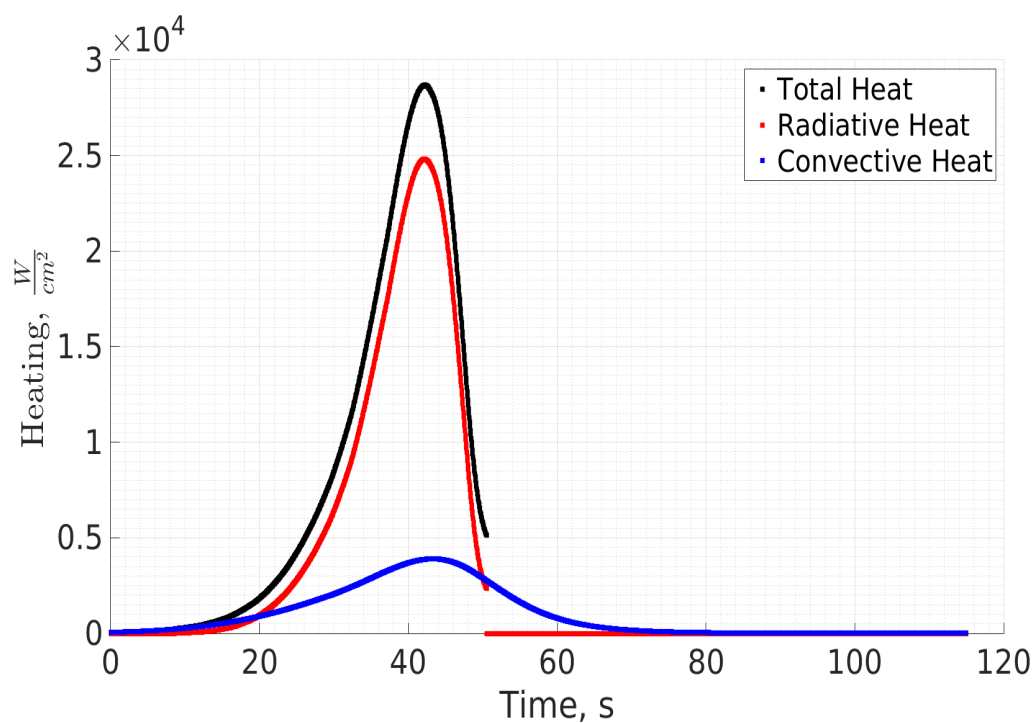


Figure 4.50: Heating-time profile for 50 cm body entering at -8 degrees and 14 km/s.

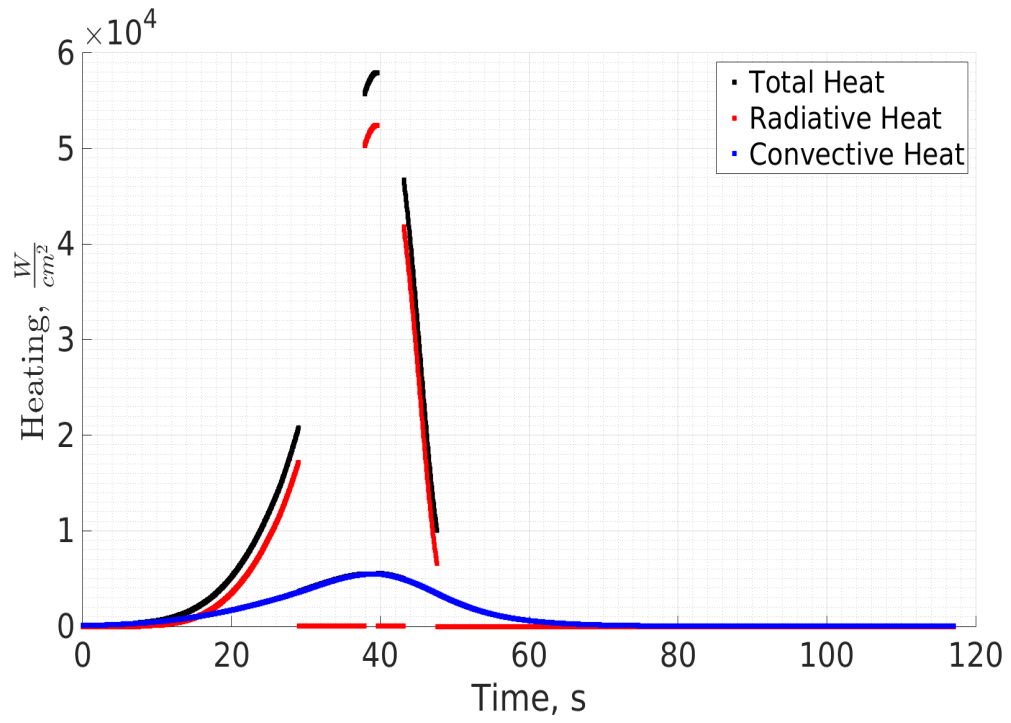


Figure 4.51: Heating-time profile for 50 cm body entering at -8 degrees and 16 km/s.

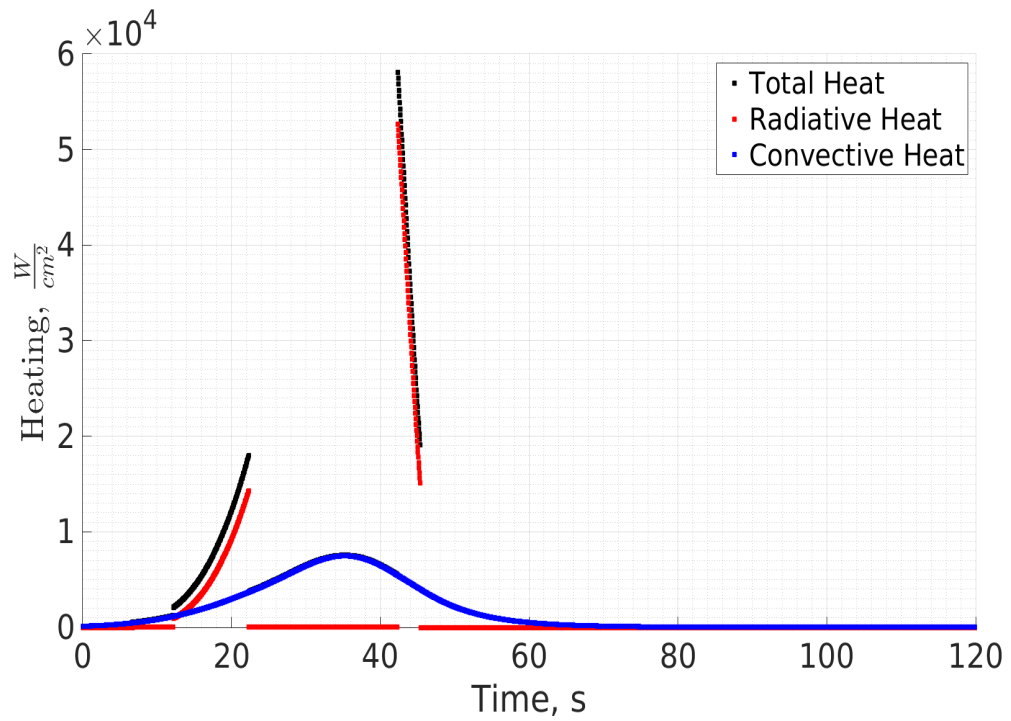


Figure 4.52: Heating-time profile for 50 cm body entering at -8 degrees and 18 km/s.

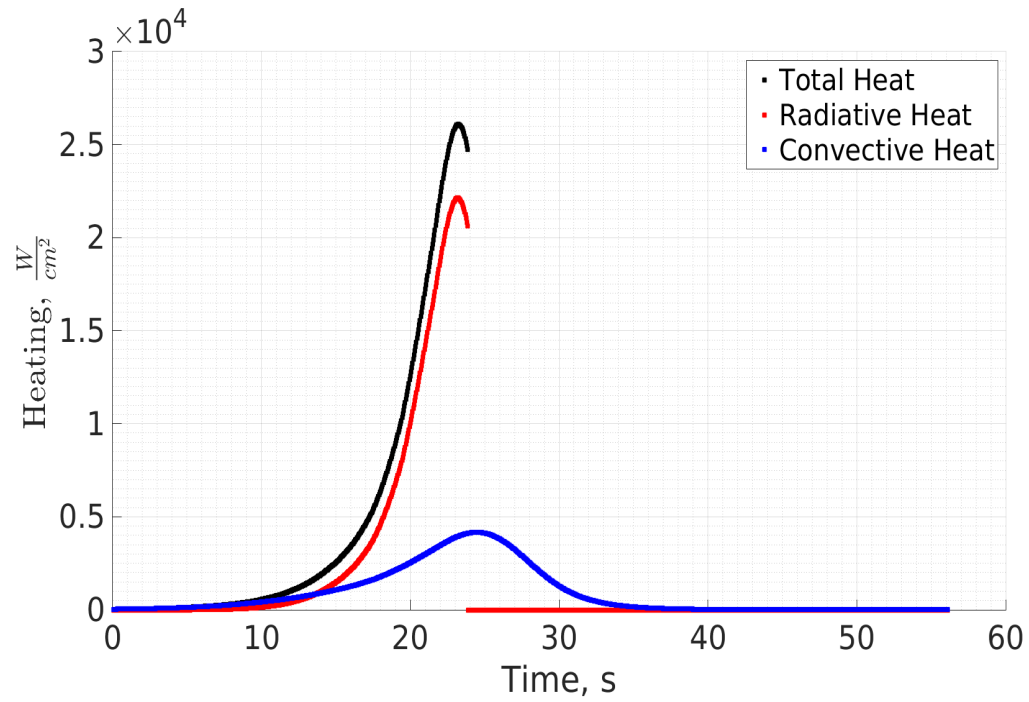


Figure 4.53: Heating-time profile for 50 cm body entering at -15 degrees and 12 km/s.

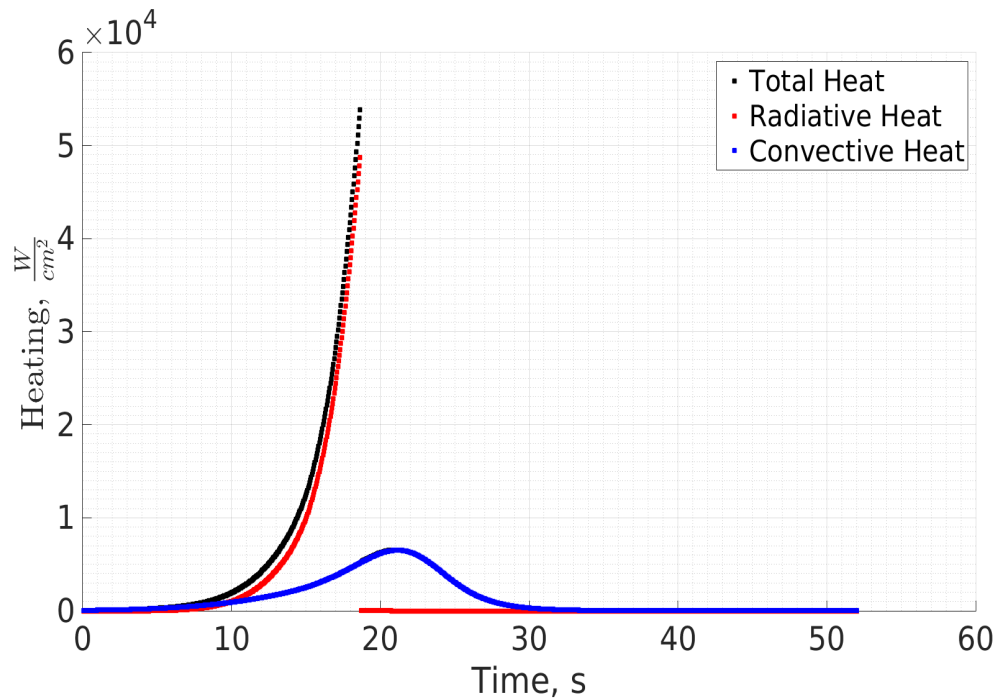


Figure 4.54: Heating-time profile for 50 cm body entering at -15 degrees and 14 km/s.

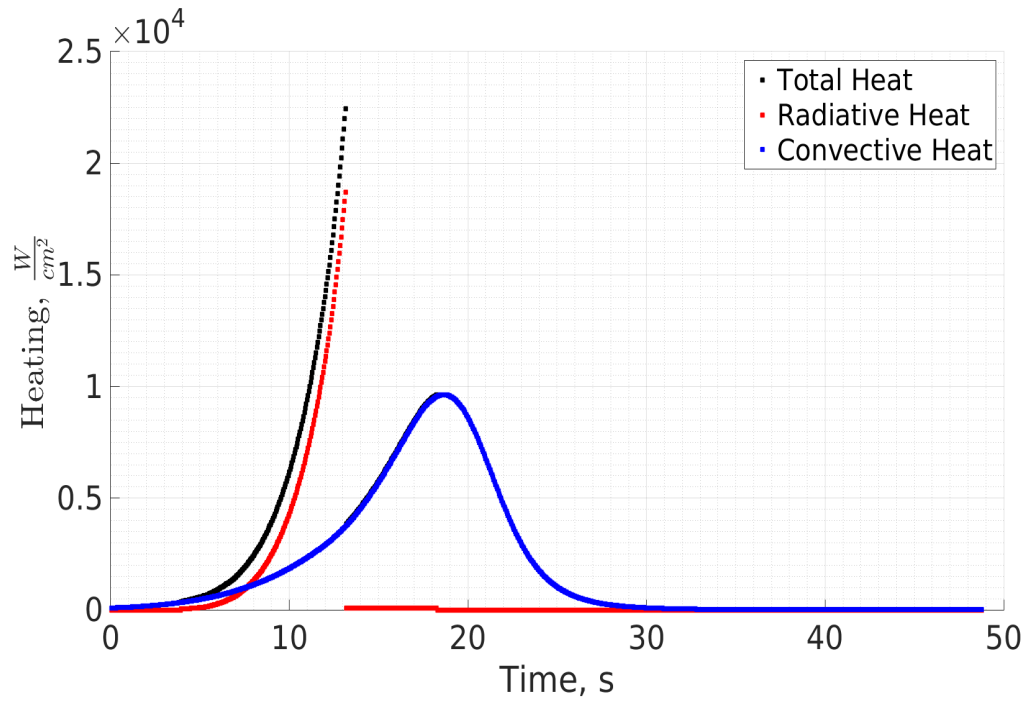


Figure 4.55: Heating-time profile for 50 cm body entering at -15 degrees and 16 km/s.

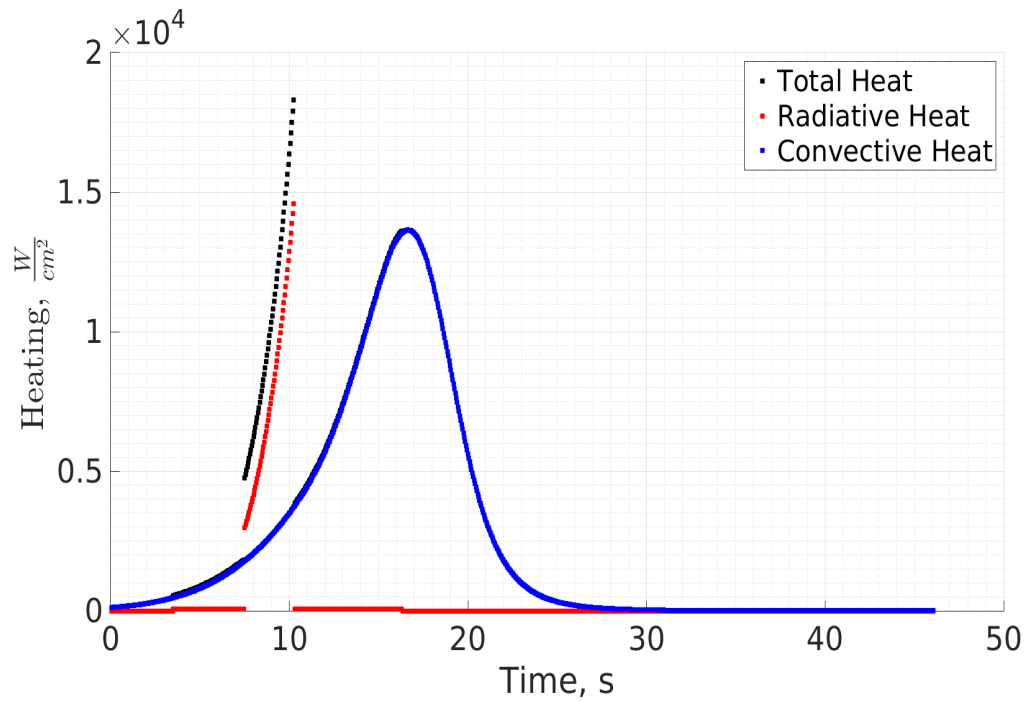


Figure 4.56: Heating-time profile for 50 cm body entering at -15 degrees and 18 km/s.

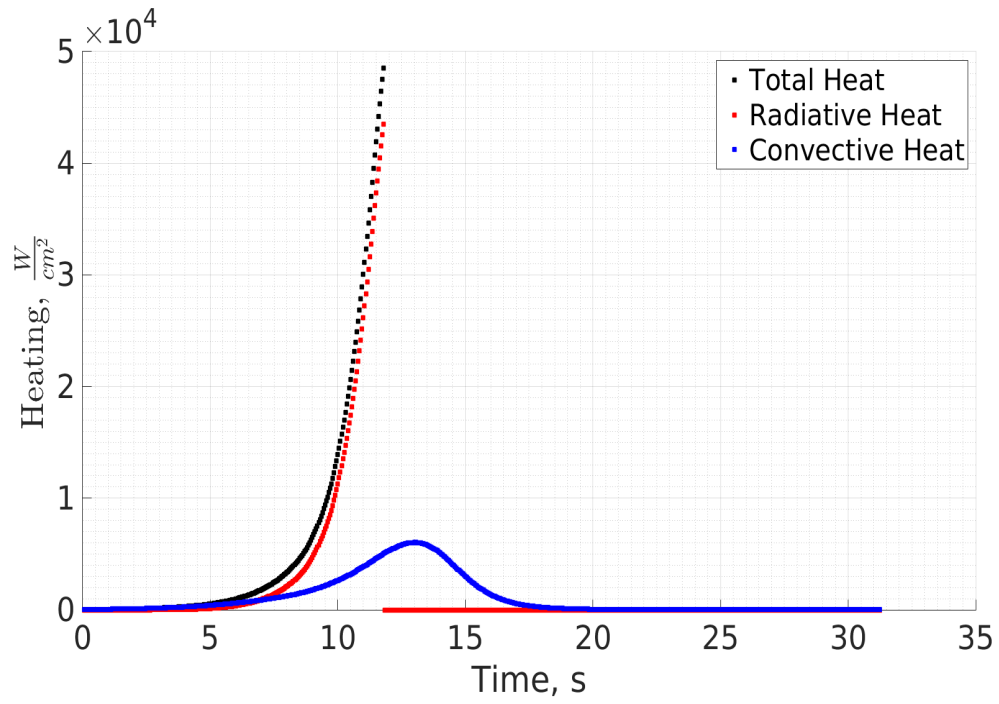


Figure 4.57: Heating-time profile for 50 cm body entering at -30 degrees and 12 km/s.

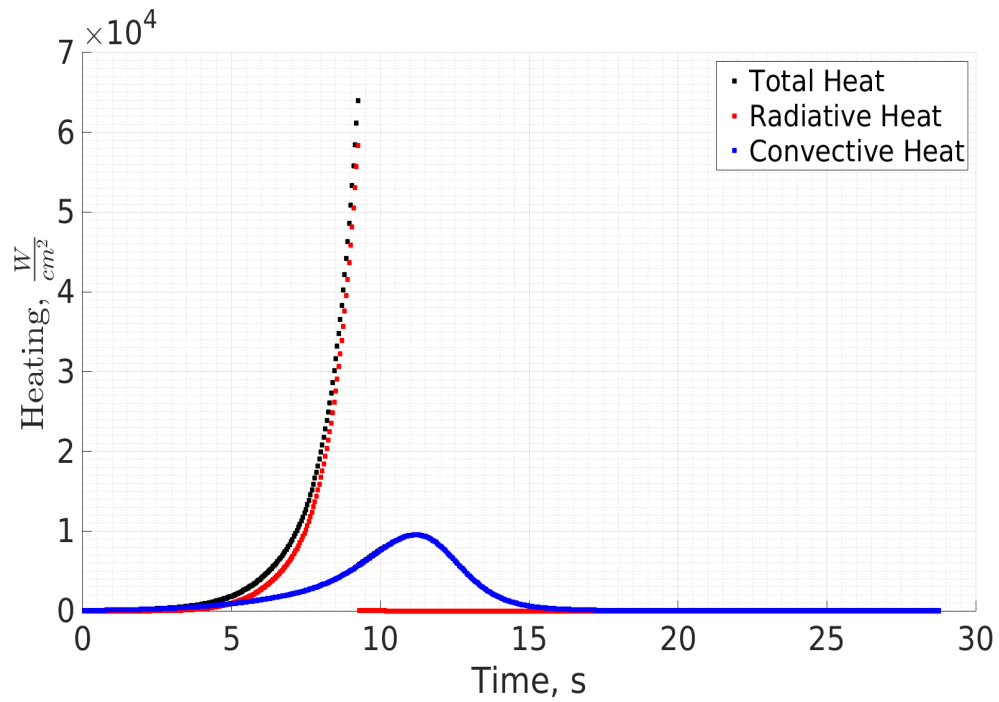


Figure 4.58: Heating-time profile for 50 cm body entering at -30 degrees and 14 km/s.

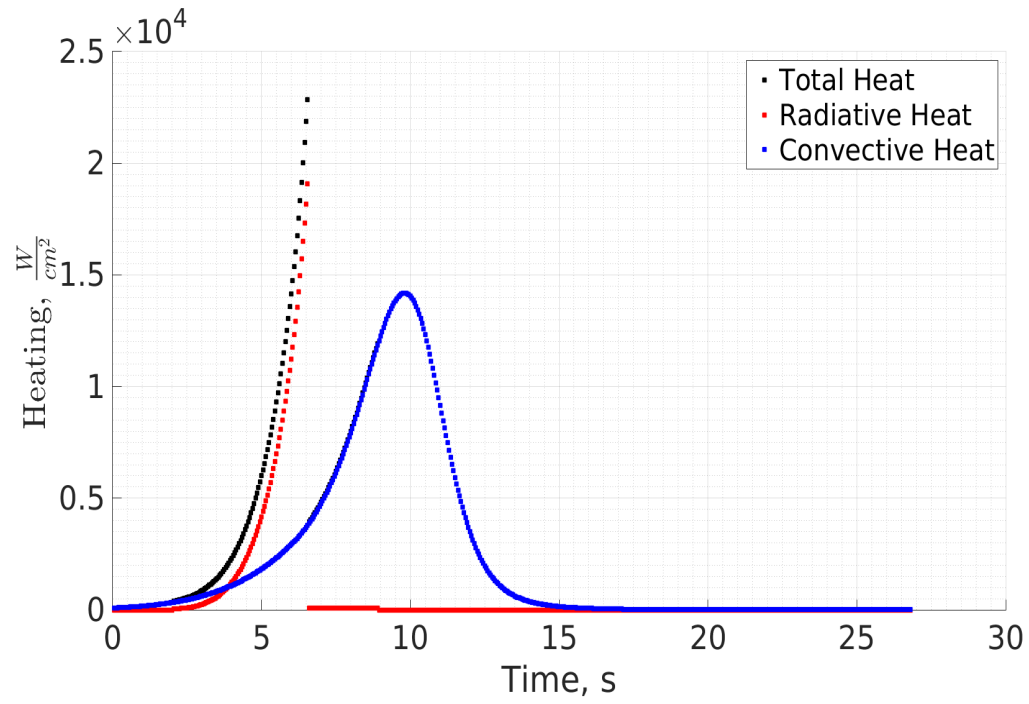


Figure 4.59: Heating-time profile for 50 cm body entering at -30 degrees and 16 km/s.

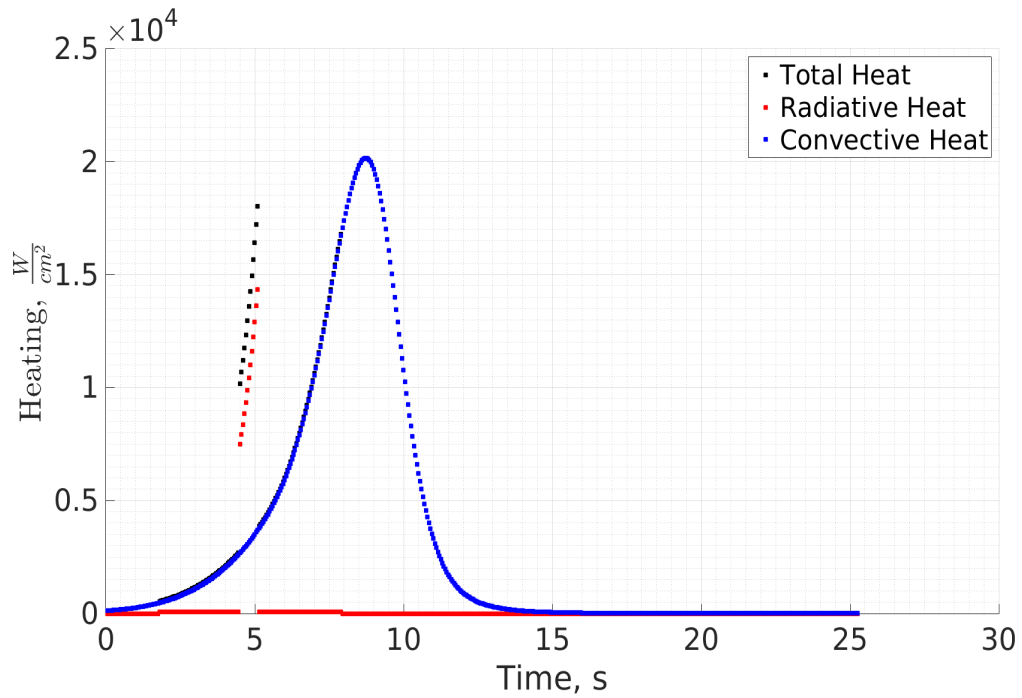


Figure 4.60: Heating-time profile for 50 cm body entering at -30 degrees and 18 km/s.

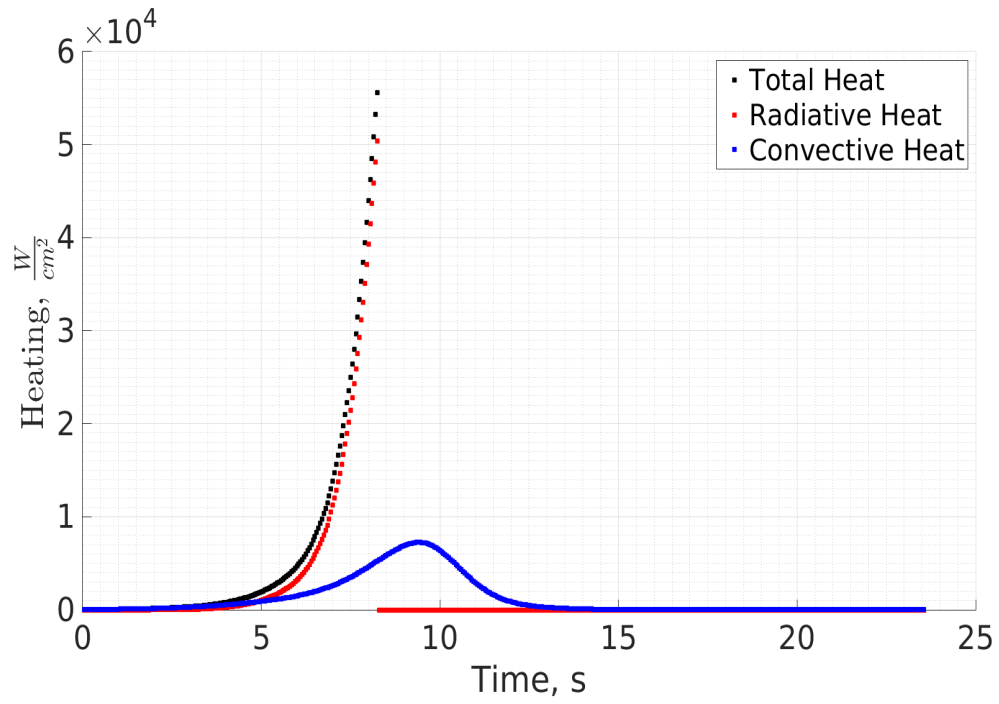


Figure 4.61: Heating-time profile for 50 cm body entering at -45 degrees and 12 km/s.

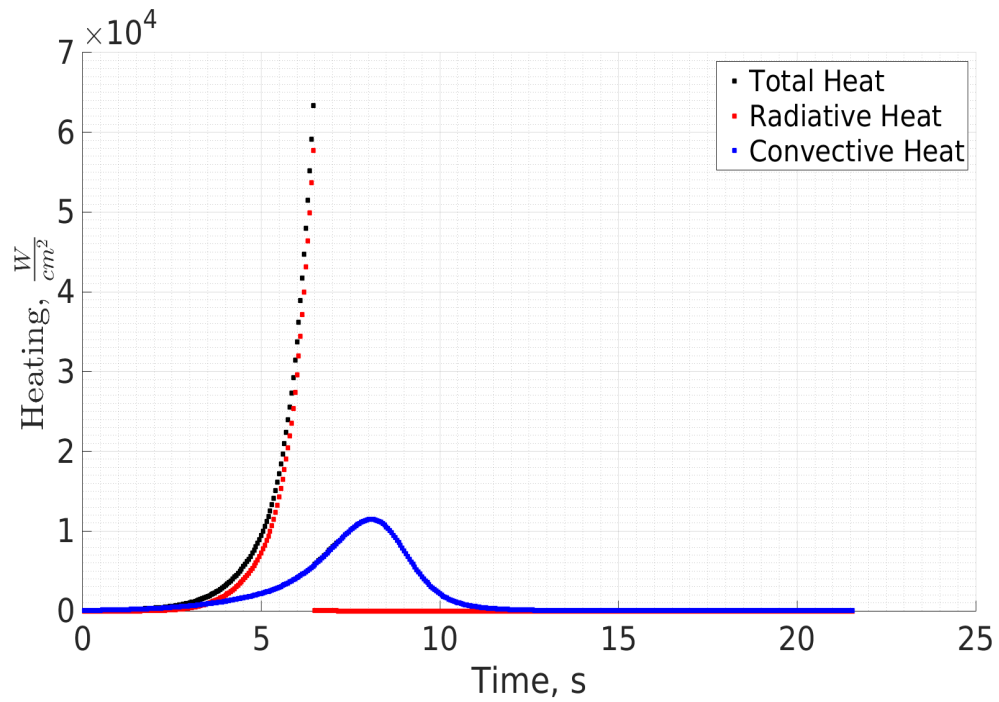


Figure 4.62: Heating-time profile for 50 cm body entering at -45 degrees and 14 km/s.

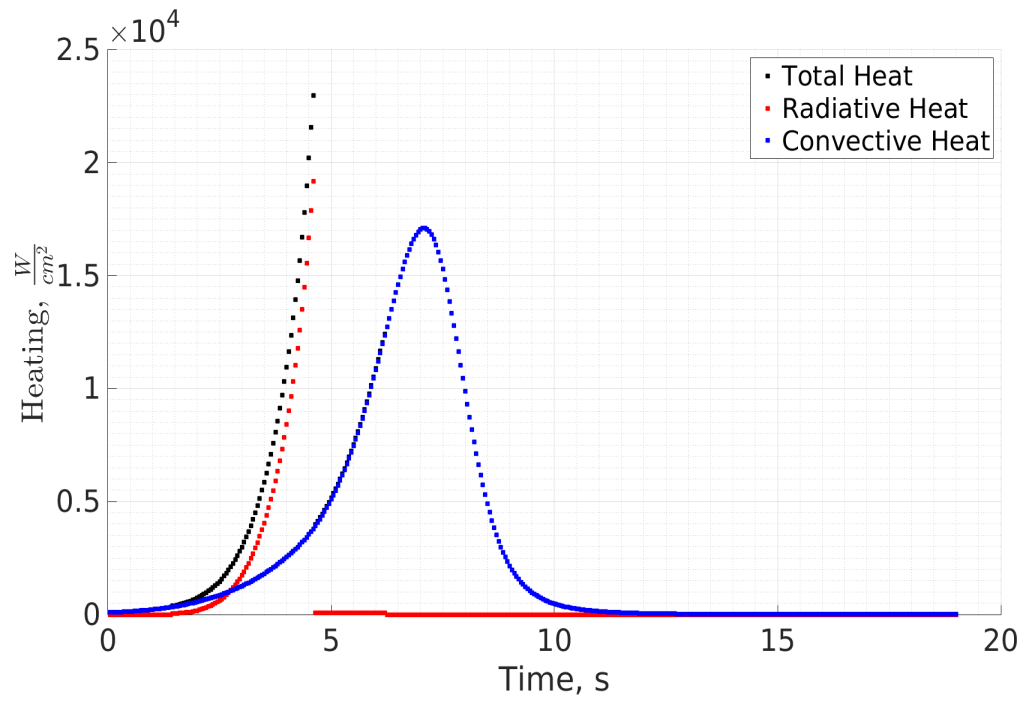


Figure 4.63: Heating-time profile for 50 cm body entering at -45 degrees and 16 km/s.

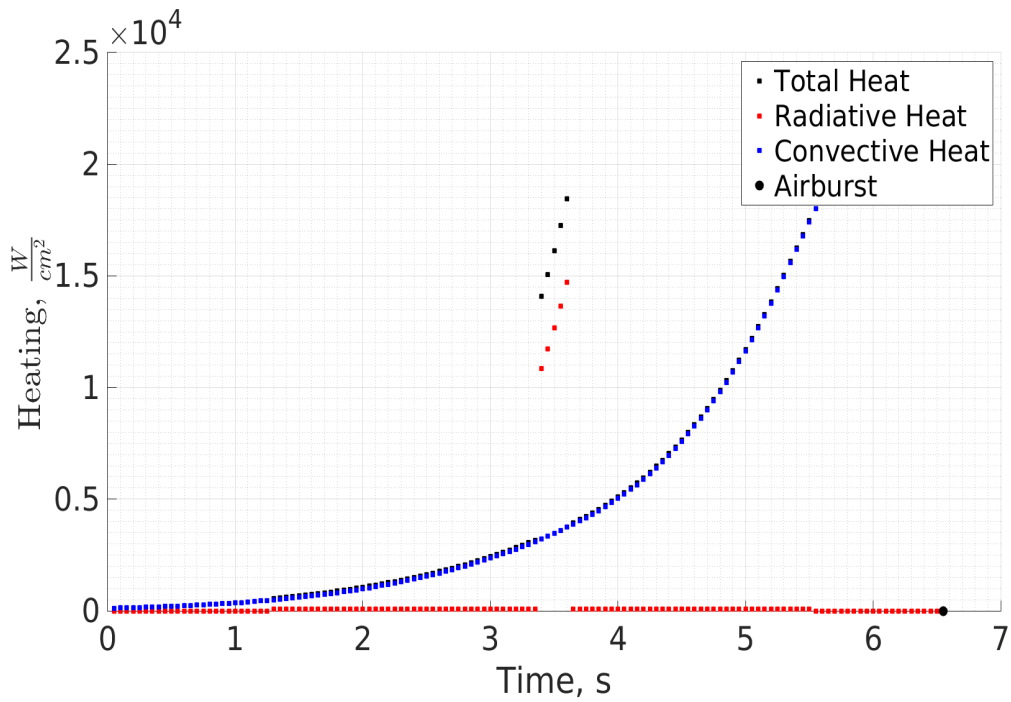


Figure 4.64: Heating-time profile for 50 cm body entering at -45 degrees and 18 km/s.

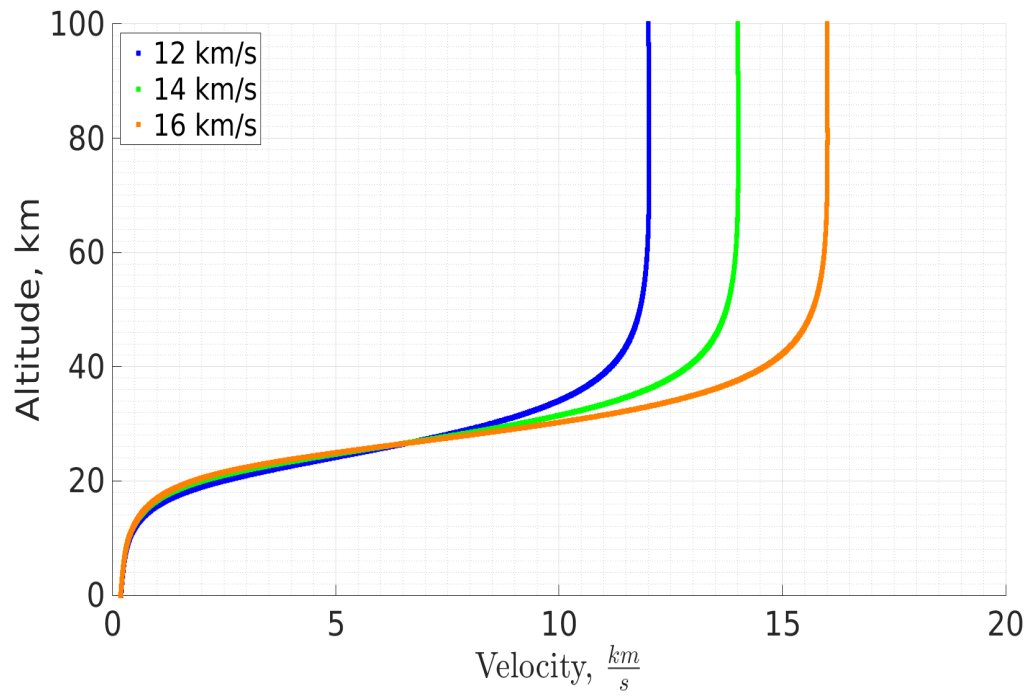


Figure 4.65: Altitude-velocity profile for 50 cm body entering at -8 degrees.

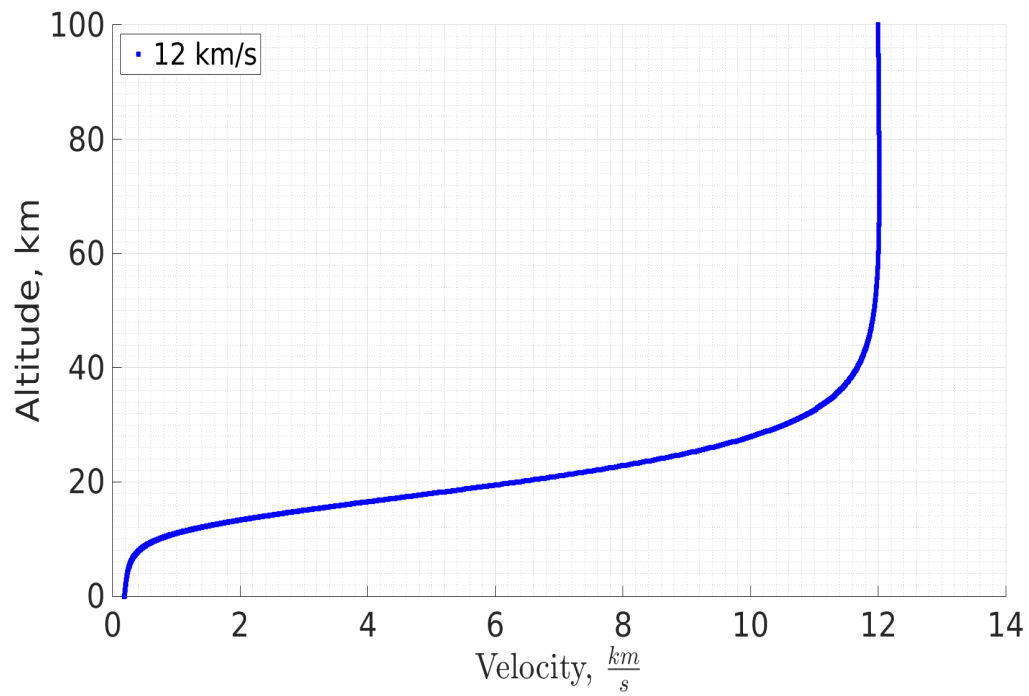


Figure 4.66: Altitude-velocity profile for 50 cm body entering at -15 degrees.

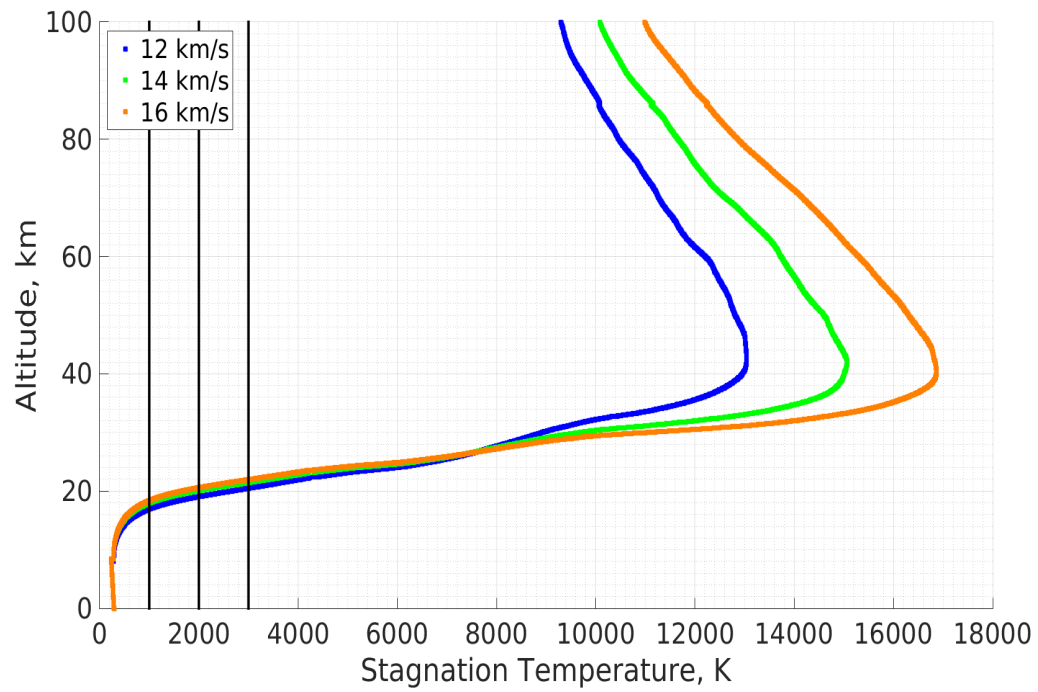


Figure 4.67: Altitude-stagnation temperature profile for 50 cm body entering at -8 degrees.

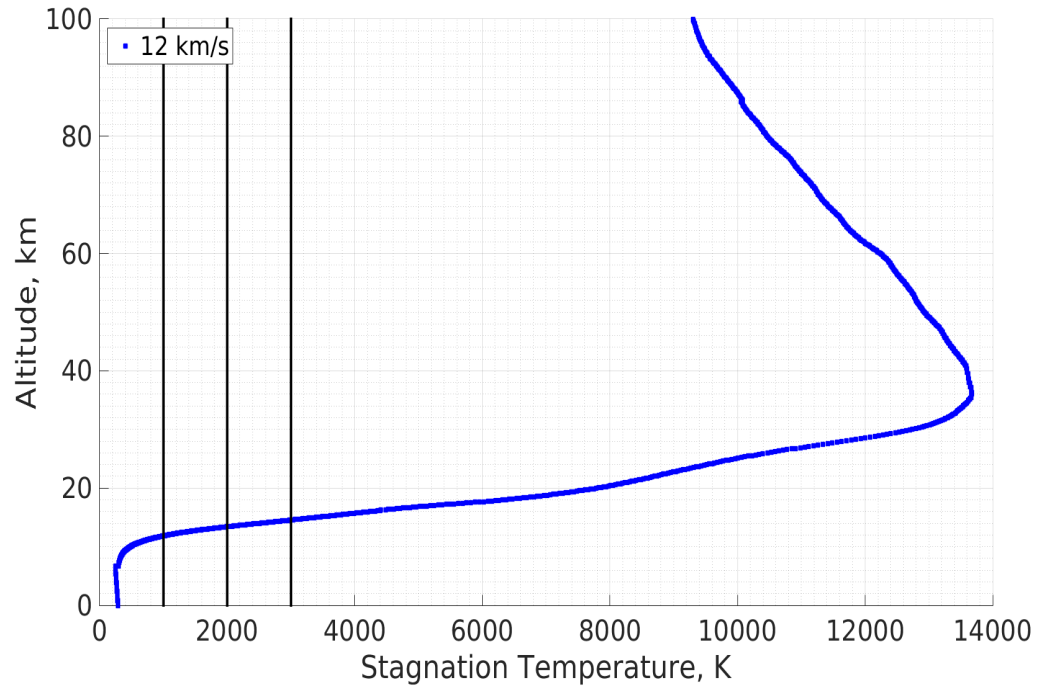


Figure 4.68: Altitude-stagnation temperature profile for 50 cm body entering at -15 degrees.

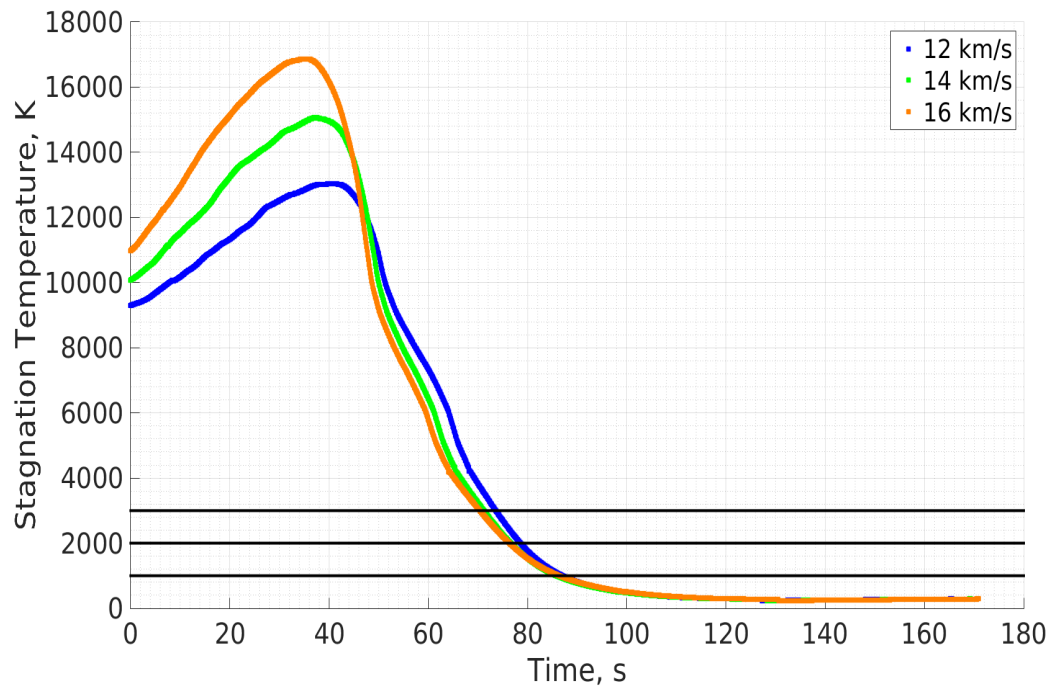


Figure 4.69: Stagnation temperature-time profile for 50 cm body entering at -8 degrees.

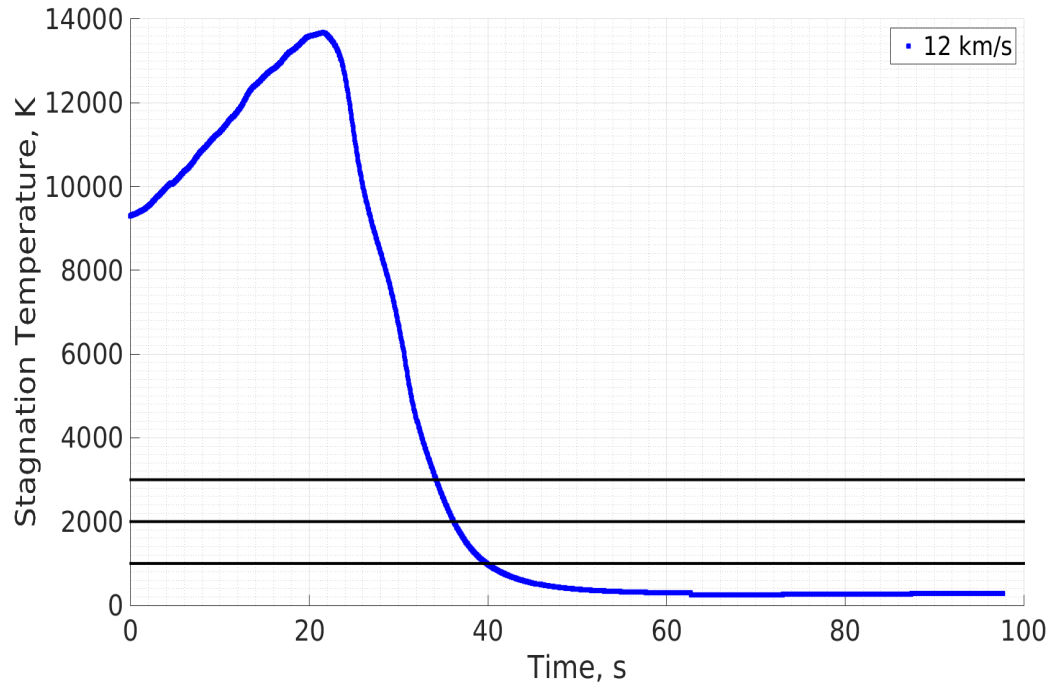


Figure 4.70: Stagnation temperature-time profile for 50 cm body entering at -15 degrees.

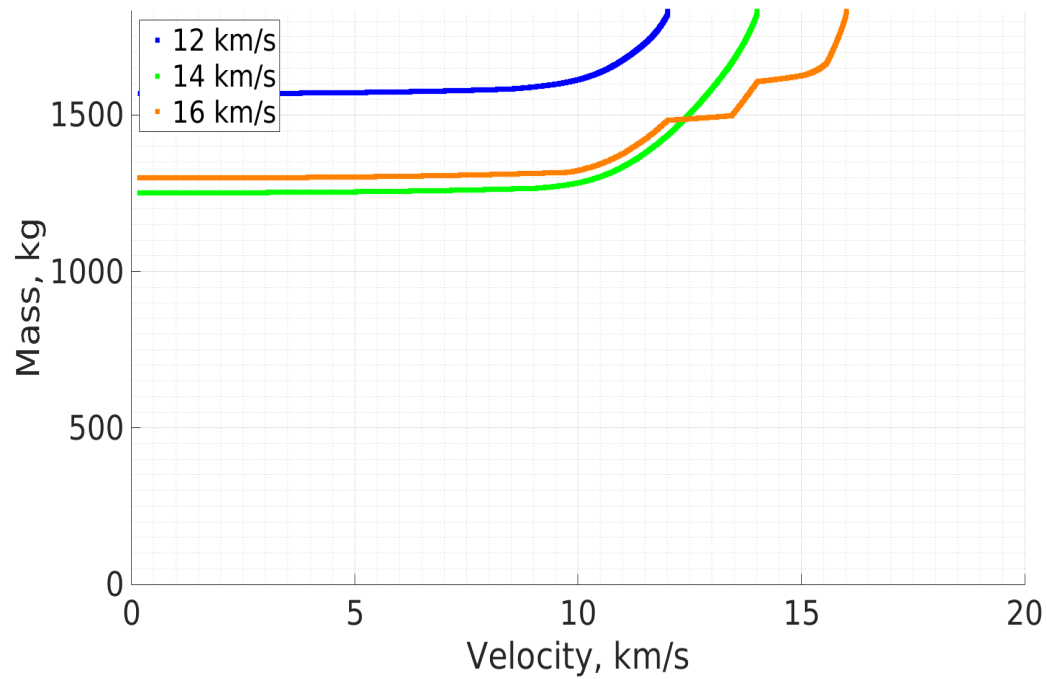


Figure 4.71: Mass-velocity profile for 50 cm body entering at -8 degrees.

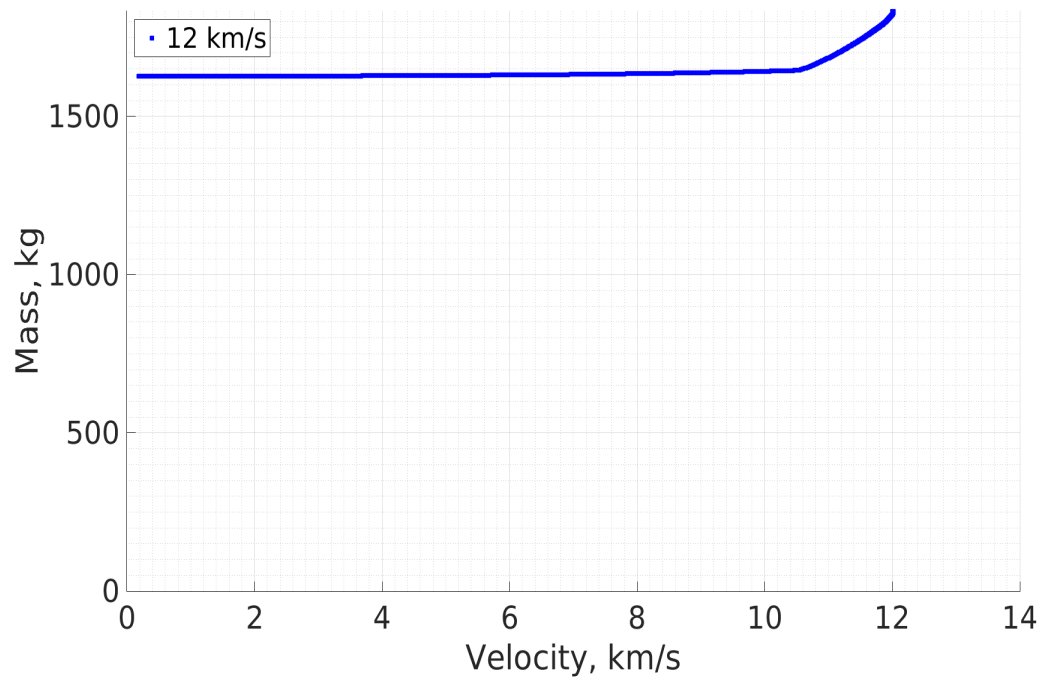


Figure 4.72: Mass-velocity profile for 50 cm body entering at -15 degrees.

4.5 1 m Nose Radius

The 1m nose radius body is the last to be discussed. The only trajectory the model produced that contains useful information is for an entry angle of -8 degrees and an entry velocity of $12 \frac{km}{s}$ (Fig. 4.74). When looking at the rest of the heating predictions, Fig. 4.75 through Fig. 4.89, the trajectories have too many gaps in the radiative heat calculations or result in an airburst. Looking at entry angles between -8 and -15 degrees should reveal other possible candidates but they were not considered for this study. Looking at the other trajectory characteristics, altitude vs. velocity (Fig. 4.90), altitude vs. stagnation temperature (Fig. 4.91), stagnation temperature vs. time (Fig. 4.92), and mass vs. velocity (Fig. 4.93) reveal the same trends as the body size has increased. The kinetic energy is preserved deeper into the atmosphere, the body is subject to higher heat loads, and more mass is being ablated.

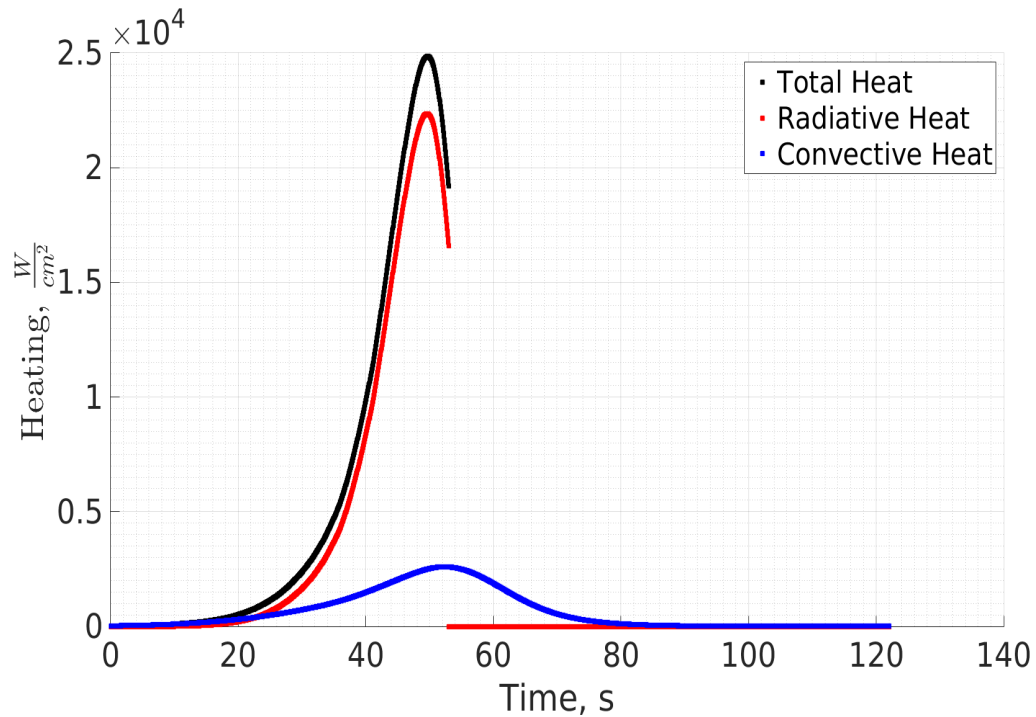


Figure 4.73: Heating-time profile for 1 m body entering at -8 degrees and 12 km/s.

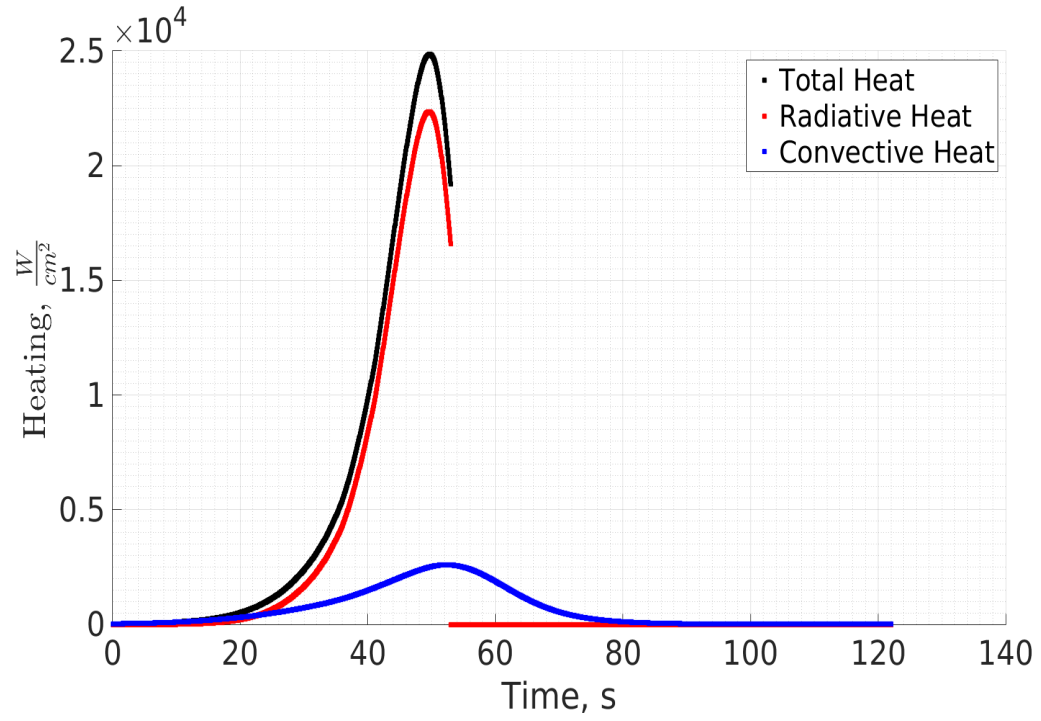


Figure 4.74: Heating-time profile for 1 m body entering at -8 degrees and 12 km/s.

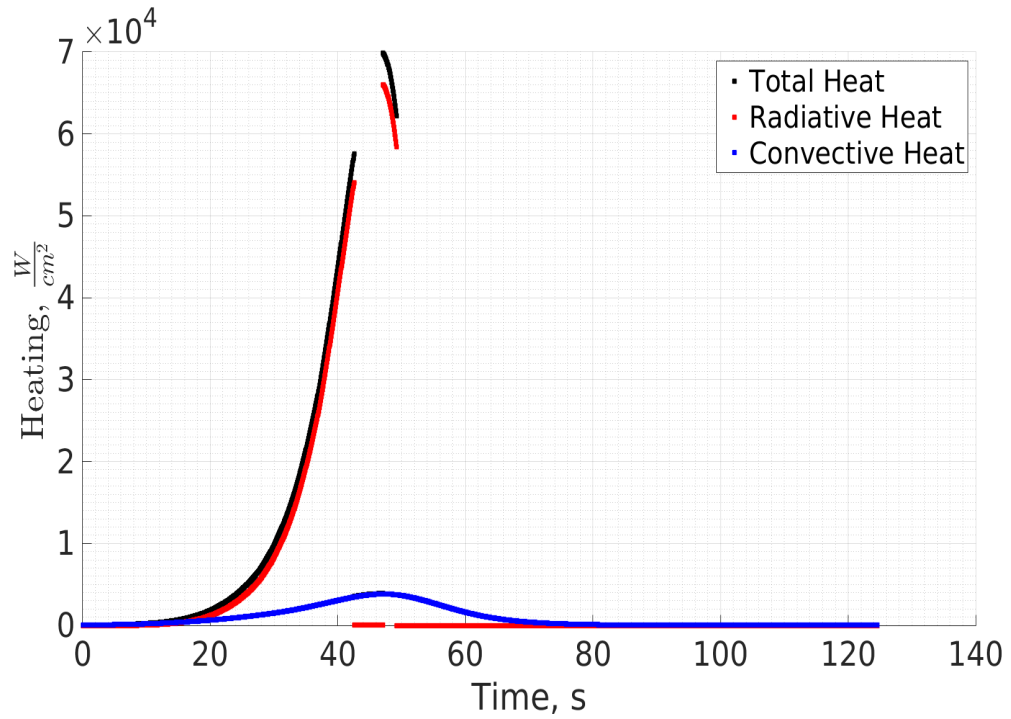


Figure 4.75: Heating-time profile for 1 m body entering at -8 degrees and 14 km/s.

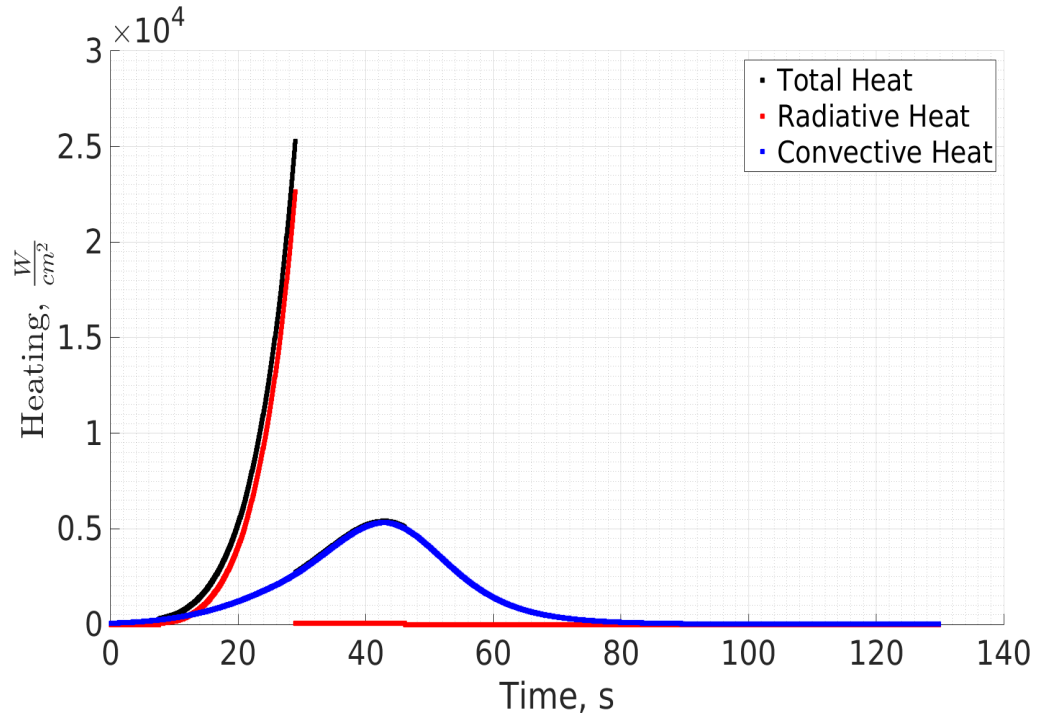


Figure 4.76: Heating-time profile for 1 m body entering at -8 degrees and 16 km/s.

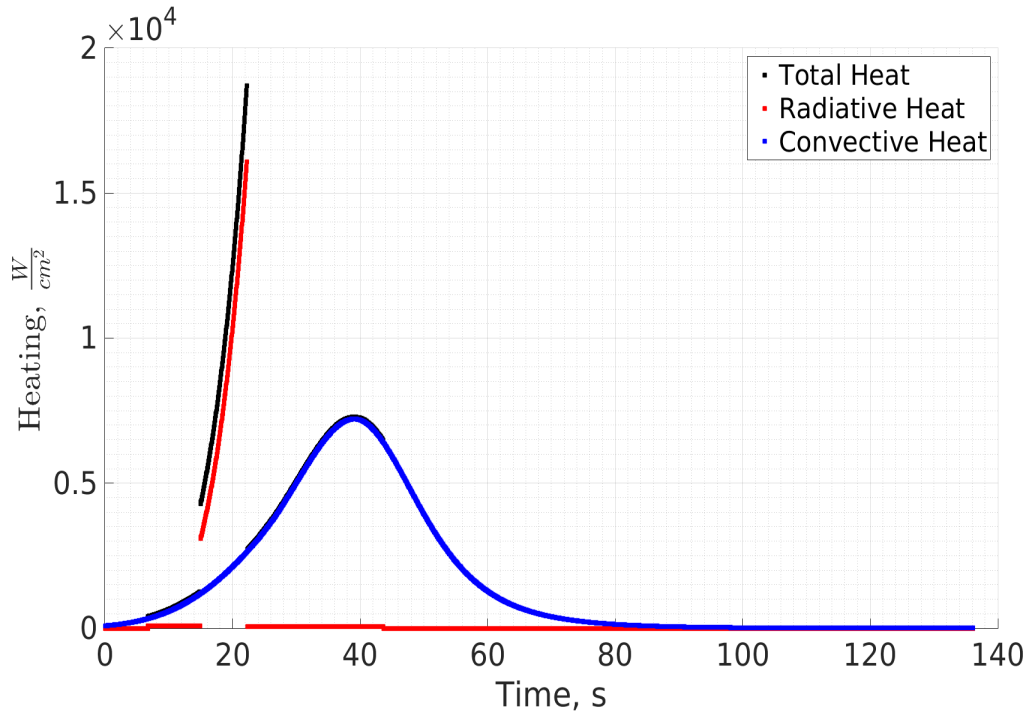


Figure 4.77: Heating-time profile for 1 m body entering at -8 degrees and 18 km/s.

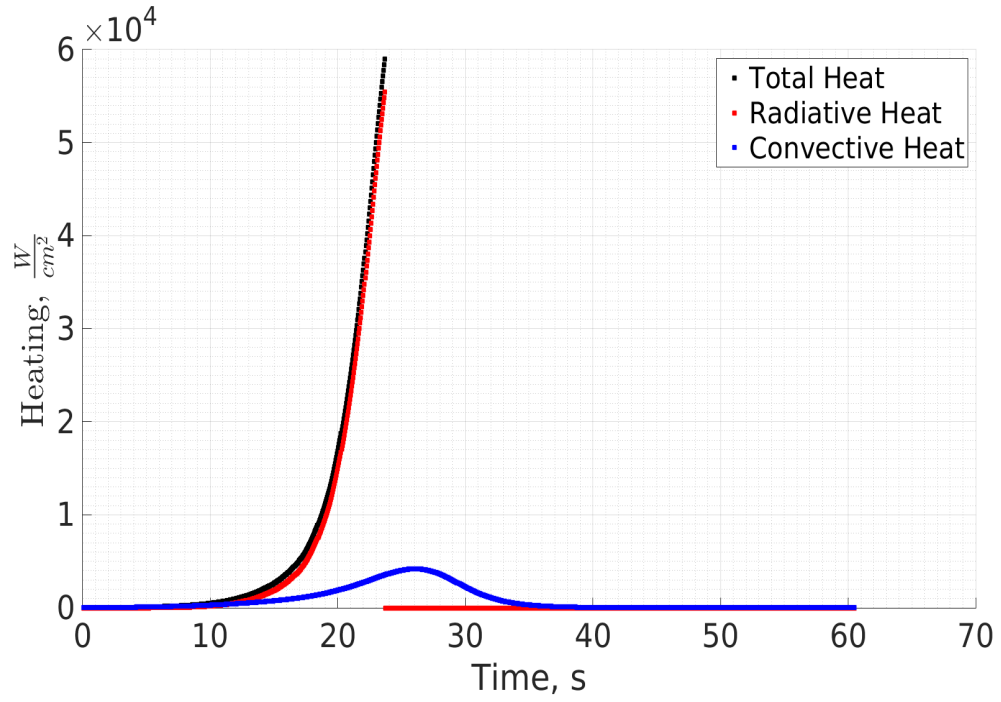


Figure 4.78: Heating-time profile for 1 m body entering at -15 degrees and 12 km/s.

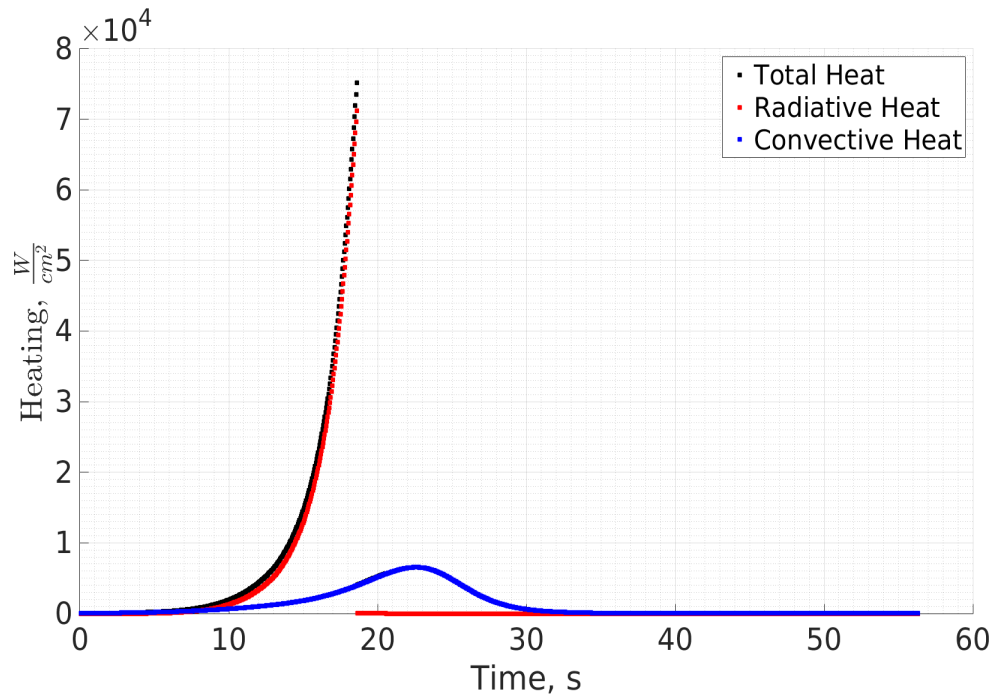


Figure 4.79: Heating-time profile for 1 m body entering at -15 degrees and 14 km/s.

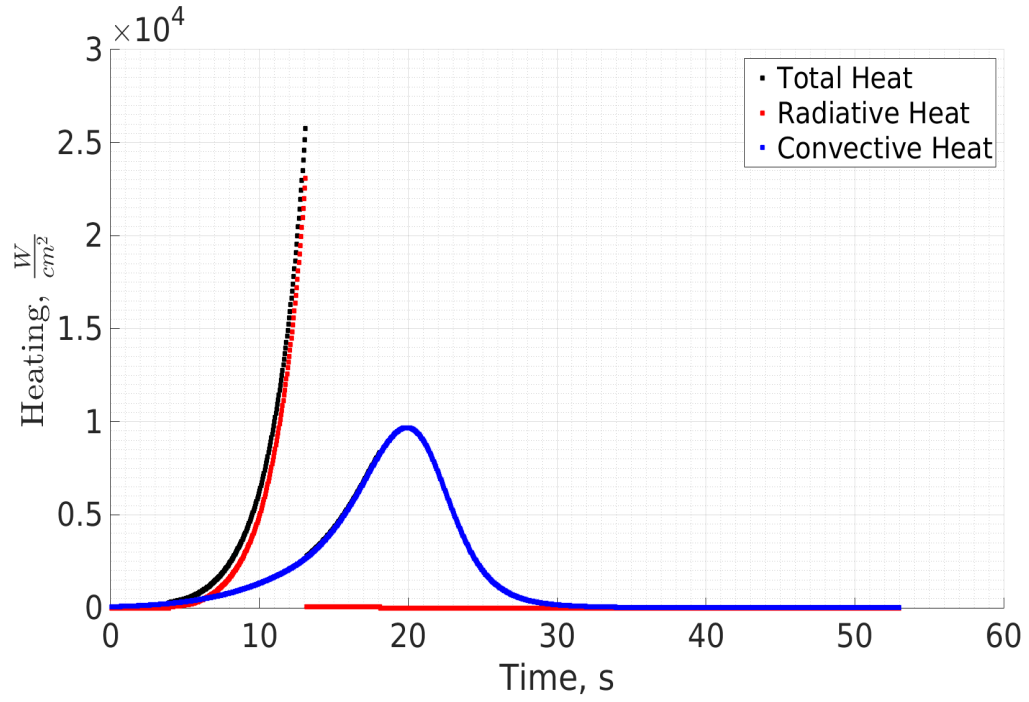


Figure 4.80: Heating-time profile for 1 m body entering at -15 degrees and 16 km/s.

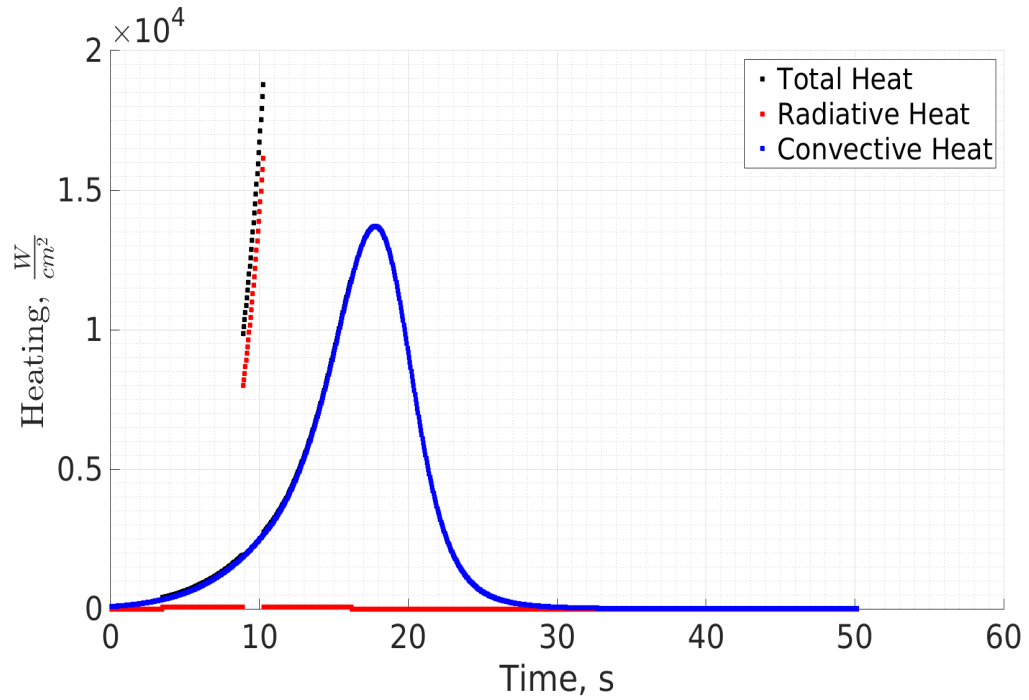


Figure 4.81: Heating-time profile for 1 m body entering at -15 degrees and 18 km/s.

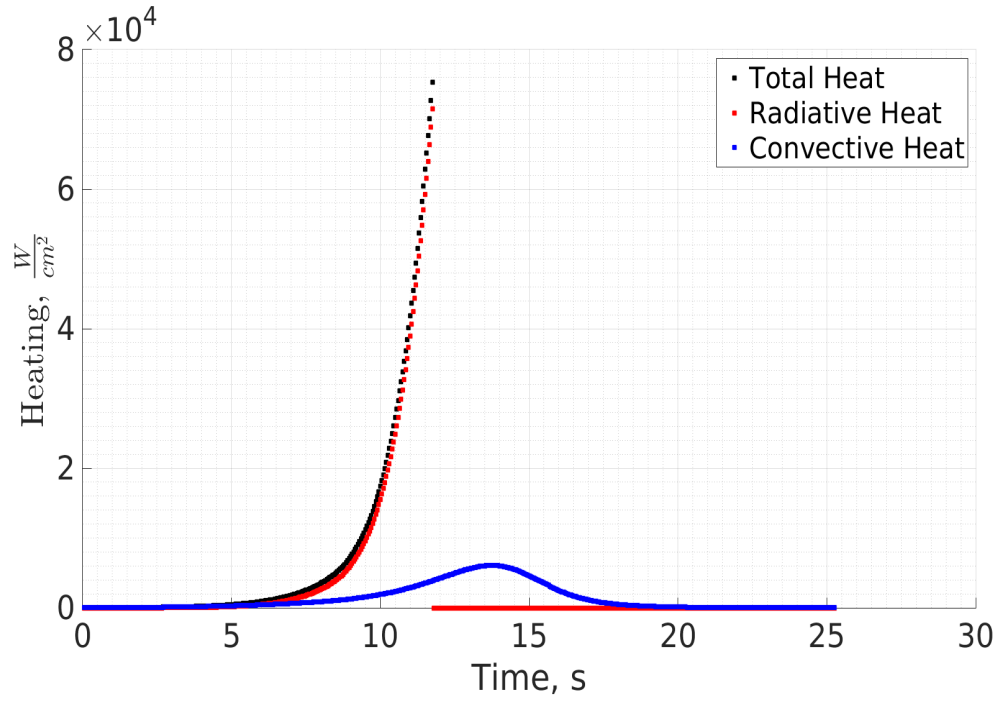


Figure 4.82: Heating-time profile for 1 m body entering at -30 degrees and 12 km/s.

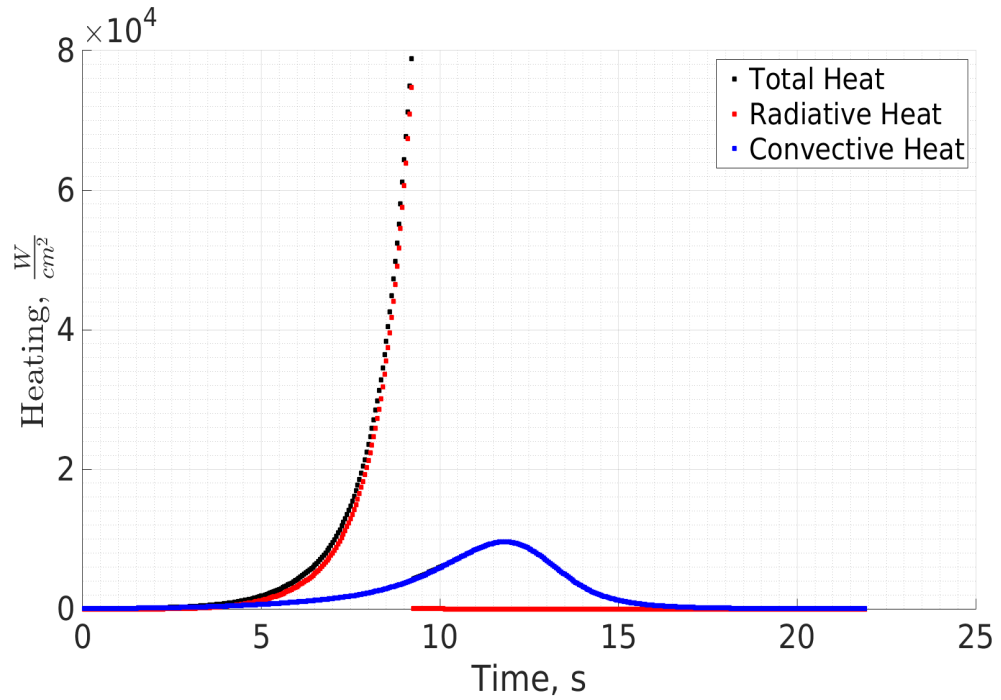


Figure 4.83: Heating-time profile for 1 m body entering at -30 degrees and 14 km/s.

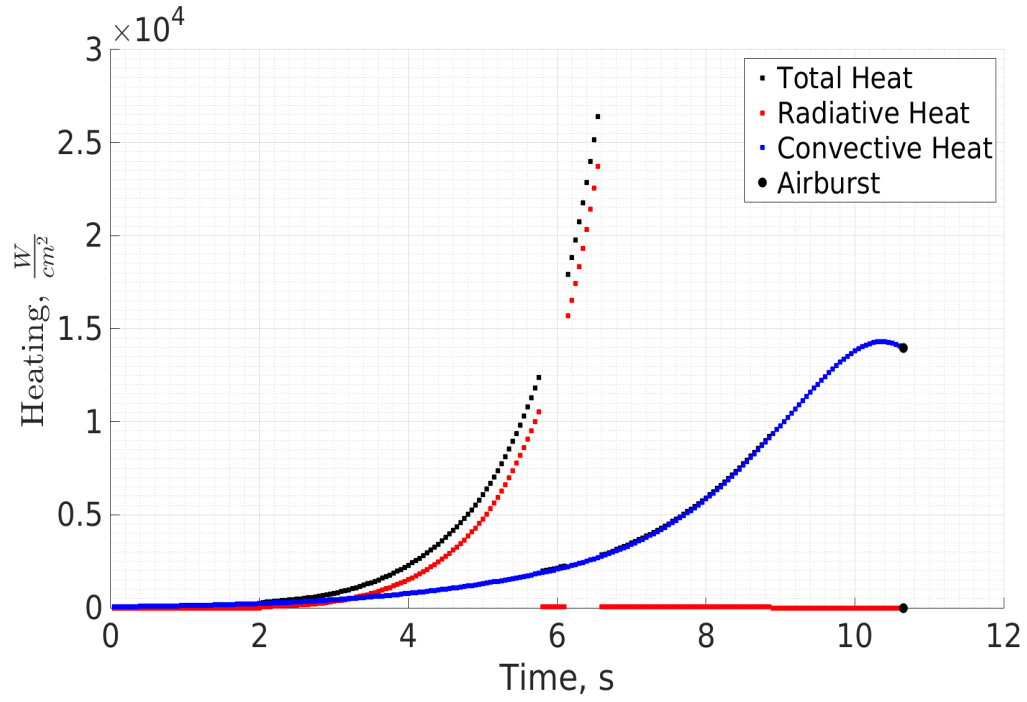


Figure 4.84: Heating-time profile for 1 m body entering at -30 degrees and 16 km/s.

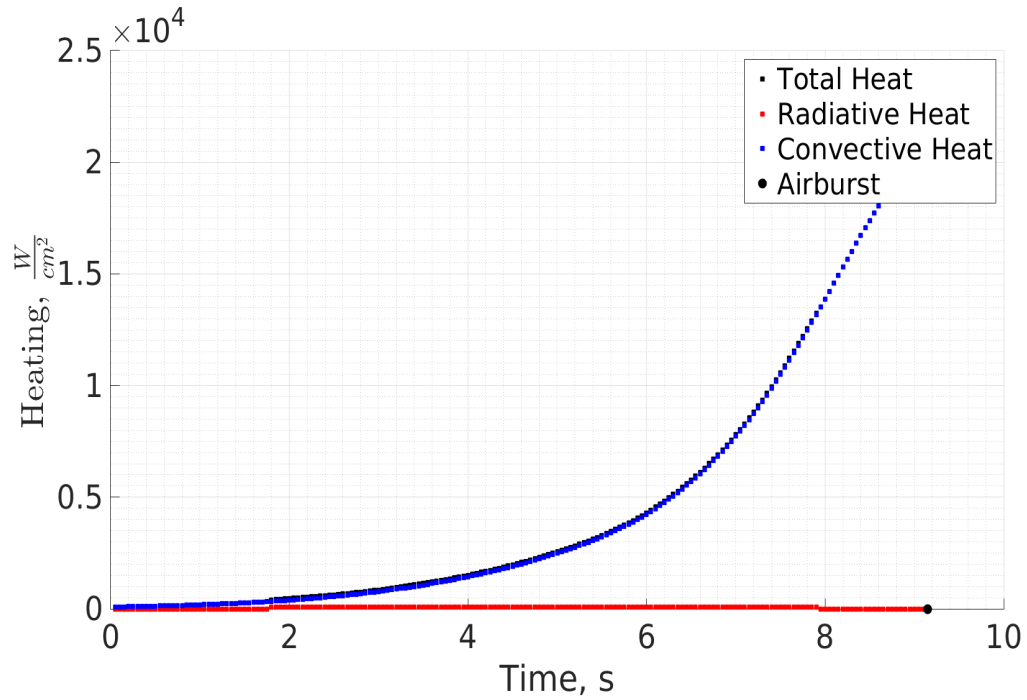


Figure 4.85: Heating-time profile for 1 m body entering at -30 degrees and 18 km/s.

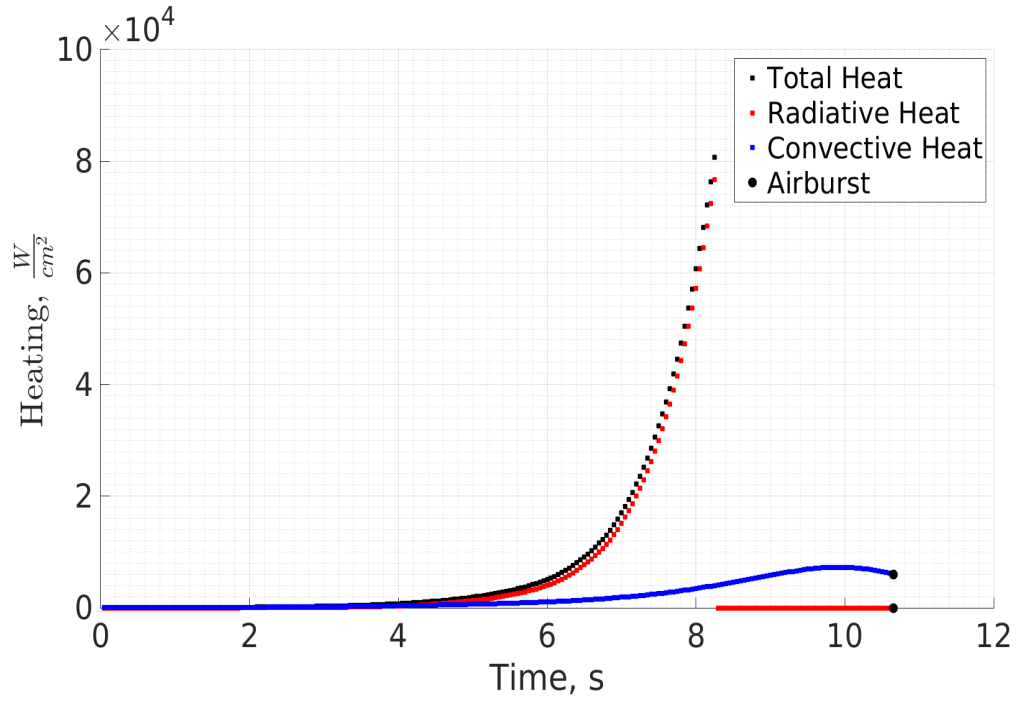


Figure 4.86: Heating-time profile for 1 m body entering at -45 degrees and 12 km/s.

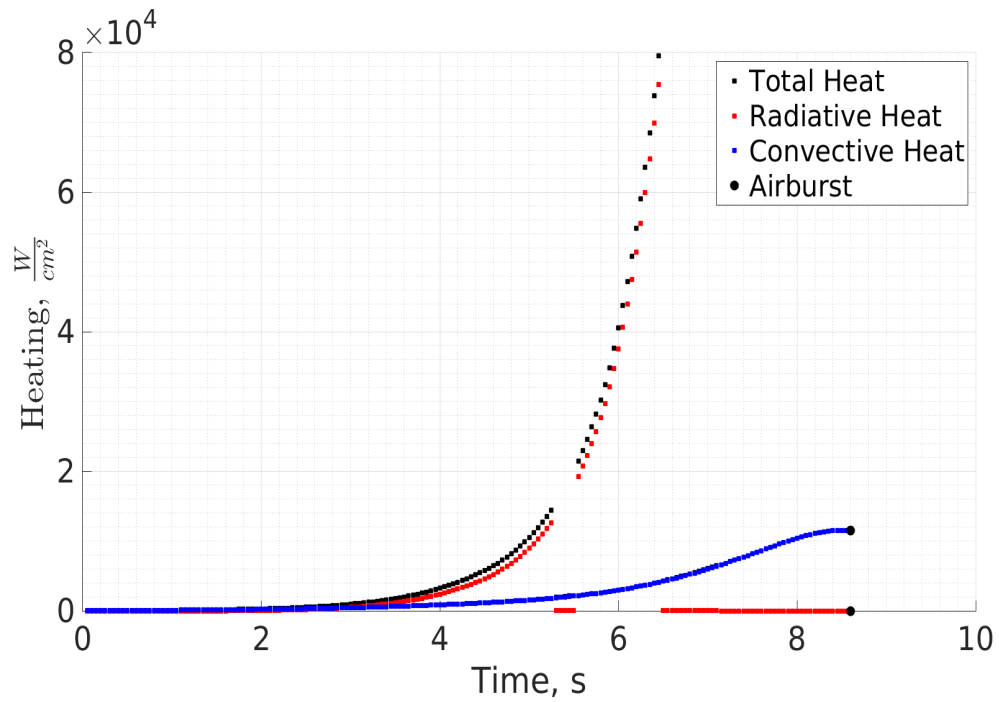


Figure 4.87: Heating-time profile for 1 m body entering at -45 degrees and 14 km/s.

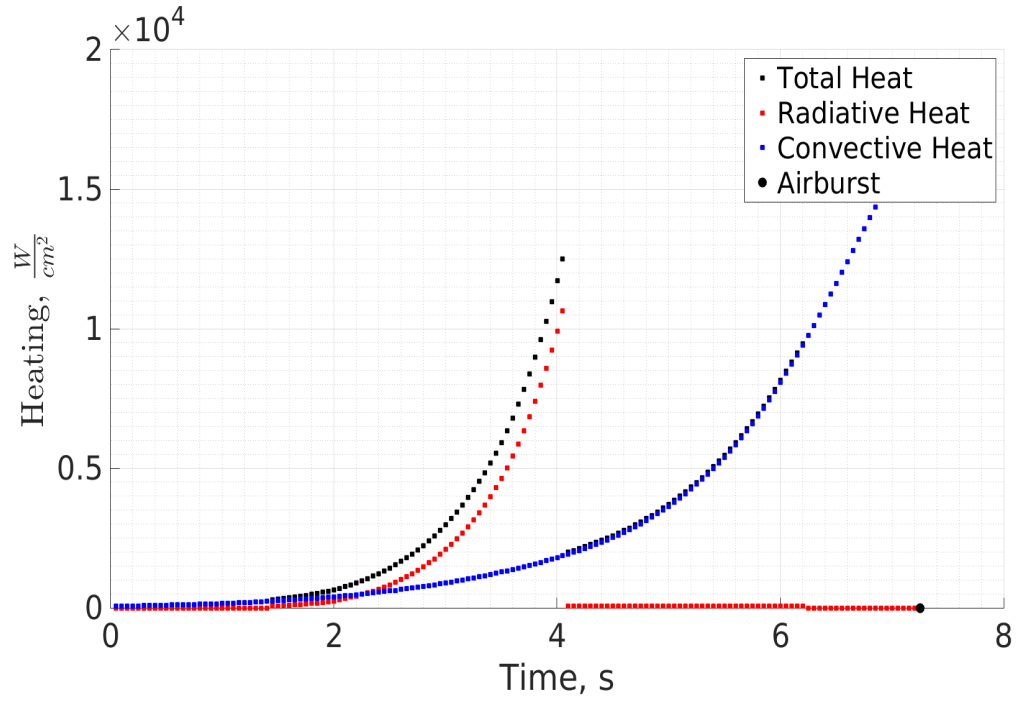


Figure 4.88: Heating-time profile for 1 m body entering at -45 degrees and 16 km/s.

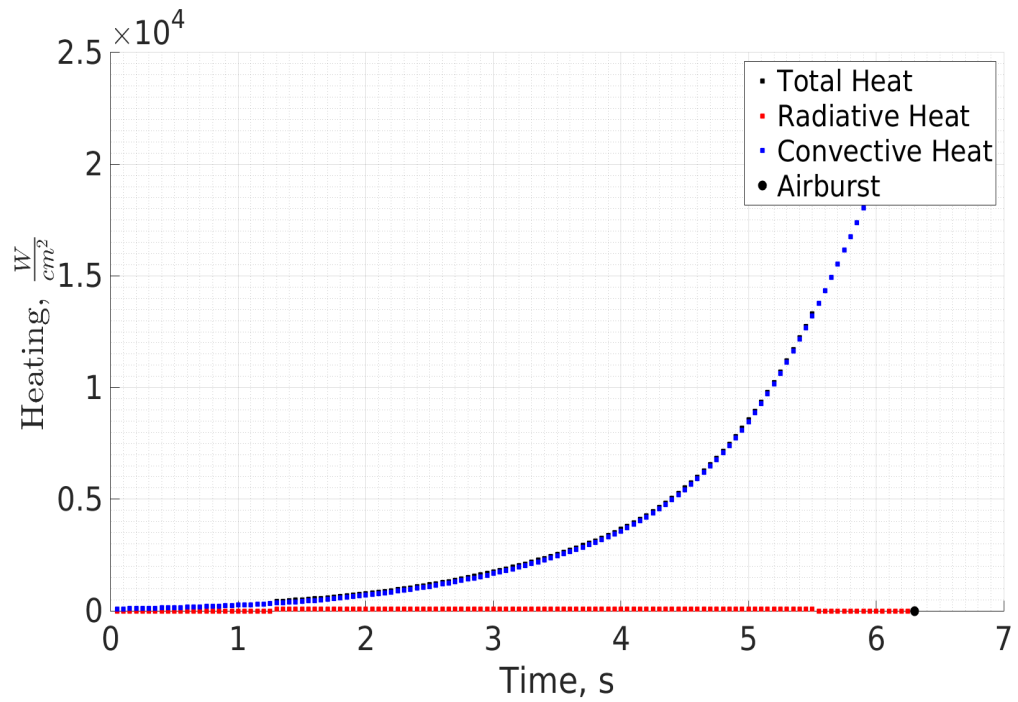


Figure 4.89: Heating-time profile for 1 m body entering at -45 degrees and 18 km/s.

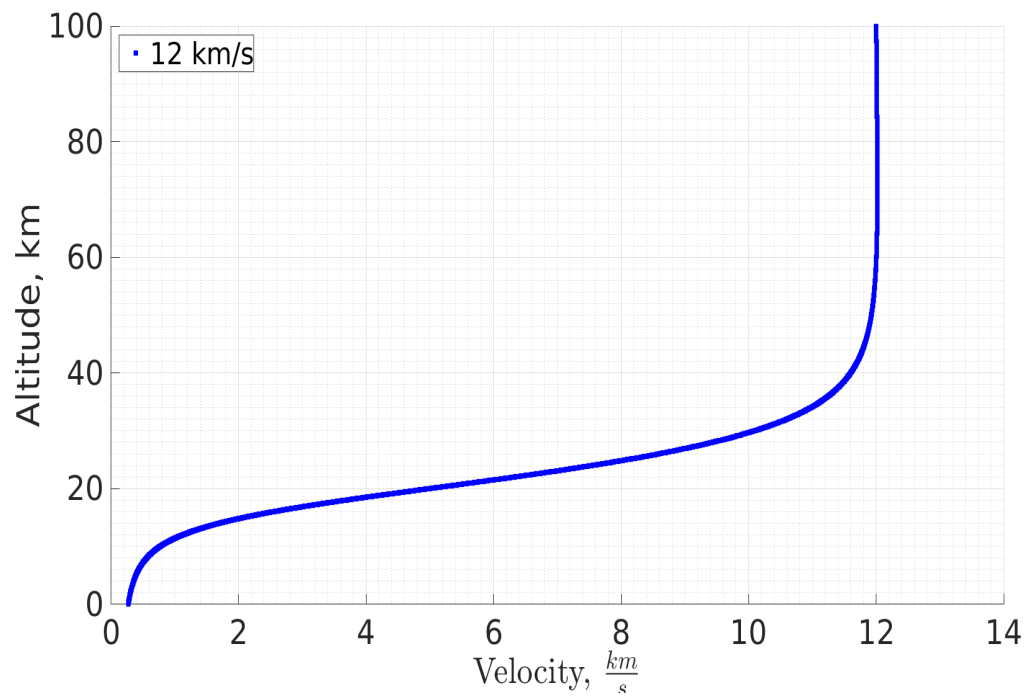


Figure 4.90: Altitude-velocity profile for 1 m body entering at -8 degrees.

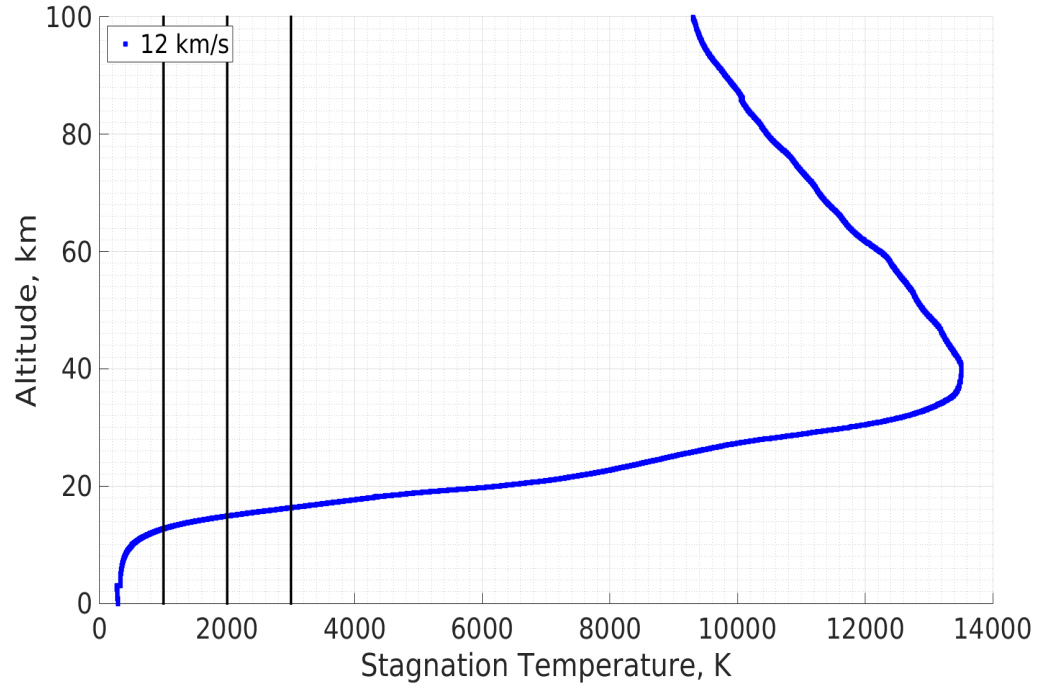


Figure 4.91: Altitude-stagnation temperature profile for 1 m body entering at -8 degrees.

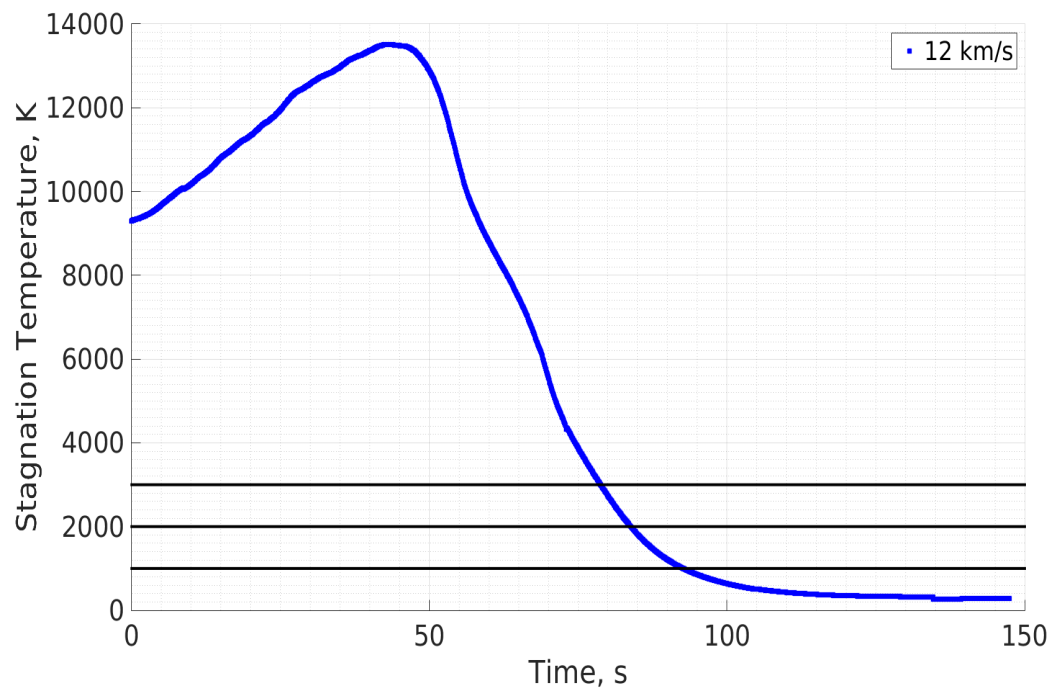


Figure 4.92: Stagnation temperature-time profile for 1 m body entering at -8 degrees.

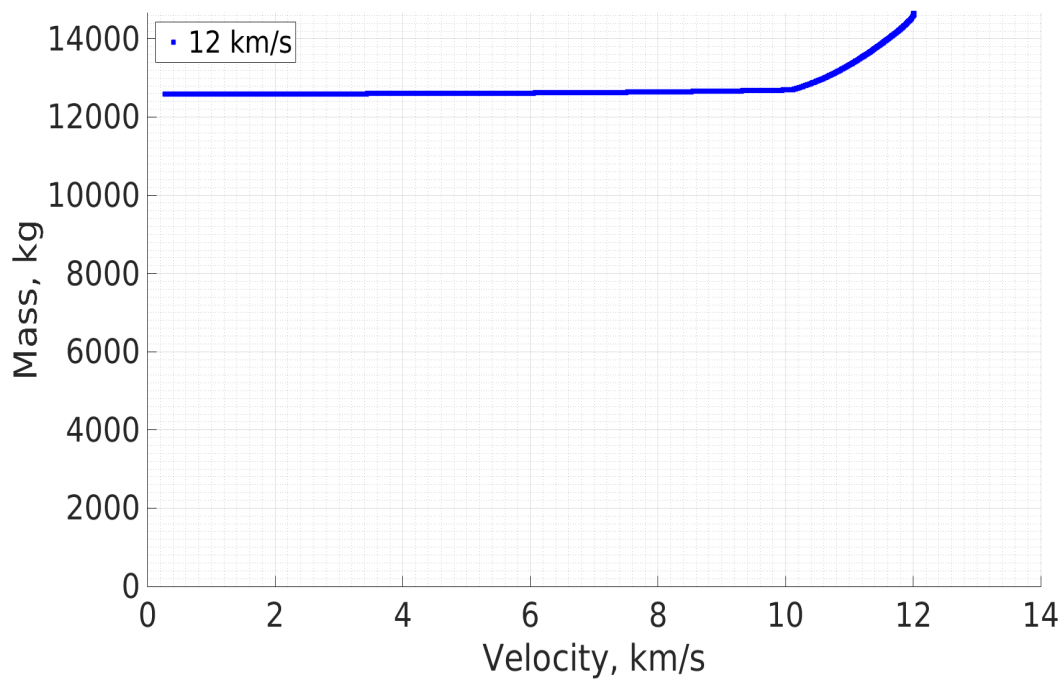


Figure 4.93: Mass-velocity profile for 1 m body entering at -8 degrees.

Chapter 5

Conclusions

5.1 Conclusions

The model previously developed by Fought (1995) and Thames (1997) was a great tool developed for modeling the aerothermodynamics of meteor entry. Their model worked well for large bodies but was not setup to correctly account for entry of small meteors. The premature use of the normal shock relations for an ideal gas caused discontinuities in the density ratio across the shockwave and resulted in bad data. The code adaptations made during the current study have extended the range for which the model is applicable. The model can now predict the heat load for small meteors, on the order of 10cm to 1m, entering at a range of velocities and angles. The range of entry parameters the model can accurately calculate can be seen in Table 5.1. This model was primarily used to look at when high-speed convective cooling becomes an important phenomenon. For bodies with nose radii of 10cm to 50cm there are extended portions of the trajectory where the shock layer temperature is below that of meteoric vaporization. These trajectories aid in the plausibility high-speed convective cooling.

Table 5.1: Model range of applicability.

Valid range of entry parameters for model calculations		
Nose Radius, m	Entry Angle, degrees	Velocity Range, $\frac{km}{s}$
0.1	8	12 - 16
0.1	15	12 - 16
0.1	30	12 - 14
0.1	45	12
0.5	8	12 - 16
0.5	15	12
1	8	12

5.2 Future Work

Although these additions to the model were an improvement for the range of applicability of the model, there are numerous areas that can still be refined. One improvement that could be made is an algorithm that calculates trajectories while making small incremental adjustments to the entry parameters. A more precise limit to the velocity range of the model could be found by holding the nose radius and entry angle constant while increasing the entry velocity by small amounts. The algorithm would increase the velocity until the trajectory was out of the range for the `qfit` subroutine. Once this upper velocity limit was found, the nose radius could be increased by a small increment. With this new nose radius a new upper limit on the velocity could be found. This process could be done for the entry angle and any other characteristics of the entry body, such as entry altitude, body density, and drag coefficient. Also, another algorithm could be developed to bridge the gaps in radiative heating for trajectories that pass in and out of the range of the current `qfit` subroutine. Examples of good candidates for this type of algorithm would be Fig. 4.20, 4.23, 4.51, and 4.53. These trajectories appear to catch the peak radiative heating and have gaps on the sides of the profile. Doing this would effectively extrapolate data from the Sutton and Hartung (1990) tables and extend their range of applicability.

Other notable improvements would be to couple this model with a finite element analysis model. The FEA model could be run at each time step in the trajectory to more accurately represent the physics going on inside the body. This would give a detailed heat flux approximation to aid in determining if high-speed convective cooling is happening. Lastly, a better ablation model could be implemented. A better ablation model will aid in the fidelity of the model by more accurately calculating the flight characteristics at each time step.

Bibliography

- Agrawal, P., Sepka, S. A., Aliaga, J. F., Venkatapathy, E., and Samareh, J. A. (2012). Thermal Soak Analysis of Earth Entry Vehicles. In *43rd AIAA Thermophysics Conference*. 4, 5
- Akima, H. (1974). Algorithm 474: Bivariate Interpolation and Smooth Surface Fitting Based on Local Procedures [E2]. *Communications of the ACM*, 17(1). 14
- Anderson, J. (2000). *Hypersonic and High Temperature Gas Dynamics*. McGraw-Hill series in aeronautical and aerospace engineering. American Institute of Aeronautics and Astronautics. 4
- Bauer, E. (1990). *Physics of High-Temperature Air Part I Basics*. Institute for Defense Analyses. 4
- Bergman, T., Incropera, F., and Lavine, A. (2011). *Fundamentals of Heat and Mass Transfer*. Wiley. 3
- Brauer, G. L., Comick, D. E., Olson, D. W., Peterson, F. W., and Stevenson, R. (1989). Program to Optimize Simulated Trajectories (POST). 25
- Chyba, C. F. (1991). Terrestrial Mantle Siderophiles and the Lunar Impact Record. *Icarus*, 92:217–233. 14
- Chyba, C. F., Thomas, P. J., Brookshaw, L., and Sagan, C. (1990). Cometary Delivery of Organic Molecules to the Early Earth. *Science*. 5
- Chyba, C. F., Thomas, P. J., and Zahnle, K. J. (1993). The 1908 Tunguska Explosion: Atmospheric Disruption of a Stony Asteroid. *Nature*, 361(6407):40–44. viii, 6, 7, 9, 11, 12, 13, 17

- Fought, R. (1995). A Computer Model and Parametric Study of Earth-Atmospheric Entry of Meteors. Master's thesis, The University of Tennessee, Knoxville. 4, 8, 9, 10, 11, 12, 13, 16, 17, 18, 19, 26, 88
- Fought, R., Lyne, J. E., and Tauber, M. (1996). An analytical model of the atmospheric entry of large meteors and its application to the Tunguska Event. *Journal of Geophysical Research: Planets*, 101(E10):23207–23212. 9, 12
- Fought, R. M., Gupta, R. N., and Lyne, J. E. (1997). Extension of curve fits for enthalpy of equilibrium air to 1000 atmospheres. *AIAA Journal*, 11(2):126–128. 11
- Goulard, R. (1964). Preliminary Estimates of Radiative Transfer Effects on Detached Shock Layers. *AIAA Journal*, 2(2):494–502. 11
- Green, M. J. and Nicolet, W. E. (1978). Aerothermodynamic Environment for Jovian Entry with Silica Heat Shield. *AIAA*. 14, 35
- Greenwood, T. F. and Lee, Y. C. (1983). Space Shuttle Base Heating. *AIAA 18th Thermophysics Conference*. x, 2
- Gupta, R., Lee, K., Thompson, R. A., and Yos, J. M. (1991). Calculations and Curve Fits of Thermodynamic and Transport Properties for Equilibrium Air to 30000 K. 11
- Gupta, R. N., Lee, K. P., Moss, J. N., and Sutton, K. (1994). Viscous Shock-Layer Solutions With Coupled Radiation and Ablation for Earth Entry. *Journal of Spacecraft and Rockets*, 29(2). 12
- Hochstim, A. R. (1963). Hypersonic Chemosynthesis and Possible Formation of Organic Compounds From Impact of Meteorites on Water. 18
- Howell, J., Menguc, M., and Siegel, R. (2015). *Thermal Radiation Heat Transfer*, 6th Edition. CRC Press. 3

- Huber, P. W. (1963). Hypersonic Shock-Heated Flow Parameters for Velocities to 46000 Feet per Second and Altitudes to 323000 Feet. x, 18, 19, 20, 22
- Ip, W. H. (1990). Meteoroid Ablation Processes in Titan's Atmosphere. *Nature*, 345. 7
- Kopal, Z. (1971). *Physics and Astronomy of the Moon*. Academic Press. 16
- Lyne, J. E. (1992). *Physiologically Constrained Aerocapture for Manned Mars Missions*. PhD thesis, North Carolina State, Raleigh, North Carolina. 34
- Lyne, J. E. and Tauber, M. (1994). An Analysis of the Tunguska Event. Report to NASA under the 1994 Summer Faculty Fellowship Program. 12
- Lyne, J. E. and Tauber, M. E. (1995). Origin of the Tunguska Event. *Nature*, 375:638–639. 9
- Lyne, J. E., Tauber, M. E., and Fought, R. M. (1998). A computer model of the atmospheric entry of the Tunguska object. *Planetary and Space Sciences*, 46:245–252. 4
- Marvin, J. G. and Deiwert, G. S. (1965). Convective Heat Transfer in Planetary Gases. *NASA TR R-224*. 34
- McAuliffe, J. P. and Apostolos, C. A. (2005). Modelling Meteor Ablation in the Venusian Atmosphere. *Icarus*, 180:8–22. 7
- Moses, J. I. (1992). Meteoroid Ablation in Neptune's Atmosphere. *Icarus*, 99:368–383. 7
- Opik, E. (1958). *Physics of Meteor Flight in the Atmosphere*. Interscience tracts on physics and astronomy. Interscience Publishers. 37
- Pai, S. (2012). *Radiation Gas Dynamics*. Springer Vienna. 26
- Sewell, K. G. (1961). Thermodynamic Properties of High Temperature Air. 13, 14

- Sutton, K. (1973). Characteristics of Couple Nongray Radiating Gas Flows with Ablation Products Effects About Blunt Bodies During Planetary Entries. Master's thesis, North Carolina State, Raleigh, North Carolina. 26
- Sutton, K. and Hartung, L. C. (1990). Equilibrium Heating Tables for Earth Entry. *NASA TM 102652*. viii, ix, 16, 26, 32, 33, 36, 37, 89, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106
- Tauber, M. E. (1969). Atmospheric entry to jupiter. *Journal of Spacecraft and Rockets*, 6(10):1103–1109. 11
- Tauber, M. E. (AA-213). Atmospheric Trajectories. 3
- Tauber, M. E. and Sutton, K. (1991). Stagnation-Point Radiative Heating Relations for Earth and Mars Entries. *Journal of Spacecraft and Rockets*, 28:40–42. 33
- Thames, C. (1997). A Computational Model of Meteor Entry Into Earth's Atmosphere. Master's thesis, The University of Tennessee. 13, 14, 15, 16, 17, 18, 19, 26, 88
- Thames, C. and Lyne, J. E. (1997). A Computational Model of the Atmospheric Entry of Meteors at Speeds up to 50 km/sec. In *22nd Atmospheric Flight Mechanics Conference*. 4

Appendix

A Radiative Tables

Table A.1: Radiative Heating Prediction at 30 km Altitude (Sutton and Hartung, 1990)

Predicted Radiative Heating, Q_r in $\frac{MW}{m^2}$

$\begin{array}{c} \text{Rn} \\ \text{V} \end{array}$.05 m	.1 m	.23 m	.5 m
8.0 km/s	1.332e+00	2.410e+00	4.884e+00	9.482e+00
8.5 km/s	2.208e+00	3.815e+00	7.609e+00	1.460e+01
9.0 km/s	4.006e+00	6.676e+00	1.283e+01	2.389e+01
9.5 km/s	9.316e+00	1.442e+01	2.578e+01	4.517e+01
10.0 km/s	2.668e+01	3.868e+01	6.305e+01	1.011e+02
11.0 km/s	1.269e+02	1.703e+02	2.468e+02	3.460e+02
12.0 km/s	3.097e+02	3.961e+02	5.421e+02	7.171e+02
$\begin{array}{c} \text{Rn} \\ \text{V} \end{array}$	1 m	2.3 m	5 m	10 m
8.0 km/s	1.664e+01	3.074e+01	5.155e+01	7.685e+01
8.5 km/s	2.531e+01	4.682e+01	7.700e+01	1.123e+02
9.0 km/s	4.093e+01	7.386e+01	1.184e+02	1.649e+02
9.5 km/s	7.376e+01	1.259e+02	1.911e+02	2.514e+02
10.0 km/s	1.508e+02	2.325e+02	3.201e+02	3.861e+02
11.0 km/s	4.550e+02	5.962e+02	7.013e+02	7.709e+02
12.0 km/s	8.923e+02	1.077e+03	1.215e+03	1.171e+03

Table A.2: Radiative Heating Prediction at 36 km Altitude (Sutton and Hartung, 1990)

Predicted Radiative Heating, Q_r in $\frac{MW}{m^2}$				
$\begin{array}{c} \text{Rn} \\ \hline \text{V} \end{array}$.05 m	.1 m	.23 m	.5 m
8.0 km/s	2.493e-01	4.548e-01	9.506e-01	1.871e+00
8.5 km/s	4.345e-01	7.658e-01	1.517e+00	2.931e+00
9.0 km/s	9.116e-01	1.476e+00	2.711e+00	5.032e+00
9.5 km/s	2.659e+00	3.963e+00	6.640e+00	1.115e+01
10.0 km/s	9.384e+00	1.296e+01	1.963e+01	3.010e+01
11.0 km/s	4.716e+01	6.065e+01	8.480e+01	1.188e+02
12.0 km/s	1.178e+02	1.459e+02	1.951e+02	2.611e+02
14.0 km/s	3.569e+02	4.168e+02	5.252e+02	6.666e+02
$\begin{array}{c} \text{Rn} \\ \hline \text{V} \end{array}$	1 m	2.3 m	5 m	10 m
8.0 km/s	3.482e+00	7.120e+00	1.317e+01	2.194e+01
8.5 km/s	5.358e+00	1.084e+01	1.994e+01	3.286e+01
9.0 km/s	8.889e+00	1.749e+01	3.151e+01	5.016e+01
9.5 km/s	1.824e+01	3.312e+01	5.572e+01	8.326e+01
10.0 km/s	4.492e+01	7.213e+01	1.082e+02	1.470e+02
11.0 km/s	1.604e+02	2.232e+02	2.887e+02	3.574e+02
12.0 km/s	3.361e+02	4.368e+02	5.264e+02	6.196e+02
14.0 km/s	8.209e+02	1.001e+03	1.189e+03	1.234e+03

Table A.3: Radiative Heating Prediction at 42 km Altitude (Sutton and Hartung, 1990)

Predicted Radiative Heating, Q_r in $\frac{MW}{m^2}$				
$\begin{array}{c} \text{Rn} \\ \hline \text{V} \end{array}$.05 m	.1 m	.23 m	.5 m
8.0 km/s	5.083e-02	9.289e-02	1.946e-01	3.918e-01
8.5 km/s	9.308e-02	1.633e-01	3.268e-01	6.296e-01
9.0 km/s	2.124e-01	3.515e-01	6.467e-01	1.153e+00
9.5 km/s	8.145e-01	1.231e+00	2.002e+00	3.212e+00
10.0 km/s	3.336e+00	4.720e+00	7.026e+00	1.018e+01
11.0 km/s	1.767e+01	2.315e+01	3.135e+01	4.201e+01
12.0 km/s	4.535e+01	5.667e+01	7.286e+01	9.429e+01
14.0 km/s	1.437e+02	1.689e+02	2.044e+02	2.518e+02
$\begin{array}{c} \text{Rn} \\ \hline \text{V} \end{array}$	1 m	2.3 m	5 m	10 m
8.0 km/s	7.251e-01	1.550e+00	3.090e+00	5.603e+00
8.5 km/s	1.136e+00	2.380e+00	4.701e+00	8.361e+00
9.0 km/s	2.008e+00	4.008e+00	7.657e+00	1.313e+01
9.5 km/s	5.012e+00	8.851e+00	1.528e+01	2.415e+01
10.0 km/s	1.446e+01	2.266e+01	3.471e+01	4.914e+01
11.0 km/s	5.557e+01	7.849e+01	1.056e+02	1.316e+02
12.0 km/s	1.208e+02	1.618e+02	2.047e+02	2.553e+02
14.0 km/s	3.090e+02	4.105e+02	4.945e+02	5.560e+02

Table A.4: Radiative Heating Prediction at 48 km Altitude (Sutton and Hartung, 1990)

Predicted Radiative Heating, Q_r in $\frac{MW}{m^2}$				
$\begin{array}{c} \text{Rn} \\ \text{V} \end{array}$.05 m	.1 m	.23 m	.5 m
9.0 km/s	5.477e-02	9.356e-02	1.741e-01	3.077e-01
9.5 km/s	2.660e-01	4.080e-01	6.803e-01	1.084e+00
10.0 km/s	1.174e+00	1.734e+00	2.674e+00	3.882e+00
11.0 km/s	6.404e+00	8.824e+00	1.239e+01	1.650e+01
12.0 km/s	1.684e+01	2.227e+01	2.958e+01	3.745e+01
14.0 km/s	5.597e+01	7.013e+01	8.606e+01	1.025e+02
16.0 km/s	1.229e+02	1.486e+02	1.726e+02	1.960e+02
$\begin{array}{c} \text{Rn} \\ \text{V} \end{array}$	1 m	2.3 m	5 m	10 m
9.0 km/s	5.193e-01	1.012e+00	1.936e+00	3.435e+00
9.5 km/s	1.644e+00	2.758e+00	4.588e+00	7.275e+00
10.0 km/s	5.377e+00	8.017e+00	1.185e+01	1.688e+01
11.0 km/s	2.119e+01	2.897e+01	3.905e+01	4.996e+01
12.0 km/s	4.639e+01	6.098e+01	7.856e+01	1.015e+02
14.0 km/s	1.278e+02	1.617e+02	1.992e+02	2.324e+02
16.0 km/s	2.400e+02	2.953e+02	3.522e+02	3.973e+02

Table A.5: Radiative Heating Prediction at 54 km Altitude (Sutton and Hartung, 1990)

Predicted Radiative Heating, Q_r in $\frac{MW}{m^2}$				
$\begin{array}{c} \text{Rn} \\ \hline \text{V} \end{array}$.05 m	.1 m	.23 m	.5 m
9.0 km/s	1.526e-02	2.710e-02	5.316e-02	9.716e-02
9.5 km/s	9.657e-02	1.521e-01	2.582e-01	4.147e-01
10.0 km/s	4.329e-01	6.610e-01	1.064e+00	1.598e+00
11.0 km/s	2.355e+00	3.438e+00	5.090e+00	7.006e+00
12.0 km/s	6.297e+00	8.886e+00	1.250e+01	1.635e+01
14.0 km/s	2.148e+01	2.923e+01	3.839e+01	4.669e+01
16.0 km/s	4.784e+01	6.393e+01	8.004e+01	9.217e+01
18.0 km/s	8.421e+01	1.127e+02	1.370e+02	1.509e+02
$\begin{array}{c} \text{Rn} \\ \hline \text{V} \end{array}$	1 m	2.3 m	5 m	10 m
9.0 km/s	1.632e-01	3.062e-01	5.744e-01	1.021e+00
9.5 km/s	6.268e-01	1.029e+00	1.651e+00	2.549e+00
10.0 km/s	2.239e+00	3.293e+00	4.707e+00	6.547e+00
11.0 km/s	9.059e+00	1.215e+01	1.598e+01	2.045e+01
12.0 km/s	2.024e+01	2.592e+01	3.270e+01	4.253e+01
14.0 km/s	5.447e+01	6.961e+01	8.532e+01	1.015e+02
16.0 km/s	1.083e+02	1.291e+02	1.541e+02	1.774e+02
18.0 km/s	1.646e+02	2.000e+02	2.344e+02	2.575e+02

Table A.6: Radiative Heating Prediction at 60 km Altitude (Sutton and Hartung, 1990)

Predicted Radiative Heating, Q_r in $\frac{MW}{m^2}$				
$\begin{array}{c} \text{Rn} \\ \hline \text{V} \end{array}$.05 m	.1 m	.23 m	.5 m
9.0 km/s	4.066e-03	7.402e-03	1.518e-02	2.926e-02
9.5 km/s	3.462e-02	5.639e-02	9.802e-02	1.618e-01
10.0 km/s	1.539e-01	2.391e-01	4.020e-01	6.277e-01
11.0 km/s	8.078e-01	1.246e+00	1.974e+00	2.860e+00
12.0 km/s	2.174e+00	3.281e+00	4.996e+00	6.872e+00
14.0 km/s	7.540e+00	1.112e+01	1.607e+01	2.065e+01
16.0 km/s	1.688e+01	2.480e+01	3.478e+01	4.258e+01
18.0 km/s	2.924e+01	4.355e+01	6.087e+01	7.203e+01
$\begin{array}{c} \text{Rn} \\ \hline \text{V} \end{array}$	1 m	2.3 m	5 m	10 m
9.0 km/s	5.156e-02	9.884e-02	1.822e-01	3.223e-01
9.5 km/s	2.470e-01	4.021e-01	6.324e-01	9.508e-01
10.0 km/s	9.077e-01	1.360e+00	1.940e+00	2.644e+00
11.0 km/s	3.822e+00	5.196e+00	6.784e+00	8.543e+00
12.0 km/s	8.765e+00	1.130e+01	1.411e+01	1.701e+01
14.0 km/s	2.472e+01	2.969e+01	3.720e+01	4.383e+01
16.0 km/s	4.854e+01	5.510e+01	6.810e+01	7.830e+01
18.0 km/s	7.927e+01	9.170e+01	1.045e+02	1.174e+02

Table A.7: Radiative Heating Prediction at 66 km Altitude (Sutton and Hartung, 1990)

Predicted Radiative Heating, Q_r in $\frac{MW}{m^2}$				
$\begin{array}{c} \text{Rn} \\ \hline \text{V} \end{array}$.05 m	.1 m	.23 m	.5 m
9.0 km/s	1.066e-03	1.954e-03	4.083e-03	8.107e-03
9.5 km/s	1.192e-02	1.980e-02	3.567e-02	6.070e-02
10.0 km/s	5.211e-02	8.288e-02	1.442e-01	2.347e-01
11.0 km/s	2.591e-01	4.116e-01	6.968e-01	1.074e+00
12.0 km/s	6.892e-01	1.089e+00	1.800e+00	2.672e+00
14.0 km/s	2.405e+00	3.773e+00	6.020e+00	8.454e+00
16.0 km/s	5.401e+00	8.473e+00	1.335e+01	1.814e+01
18.0 km/s	9.138e+00	1.465e+01	2.343e+01	3.145e+01
$\begin{array}{c} \text{Rn} \\ \hline \text{V} \end{array}$	1 m	2.3 m	5 m	10 m
9.0 km/s	1.484e-02	3.010e-02	5.611e-02	9.810e-02
9.5 km/s	9.651e-02	1.612e-01	2.503e-01	3.696e-01
10.0 km/s	3.497e-01	5.428e-01	7.805e-01	1.069e+00
11.0 km/s	1.511e+00	2.147e+00	2.845e+00	3.590e+00
12.0 km/s	3.585e+00	4.810e+00	6.056e+00	7.340e+00
14.0 km/s	1.065e+01	1.323e+01	1.560e+01	1.907e+01
16.0 km/s	2.181e+01	2.559e+01	3.050e+01	3.470e+01
18.0 km/s	3.667e+01	4.125e+01	4.760e+01	5.260e+01

Table A.8: Radiative Heating Prediction at 72 km Altitude (Sutton and Hartung, 1990)

Predicted Radiative Heating, Q_r in $\frac{MW}{m^2}$				
$\begin{array}{c} \text{Rn} \\ \hline \text{V} \end{array}$.05 m	.1 m	.23 m	.5 m
9.0 km/s	2.619e-04	4.789e-04	1.004e-03	2.016e-03
9.5 km/s	3.420e-03	5.749e-03	1.076e-02	1.891e-02
10.0 km/s	1.576e-02	2.512e-02	4.497e-02	7.685e-02
11.0 km/s	7.573e-02	1.204e-01	2.109e-01	3.463e-01
12.0 km/s	1.995e-01	3.145e-01	5.414e-01	8.674e-01
14.0 km/s	6.750e-01	1.079e+00	1.847e+00	2.867e+00
16.0 km/s	1.491e+00	2.418e+00	4.146e+00	6.342e+00
18.0 km/s	2.501e+00	4.074e+00	7.169e+00	1.107e+01
$\begin{array}{c} \text{Rn} \\ \hline \text{V} \end{array}$	1 m	2.3 m	5 m	10 m
9.0 km/s	3.759e-03	7.889e-03	1.550e-02	2.731e-02
9.5 km/s	3.176e-02	5.744e-02	9.484e-02	1.424e-01
10.0 km/s	1.207e-01	1.965e-01	2.939e-01	4.052e-01
11.0 km/s	5.131e-01	7.974e-01	1.082e+00	1.389e+00
12.0 km/s	1.252e+00	1.804e+00	2.382e+00	2.918e+00
14.0 km/s	3.947e+00	5.270e+00	6.465e+00	7.443e+00
16.0 km/s	8.453e+00	1.066e+01	1.239e+01	1.452e+01
18.0 km/s	1.459e+01	1.773e+01	2.048e+01	2.234e+01

Table A.9: Radiative Heating Prediction at 78 km Altitude (Sutton and Hartung, 1990)

Predicted Radiative Heating, Q_r in $\frac{MW}{m^2}$				
$\begin{array}{c} \text{Rn} \\ \hline \text{V} \end{array}$.05 m	.1 m	.23 m	.5 m
9.0 km/s	4.728e-05	8.719e-05	1.833e-04	3.698e-04
9.5 km/s	6.853e-04	1.154e-03	2.193e-03	4.055e-03
10.0 km/s	3.168e-03	5.248e-03	9.373e-03	1.667e-02
11.0 km/s	1.640e-02	2.590e-02	4.585e-02	7.863e-02
12.0 km/s	4.333e-02	6.752e-02	1.168e-01	1.952e-01
14.0 km/s	1.435e-01	2.247e-01	3.873e-01	6.393e-01
16.0 km/s	3.153e-01	4.969e-01	8.626e-01	1.419e+00
18.0 km/s	5.798e-01	9.102e-01	1.529e+00	2.558e+00
$\begin{array}{c} \text{Rn} \\ \hline \text{V} \end{array}$	1 m	2.3 m	5 m	10 m
9.0 km/s	6.922e-04	1.476e-03	2.983e-03	5.630e-03
9.5 km/s	6.975e-03	1.317e-02	2.356e-02	3.890e-02
10.0 km/s	2.793e-02	5.058e-02	8.350e-02	1.251e-01
11.0 km/s	1.250e-01	2.068e-01	3.096e-01	4.250e-01
12.0 km/s	3.027e-01	4.817e-01	6.938e-01	9.142e-01
14.0 km/s	9.707e-01	1.478e+00	2.009e+00	2.493e+00
16.0 km/s	2.120e+00	3.245e+00	4.141e+00	4.840e+00
18.0 km/s	3.773e+00	5.593e+00	6.986e+00	7.984e+00

Table A.10: Radiative Heating Prediction at 84 km Altitude (Sutton and Hartung, 1990)

Predicted Radiative Heating, Q_r in $\frac{MW}{m^2}$				
$\begin{array}{c} \text{Rn} \\ \backslash \\ \text{V} \end{array}$.05 m	.1 m	.23 m	.5 m
9.0 km/s	9.285e-06	1.749e-05	3.748e-05	7.665e-05
9.5 km/s	1.423e-04	2.442e-04	4.653e-04	8.583e-04
10.0 km/s	6.142e-04	1.032e-03	1.929e-03	3.406e-03
11.0 km/s	3.181e-03	5.120e-03	9.155e-03	1.611e-02
12.0 km/s	8.620e-03	1.385e-02	2.462e-02	4.281e-02
14.0 km/s	3.181e-02	5.057e-02	8.757e-02	1.464e-01
16.0 km/s	7.707e-02	1.164e-01	2.049e-01	3.306e-01
18.0 km/s	1.245e-01	2.062e-01	3.536e-01	5.850e-01
$\begin{array}{c} \text{Rn} \\ \backslash \\ \text{V} \end{array}$	1 m	2.3 m	5 m	10 m
9.0 km/s	1.455e-04	3.138e-04	6.390e-04	1.199e-03
9.5 km/s	1.497e-03	2.927e-03	5.296e-03	8.960e-03
10.0 km/s	5.746e-03	1.090e-02	1.945e-02	3.146e-02
11.0 km/s	2.704e-02	4.958e-02	8.296e-02	1.233e-01
12.0 km/s	7.001e-02	1.230e-01	1.930e-01	2.724e-01
14.0 km/s	2.310e-01	3.682e-01	5.810e-01	7.695e-01
16.0 km/s	5.035e-01	8.223e-01	1.214e+00	1.572e+00
18.0 km/s	8.757e-01	1.434e+00	2.096e+00	2.649e+00

B Matlab Automation

```
%% Set title strings in cell arrays
entryheaders = {'Entry Altitude , km', 'Velocity , km', 'Body
    Density , kg/m^3', 'Nose radius , m', 'Entry Angle , degrees ', '
    Time step , s', 'Heat of Ablation', 'Yield Stength , MPa', 'Max
    Altitude of Atmosphere , km', 'Drag Coefficient ', 'Print
    Interval ', 'Density Tolerance '};
trajheaders = {'Time, s', 'Altitude , km', 'Velocity ,km/s', '
    Kinetic Energy, %', 'Stagnation Temperature Behind Shock, K
    ', 'Stagnation Pressure , MPa', 'Mass, Tons', 'Radius , m', '
    Qdot, W/cm^2', 'Enthalpy , J/kg', 'Density Before Shock , kg/m
    ^3', 'Density Behind Shock , kg/m^3', 'Goulard #'};

%% check for traj files before meteor.f90 runs and delete
% the fortran code will not run if the files exist
if exist('traj1.out', 'file') > 0
    system('rm traj*');
    pause(2);
end

warning off

%% Set constants for meteor program
he = 100000; %km
rhoc = 3500; %kg/m^3
dt = 0.0500000000000000002776; %s
qabl = 8500000; %
```



```

pyield_MPa = 10; %MPa
alton = 100000; %m
%Cd = 1.199999999999999556; %
Cd = 1.000000000000000; %
tprint = 70.989999999999994884;
rho_tol = 0.0100000000000000000208;
time = zeros(50000,1);
for i = 1:length(time);
    time(i) = (i-1).*0.05;
end
%% Set range of parameters to loop through
count = 0;
tic;
g = [8 15 30 45];
v = [12 14 16 18];
r = [.05 .1 .5 1 5 10];
for gammae = -1.*[15]; % negative entry angle in degress
    disp(['new angle= ',num2str(gammae)]);
    for ve = 1000.*[14]; %m/s
        disp(['new velocity= ',num2str(ve)]);
        for rn = [1]; % m
            disp(['new nose radius= ',num2str(rn)]);
            rnstr = rn*1000;
            gammaestr = abs(gammae);
            first = [he, ve, rhoc, rn, gammae, dt, qabl,
                pyield_MPa];
            second = [alton, Cd, tprint, rho_tol];
            entry = [first second];

```

```

    dlmwrite('entry.dat',first,'delimiter',' ',' ',
            precision',25)
    dlmwrite('entry.dat',second,'delimiter',' ',' ',
            precision',25','–append');
%% run meteor fortran executable
system('./origmet');
%system('./metallrad2')
%% import traj data and do something with it
traj1 = dlmread('traj1.out',' ',1,1);
traj2 = dlmread('traj2.out',' ',1,1);
    %traj3 = dlmread('traj3.out',' ',1,0);
system('rm traj*');
traj_length = length(traj1);
%trajdata = {time(1:traj_length) traj1 traj2
    traj3};
trajdata = {time(1:traj_length) traj1 traj2};
mat_file_mk = sprintf('gam%.0fve%.0frn%.0fmm2.mat
    ',gammaestr,ve,rnstr);
save(mat_file_mk,'trajdata','trajheaders');
mv_str = sprintf('mv %s Dec29/%s',mat_file_mk,
    mat_file_mk);
system(mv_str);
%
    save(mat_file_mk,'traj3');
%
    mv_str = sprintf('mv %s Apollorad/%s',
mat_file_mk,mat_file_mk);
%
    system(mv_str);
count = count + 1;
disp(['time elapsed= ',num2str(toc)]);

```

```

        total_trajs = length(g)*length(v)*length(r);
        disp(['traj num= ',num2str(count),' ', 'total
              trajs= ',num2str(total_trajs)]);

    end

end

end

```

C Fortran Code

```
PROGRAM METEOR
```

```
IMPLICIT NONE
```

```
! Meteor Trajectory Analysis Code
```

```
! Last modified: December 31, 2015
```

```
!
```

```
! For information contact:
```

```
!
```

```
! Dr. Evans Lyne
```

```
! University of Tennessee, Knoxville
```

```
! (423)974-5254
```

```
! This program was written by Evans Lyne and Rich Fought as a  
  model of meteor entry. It has since been modified by  
  Chris Thames and Jakob Brisby. Thames' modifications  
  allow modeling of meteors traveling at greater speeds.  
  The modifications made by Brisby allowed for small bodies  
  to be modeled by more accurately calculating density  
  ratios across the shock wave and introducing a new curve  
  fit algorithm for radiative heat calculation. For a more  
  thorough
```

```
! description see Rich Fought's thesis, Chris Thames' thesis,  
  and Jakob Brisby's thesis in the University of Tennessee,  
  Knoxville, library.
```

```
! Supporting files can also be found on Dr. Evans Lyne's site:  
  http://web.utk.edu/~comet/mollier.html
```

```

!      shock angle is determined using the method of FOUGHT &
      PALMER
!      shock angle at shoulder is 15 degrees. The code assumes
      a
!      cylinder of diameter 2r and length r with a
      hemispherical
!      nose cap at each end. Radiation is applied to the
      hemisphere.

!      VARIABLES
!
! AA      coefficient      m2/kg
! ablfrac  mass ablated to total mass -
! ablmass  mass lost to ablation  kg
! alton    max altitude of atm    m
! areasc   surface area      m2
! balcof   ballistic coefficient  kg/m2
! Cd       drag coefficient    -
! cmass    body mass          kg
! cmass2   body mass after ablation kg
! comp     comparison value    -
! Cp       specific heat, const press J/(kg-K)
! d2rdrdt2 d(dr/dt)/dt        m/s2
! d2rdrdt2o previous d(dr/dt)/dt m/s2
! drdrdt   dr/dt              m/s
! drdrdt0  previous dr/dt      m/s

```

```

! dt    time step      s
! diff  diff ratio in heat tran  -
! enth_Jkg  enthalpy      J/kg
! enth_stag  stagnation point enthalpy J/kg
! g      gravitational accel  m/s^2
! gamma  trajectory angle  degrees
! gammae  entry trajectory angle  degrees
! goulard  Goulard number  -
! goulardsp  Goulard number @ stag pt -
! G0      Earth grav accel @ surface m/s^2
! h      altitude      m
! he      entry altitude  m
! idiff  counter      -
! ind     indicator for DVERK routine -
! ith     angle at nose  degrees
! ithmax  max angle to sweep across degrees
! iwrite  counter      -
! J       counter      -
! MachNorm  Mach # of vel norm to shock -
! mdot    mass flw rate per unit area kg/(m^2-s)
! mdotratio  non-dim mass flow rate  -
! n       number of diff. eqtns. for -
!         IVPRK routine
! p1      atmospheric pressure  Pa
! p2      pressure behind shock  Pa
! pflag   crunch flag      -
! pstag   stagnation pressure  Pa
! pyield  body yield strength  Pa

```

```

! qabl  heat of ablation    J/kg
! qdotr  rad heat transfer rate  W/cm^2
! qdotrc  temp value holder   W/cm^2
! qdotrcad adiabatic rad heat tran rate W/cm^2
! qdotredg rad heat tran rate at edge W/cm^2
! qdotrlim 15% max rad heat tnsf rate W/cm^2
! qdotrnew temporary value holder  W/cm^2
! qdotrold prev rad heat transfer rate W/cm^2
! qdotrsp  stag pt rad heat tran rate W/cm^2
! qt      cumulative energy over time J
! qt1     energy on surf over 1 deg J
! qt2     cumulative energy on surf J
! rb      body nose radius    m
! rhoc    body density        kg/m^3
! rho1    atmospheric density  kg/m^3
! rho2    density of gas inside shock kg/m^3
! rho1_stag stag pt dens front of shock kg/m^3
! rho2_stag stag pt dens behind shock kg/m^3
! rho_old  previous density   kg/m^3
! rho_tol  rho2 solution tolerance -
! rn      initial body nose radius m
! R0      Earth radius       m
! scalefac scaling factor    -
! spdsnd  speed of sound     m/s
! t       time               s
! t1      atmospheric temperature K
! t2      temperature behind shock K
! t2sp    stagnation temperature K

```

```

! t2edg  temp behind shock@top edge K
! tprint  max time before output  s
! thetash_rad shockwave angle      rad
! tnext  value of 't' where solution -
!        is desired for IVPRK routine
! tol    tolerance for error control -
!        for IVPRK routine
! v      velocity      m/s
! ve     entry velocity  m/s
! v_norm norm vel in front of shock m/s
! xke    kinetic energy  J
! xke1   initial kinetic energy  J
! xkeold  previous kinetic energy  J
! xkerate rate of kin enrgy change W
! Y[3]   array containing:
!   Y(1)  velocity      m/s
!   Y(2)  trajectory angle  rads
!   Y(3)  altitude      m
!
!*****

! Common Block Declarations
COMMON /BLOCK1/ R0, AA

! Constants
DOUBLE PRECISION, PARAMETER :: PI = 3.14159265359
DOUBLE PRECISION, PARAMETER :: Cp = 1004.0
DOUBLE PRECISION, PARAMETER :: GAMALT = 1.40

```



```

INTEGER, PARAMETER :: LX = 35    ! TEMP_FIT
INTEGER, PARAMETER :: LY = 43

INTEGER, PARAMETER :: NN = 10

INTEGER, PARAMETER :: DX = 28    ! DENS_FIT
INTEGER, PARAMETER :: DY = 12

INTEGER, PARAMETER :: DN = 10

! External Subroutines
EXTERNAL FCN1, DVERK

! Declare Variables
INTEGER J, ind, ith, ithmax, iwrite, n, idiff
LOGICAL pflag
DOUBLE PRECISION enth_stag, comp, rho_tol, he, ve, rn, mdot,
    densratio, ifitflag
DOUBLE PRECISION mdotratio, spdsnd, gammae, dt, qabl, pyield
DOUBLE PRECISION MachNorm, alton, Cd, tprint, cmass, xke1
DOUBLE PRECISION xke, balcof, rb, d2rdt2, d2rdt2o, drdt
DOUBLE PRECISION drdto, qt, qt2, qdotr, qdotrold, t, v, h
DOUBLE PRECISION R0, AA, tol, qdotrlim, rho1, qt1, v_norm
DOUBLE PRECISION p2, p1, enth_Jkg, t1, rho_old, rhoc, rho2
DOUBLE PRECISION thetash_rad, pstag, t2sp, t2, rho1_stag
DOUBLE PRECISION rho2_stag, t2edg, qdotrcad, goulard, qdotrc
DOUBLE PRECISION goulardsp, qdotrnew, scalefac, diff,
    pyield_MPa
DOUBLE PRECISION qdotredg, areasc, ablmass, ablfrac, cmass2

```

```

DOUBLE PRECISION qdotrsp , xkeold , xkerate , gamma, tnext , g ,
    y1ft , y3ft
DOUBLE PRECISION qdotconstag , TW, hw,ht , qdotconith , newqr ,
    a , b , C3, sutqr
Double Precision qrnoblow

DOUBLE PRECISION Y, C, W
DOUBLE PRECISION XX, YY, ZZ, XXX, YYY, ZZZ

! Declare Arrays
DIMENSION Y(3) , C(24) , W(9,27)
DIMENSION XX(LX) , YY(LY) , ZZ(LX,LY)
DIMENSION XXX(DX) , YYY(DY) , ZZZ(DX,DY)

! Loading Vought Data – pressure , enthalpy , temperature
OPEN(UNIT=10, FILE="thermo1.dat" , STATUS='OLD')
READ(10,*) XX(1:LX)
J1: DO J=1, LY
    READ(10,*) YY(J) , ZZ(1:LX,J)
END DO J1
CLOSE(10)

! Loading Vought Data – enthalpy , pressure , density
OPEN(UNIT=20, FILE="thermo2.dat" , STATUS='OLD')
READ(20,*) XXX(1:DX)
J2: DO J=1, DY
    READ(20,*) YYY(J) , ZZZ(1:DX,J)
END DO J2

```

```

CLOSE(20)

! Open input file and get data
OPEN(UNIT=9,FILE='entry.dat',STATUS='OLD')
    READ(9,*) he, ve, rhoc, rn, gammae, dt, qabl, pyield_MPa
    READ(9,*) alton, Cd, tprint, rho_tol
CLOSE(9)

! Open output files
OPEN(UNIT=1, FILE="traj1.out", STATUS='NEW')
OPEN(UNIT=2, FILE="traj2.out", STATUS='NEW')
open(unit=3, file="traj3.out", Status='NEW')

! Set common block values
R0 = 6378135.0
    cmass = rhoc * PI * 4.0/3.0 * rn**3
    balcof = cmass / (Cd*PI*rn**2)
AA = 1.0 / (2.0*balcof)

! Initialize values
    n = 3
    ind = 1
    ithmax = 90
    iwrite = 0
    d2rdt2 = 0.0
    d2rdt2o = 0.0
    drdt = 0.0
    drdto = 0.0

```

```

      h = he
      pflag = .FALSE.
      pyield = pyield_MPa * 1000000.0
      qt = 0.0
      qt1 = 0.0
      qt2 = 0.0
      qdotr = 0.0
      qdotrold = 0.0
      rb = rn
      tol = 0.001
      t = 0.0
      v = ve
      xke1 = 0.5 * cmass * ve**2
      xke = xke1
      Y(1) = ve
      Y(2) = PI * (gammae/180.0)
      Y(3) = he

! Calculate gravitational accel, density, and viscosity at
  altitude 'h'
      CALL ATMOS76(h, g, rho1, t1, p1, spdsnd)

! Loop until altitude is below 1 m or kinetic energy is at
  less than 0.01%
TRAJ: DO

! *** CALCULATE HEATING INFORMATION ***

```

```

! Max heat transfer , 15% limit , converted to W/cm^2,
! from experimentation and Anderson [Hypersonic , p. 289]
qdotrlim = 0.0001 * 0.15 * (0.5*rho1*V**3)

! Save previous value
qdotrold = qdotr

! call sutton equation
call sutfit(rn,v,rho1,sutqr)

! Step around nose of body calculating temperature and qdot
I1: DO ith=0, ithmax, 1

! Calculate shock angle using an experimental value for
! approx. shape of shock
! [Fought , pg. 23]
thetash_rad = (-0.833333*(ith/90.0)*PI/2.0) + 1.5708

! Determine normal velocity outside shock
v_norm = v * SIN(thetash_rad)

! Make an initial approximate calculation of pressure and
! enthalpy inside
! the shock
p2 = p1 + rho1 * v_norm**2
enth_Jkg = Cp*t1 + v_norm**2 / 2.0

```

```

! Initialize
rho_old = 0.0

! Loop to determine correct pressure , enthalpy , density
  combination
PRESS: DO

  y1ft = Y(1)*3.28084
  y3ft = Y(3)*3.28084
  ifitflag = 1

! One option for points using Vought curve fits if in
  range and
! the other for the rest

if ((y1ft<13000) .AND. (v/spdsnd > 1) .AND. (enth_Jkg <
  7445805.12) ) then

  call DENS_FIT3(y1ft,y3ft,densratio)
  ifitflag = 0
  rho2 = densratio*rho1
  exit PRESS

end if

IF ((enth_Jkg > 7445805.12) .AND. (ifitflag == 1))THEN

```

```

! Vought density curve fit
CALL DENS_FIT(p2, enth_Jkg, rho2, XXX, YYY, ZZZ, DX, DY,
             DN)

End if

if ((enth_Jkg < 7445805.12) .AND. (ifitflag==1)) then

! Eqtns from Anderson [Compressible, pg. 106]
MachNorm = v/spdsnd * SIN(thetash_rad)
rho2 = rho1 * ((GAMALT + 1.0)*MachNorm**2)/((GAMALT -
      1.0)*MachNorm**2 + 2.0)

END IF

if (v/spdsnd < 1) then
rho2 = rho1
end if

! Determine if density is within tolerance
comp = ABS((rho2-rho_old)/rho2)
IF (comp < rho_tol) THEN
EXIT PRESS
END IF

! Save previous value
rho_old = rho2

```

```

! Resolve pressure and enthalpy behind the shock [
    Anderson, Hypersonics, pg. 508]
    p2 = p1 + rho1 * v_norm**2 * (1.0 - rho1/rho2)
    enth_Jkg = Cp*t1 + v_norm**2 / 2.0 * (1.0 - (rho1/rho2)
        **2)

END DO PRESS

! Determine whether to use max limit b/c atm is too thin
    for reasonable values
IF (h > alton) THEN

    qdotr = qdotrlim

    ELSE

! Determine temperature inside shock using either Vought
    or Fought curve fit
IF (enth_Jkg > 7445805.12) THEN

    ! Vought temperature curve fit
    CALL TEMP_FIT(enth_Jkg, p2, t2, XX, YY, ZZ, LX, LY, NN)

ELSE

    ! Fought curve fit
    CALL TEMPNEW(enth_Jkg, p2, t2)

```



```

END IF

! Print time, angle, and temp if time, t, exceeds tprint
!IF (t > tprint) THEN
! WRITE(6,*) t, ith, t2
!END IF

! ** USE GOULARD NUMBER TO ACCOUNT FOR RADIATIVE COOLING
! OF SHOCK LAYER **

! Apply Stefan-Boltzman equation [Incopera, pg. 10]
qdotrcad = 5.67 * t2**4 / (1.0e12)

! Modify Stefan-Boltzman result according to Goulard
! number
! from Lyne [Geophysical Research, pg. 23208]
goulard = 10000.0 * 4.0 * qdotrcad/(rho1*v**3)
qdotrc = qdotrcad / (1.0 + 3.33*goulard**0.7)
qdotr = qdotrc

!call my function for qdotr

call qfit(v,rb,rho1,newqr)
newqr = newqr*100 !convert from MW/m^2 to W/cm^2
qdotr = newqr

! Determine stagnation point and edge values

```

```

IF (ith == 0) THEN

    pstag = rho1 * V**2 * (1.0 - rho1/rho2)
    t2sp = t2
    enth_stag = enth_Jkg
    rho1_stag = rho1
    rho2_stag = rho2
    goulardsp = goulard

ELSE IF (ith == ithmax) THEN

    t2edg = t2

END IF

! APPLY 15% FLOW ENERGY LIMIT
IF (qdotrc > qdotrlim) THEN
    qdotr = qdotrlim
END IF
qrnoblw = qdotrsp
!if (ith ==0) then
!write(*,*)  qdotrsp," ", newqr," ", sutqr," ", h," ", v
    ," ", rn," ", rho1," ", t, " | "
!end if

! ** APPLY GREEN AND NICOLET'S CORRECTION TO ACCOUNT FOR
    BLOWING **

! ** OF WALL CAUSING REDUCED RADIATIVE HEATING (FOR SIO2)
    **

```

```

! qdotr is in W/cm^2, heat of ablation is in Joules/kg.
      This
! results in a conversion factor of 10000
qdotrnew = qdotr
idiff = 0

! Loop until (diff > 0.02) and (idiff < 200)
DIFF1:  DO
      mdot = (qdotrnew/qabl) * 10000.0
      mdotratio = mdot / (rho1*V)
      IF (mdotratio < 0.04802734866) THEN
            scalefac = 1.0
      ELSE IF (mdotratio > 0.754793198787) THEN
            scalefac = 0.5
      ELSE
            scalefac = 1.0095555-0.11552317*mdotratio &
            -1.7206431*mdotratio**2-0.53616033*mdotratio**3 &
            +4.1602124*mdotratio**4-2.2932833*mdotratio**5
      END IF

      qdotrold = qdotrnew
      qdotrnew = scalefac * qdotr
      diff = ABS(qdotrold - qdotrnew) / qdotrold
      idiff = idiff + 1

      IF (idiff == 1000) THEN
            WRITE(6,*) 202020

```

```

END IF

IF ((diff < 0.02) .OR. (idiff > 200)) THEN
  EXIT DIFF1
END IF

END DO DIFF1

qdotrc = qdotrnew
qdotr = qdotrc

END IF

IF (ith == 0) THEN

  ! Calculate stagnation point value
  qdotrsp = qdotr
  qdotconstag = (1.83/(10**8))*((rho1/rb)**.5)*(V**3)
  TW = (qdotconstag/(.8*(5.67/(10**8))))**.25
  hw = 940*TW+.1043*(TW**2)
  ht = (V**2)/2+Cp*t1
  qdotconstag = qdotconstag*(1-hw/ht)

  !call my function for qdotr
  call qfit(v,rb,rho1,newqr)
  newqr = newqr*100 !convert from MW/m^2 to W/cm^2

```

```

ELse

! Calculate edge point value
IF (ith == ithmax)  qdotredg = qdotr

! Multiply heating rate by surface area to get heat input
qdotconith = qdotconstag
qdotconith = qdotconith*COS(ith/180.0*PI)
areasc = 2.0*PI * rb**2 * (COS((ith-1)/180.0*PI) - COS(
    ith/180.0*PI))
qt1 = 1.0e4 * (qdotr+qdotconith) * dt * areasc
qt2 = qt2 + qt1
! write(*,*) t," ",ith," ",h," ",V," ",rho1," ",t2," ",
    qdotconith," ",qdotr,"|"
qt1 = 0.0

END IF

END DO I1

! Sum heat input
qt = qt2 + qt
qt2 = 0.0

! Determine the amount of mass ablating
ablmass = qt / qabl
ablfrac = ablmass / cmass

```

```

!WRITE(*,*) 'cmass= ',cmass,'cmass2= ', cmass2,'ablmass= ',
    ablmass
! Determine new mass of meteor
cmass2 = cmass * (1 - ablfrac)
IF (cmass2 < 0.0) THEN
    cmass2 = 0.0
END IF

! *** CALCULATE FRACTURE MECHANICS INFORMATION ***

! calculate radius increase according to Zahnle's model
! once material yield limit is exceeded and using the
! Trapezoid Rule to solve the diff-eqtn
IF ((pyield < pstag/2.0) .OR. (pflag)) THEN

    d2rdt2 = (Cd * rho1 * V**2 / rhoc / rb) / 2.0
    drdt = drdt - 0.5 * (d2rdt2 + d2rdt2o) * dt
    rb = rb - 0.5 * (drdt + drdto) * dt

    drdto = drdt
    d2rdt2o = d2rdt2
    pflag = .TRUE.

ELSE

    ! Radius change due to ablation
    rb = ((3.0*cmass2)/(4.0*PI*rhoc))**(1.0/3.0)

```

```

END IF

! Calculate new ballistic coef and AA coef
balcof = cmass2 / (Cd * PI * rb**2)
IF (cmass2 > 0.0) THEN
  AA = 1.0 / (2.0 * balcof)
END IF

! Save previous kinetic energy and calculate new value
xkeold = xke
xke = 0.5 * cmass2 * Y(1)**2

! Calculate rate of energy loss
xkerate = (xkeold - xke) / dt

! Convert trajectory angle to degrees
gamma = Y(2) * 180.0 / PI

! Print title line every 10 iterations
iwrite = iwrite + 1
IF (iwrite == 1) THEN
  WRITE(1,91)
  WRITE(2,93)
END IF

! Output data at each iteration
WRITE(1,90) t, Y(3)/1000.0, Y(1)/1000.0, xke/xke1*100.0,
  t2sp, pstag/1000000.0, &

```

```

      cmass2/1000.0, rb
90  FORMAT(F8.2, 2X, F7.2, 2X, F9.4, 2X, F8.1, 2X, F9.0, 2X,
      F10.4, 2X, &
      F15.8, 2X, F12.5)
91  FORMAT(' TIME      alt      vel      %ke      t2sp
      pstag      mass      radius ')
      WRITE(2,92) t, Y(3)/1000.0, qdotrsp, enth_stag/1000000.0,
      rho1_stag, rho2_stag, goulardsp
92  FORMAT(F8.2, 2X, F7.2, 2X, F13.4, 2X, F9.4, 3X, F10.6, 3X
      , F10.6, 2X, F12.4)
93  FORMAT(' Time      alt      Qdot      enth
      rho1      rho2      goulard ')

      write(3,94) qdotconstag, newqr, ht, hw, TW, rho1, t
94  FORMAT(F18.4, 2X, F18.4, 2X, F13.2, 2X, F13.2, 3X, F13.3,
      2X, F12.10, 2X, F8.2)

! Increment time
tnext = t + dt

! *** CALCULATE TRAJECTORY INFORMATION ***

! Call DVERK – first order differential equation solver to
! solve the equations of motion; automatically increments
! time, t, by dt
CALL DVERK(n, FCN1, t, Y, tnext, tol, ind, C, 9, W)

! Set the new altitude

```



```

h = Y(3)

! If the altitude is less than 1 meter or kinetic energy
    less than
! 0.01% then end
! write(*,*) 'Mach number',v/spdsnd,'Altitude',Y(3)
IF ((h < 1.0) .OR. (Y(3) > 115000) .OR. (xke/xke1 <
    .000001)) then
write(*,*) 'h=',h, 'Alt=',Y(3), 'xke/xke1=',xke/xke1
EXIT TRAJ
END IF

! Determine atmospheric conditions at new altitude
CALL ATMOS76(h, g, rho1, t1, p1, spdsnd)

! Set the new velocity
v = Y(1)

! Screen Output
! WRITE(*,*) Y(3)/1000.0

END DO TRAJ

END PROGRAM METEOR

```

```

!*****

```

```

! FCN1
!   Routine to evaluate the righthand side of the
      differential equations
!   of ballistic trajectory motion.
!*****

SUBROUTINE FCN1(N,t,Y,YPRIME)
  IMPLICIT NONE

  INTEGER N
  DOUBLE PRECISION t, Y, YPRIME

  DIMENSION Y(N), YPRIME(N)

  ! Common Block
      COMMON /BLOCK1/ R0, AA
  DOUBLE PRECISION R0, AA

  ! Declare Variables
  DOUBLE PRECISION v, h, sgamma, cgamma, R, g
  DOUBLE PRECISION a1, a2, v2, rho1, p1, t1, spdsnd

  ! Set values for differential equations
  v = Y(1)
      h = Y(3)
      sgamma = SIN(Y(2))
      cgamma = COS(Y(2))

```

```

R = R0 + h

! Determine density at current altitude
CALL ATMOS76(h,g,rho1,t1,p1,spdsnd)

a1 = rho1 * AA
v2 = v**2
a2 = g - v2/R

! Differential Equations of Motion [Vinh et. al., pg. 28]
YPRIME(1) = -a1 * v2 - g * sgamma
YPRIME(2) = -a2 * cgamma/v
YPRIME(3) = v * sgamma

RETURN
END SUBROUTINE FCN1

!*****

! DENS_FIT3
!
!
!*****

subroutine DENS_FIT3(y1ft,y3ft,densratio)
double precision y1ft, y3ft, densratio, t, b, x, ta, ba,
percdiff, densdiff, alt

```

```

x = y1ft
alt = y3ft

if (x<=9000) then
if ((alt <322900) .AND. (alt >294800)) Then
    ta = 322900
    ba = 294800

    t = 8.947930 + 2.574996*cos(x*0.000592) + 0.480420*sin(x
        *0.000592)

    b = 6.744912 - 0.435223*cos(x*0.000478) - 2.913602*sin(x
        *0.000478) &
    - 0.684476*cos(2*x*0.000478) - 0.594066*sin(2*x*0.000478)

end if

if ((alt <294800) .AND. (alt >259700)) then
    ta = 294800
    ba = 259700

    t = 6.744912 - 0.435223*cos(x*0.000478) - 2.913602*sin(x
        *0.000478) &
    - 0.684476*cos(2*x*0.000478) - 0.594066*sin(2*x*0.000478)

    b = 7.835242*exp(-((x-10480.238707)/3032.694926)**2) +
        5.341591*exp(-((x-5266.257128)/4650.029554)**2)

```

```

end if

if ((alt < 259700) .AND. (alt > 230400)) then
  ta = 259700
  ba = 230400

  t = 7.835242*exp(-((x-10480.238707)/3032.694926)**2) +
      5.341591*exp(-((x-5266.257128)/4650.029554)**2)

  b = 4.252206*exp(-((x-9943.609700)/2081.869232)**2) +
      5.970174*exp(-((x-6746.803628)/5940.161121)**2)

end if

if ((alt < 230400) .AND. (alt > 200100)) then
  ta = 230400
  ba = 200100

  t = 4.252206*exp(-((x-9943.609700)/2081.869232)**2) +
      5.970174*exp(-((x-6746.803628)/5940.161121)**2)

  b = 4.897217*exp(-((x-10533.108308)/2621.115930)**2) +
      5.667983*exp(-((x-6638.610203)/5901.823490)**2)

end if

if ((alt < 200100) .AND. (alt > 173500)) then
  ta = 200100

```

ba = 173500

t = 4.897217*exp(-((x-10533.108308)/2621.115930)**2) +
5.667983*exp(-((x-6638.610203)/5901.823490)**2)

b = 2.349407*exp(-((x-9742.109724)/1765.964401)**2) +
6.210003*exp(-((x-7946.948959)/7022.738360)**2)

end if

IF ((alt < 173500) .AND. (alt > 154800)) then

ta = 173500

ba = 154800

t = 2.349407*exp(-((x-9742.109724)/1765.964401)**2) +
6.210003*exp(-((x-7946.948959)/7022.738360)**2)

b = 2.497634*exp(-((x-9940.407545)/1929.716063)**2) +
6.143941*exp(-((x-7811.977224)/6896.484229)**2)

end if

if ((alt < 154800) .AND. (alt > 120300)) then

ta = 154800

ba = 120300

t = 2.497634*exp(-((x-9940.407545)/1929.716063)**2) +
6.143941*exp(-((x-7811.977224)/6896.484229)**2)

```

b = 1.458985*exp(-((x-9746.148083)/1561.439415)**2) +
    6.570425*exp(-((x-8688.202315)/7970.875296)**2)

end if

if ((alt <120300) .AND. (alt >100000)) then
    ta = 120300
    ba = 100000

    t = 1.458985*exp(-((x-9746.148083)/1561.439415)**2) +
        6.570425*exp(-((x-8688.202315)/7970.875296)**2)

    b = 1.123350*exp(-((x-9710.148086)/1477.022690)**2) +
        6.701312*exp(-((x-9059.556231)/8382.227364)**2)

end if

if ((alt <100000) .AND. (alt >82200)) then
    ta = 100000
    ba = 82200

    t = 1.123350*exp(-((x-9710.148086)/1477.022690)**2) +
        6.701312*exp(-((x-9059.556231)/8382.227364)**2)

    b = 3.295163*exp(-((x-10907.904523)/2672.434393)**2) +
        5.956844*exp(-((x-7138.420086)/6768.223118)**2)

```

```

end if

if ((alt < 82200) .AND. (alt > 59800)) then
    ta = 82200
    ba = 59800

    t = 3.295163*exp(-((x-10907.904523)/2672.434393)**2) +
        5.956844*exp(-((x-7138.420086)/6768.223118)**2)

    b = 6.877943*exp(-((x-11904.509124)/4373.408166)**2) +
        4.870239*exp(-((x-5345.792170)/5181.183316)**2)
end if

if ((alt < 59800) .AND. (alt > 35900)) then
    ta = 59800
    ba = 35900

    t = 6.877943*exp(-((x-11904.509124)/4373.408166)**2) +
        4.870239*exp(-((x-5345.792170)/5181.183316)**2)

    b = 6.421499*exp(-((x-11142.837712)/4031.784877)**2) +
        4.792749*exp(-((x-5124.989520)/4769.217369)**2)

end if

if ((alt < 35900) .AND. (alt >= 0)) then

    ta = 35900

```



```

ba = 0
t = 6.421499*exp(-((x-11142.837712)/4031.784877)**2) +
    4.792749*exp(-((x-5124.989520)/4769.217369)**2)

b = 2.400060*exp(-((x-11388.844411)/3456.501583)**2) +
    6.065863*exp(-((x-7860.969194)/8639.747216)**2)
end if

if (t>b) then
percdiff = (alt-ba)/((ta-alt)+(alt-ba))
densdiff = t-b
densratio = densdiff*percdiff+b
end if

if (b>t) then
percdiff =(ta-alt)/((alt-ba)+(ta-alt))
densdiff = b-t
densratio = densdiff*percdiff+t
end if

!write(*,*) t,b,densratio,'Less than 9k'
! END < 9000
end if

if (x>9000) then
if ((alt<322900) .AND. (alt>294800)) Then
    ta = 322900
    ba = 294800

```

```

t = 11.563164 + -0.879523*cos(x*0.000643) + 1.535478*sin(x
    *0.000643) + 0.635560*cos(2*x*0.000643) &
+ 0.738088*sin(2*x*0.000643) + -0.090300*cos(3*x*0.000643) +
    -0.285679*sin(3*x*0.000643)

b = 11.492025 - 1.290057*cos(x*0.000757) - 0.451344*sin(x
    *0.000757) - 0.546117*cos(2*x*0.000757) &
- 0.464962*sin(2*x*0.000757) - 0.250490*cos(3*x*0.000757) +
    0.148663*sin(3*x*0.000757)

end if

if ((alt < 294800) .AND. (alt > 259700)) then
    ta = 294800
    ba = 259700

t = 11.492025 - 1.290057*cos(x*0.000757) - 0.451344*sin(x
    *0.000757) - 0.546117*cos(2*x*0.000757) &
- 0.464962*sin(2*x*0.000757) - 0.250490*cos(3*x*0.000757) +
    0.148663*sin(3*x*0.000757)

b = 0.131162*exp(-((x-12350.724491)/525.871188)**2) +
    6.723413*exp(-((x-15875.286705)/1847.816678)**2) &
+ 4.163375*exp(-((x-12655.943985)/2754.416413)**2)+
    8.593603*exp(-((x-10204.364413)/6304.839654)**2)

end if

```

```

if ((alt < 259700) .AND. (alt > 230400)) then
  ta = 259700
  ba = 230400

  t = 0.131162*exp(-((x-12350.724491)/525.871188)**2) +
    6.723413*exp(-((x-15875.286705)/1847.816678)**2) &
+ 4.163375*exp(-((x-12655.943985)/2754.416413)**2)+
    8.593603*exp(-((x-10204.364413)/6304.839654)**2)

  b = 10.626446 - 0.793770*cos(x*0.000590) + 1.503455*sin(x
    *0.000590) + 0.855377*cos(2*x*0.000590) &
+ 0.256204*sin(2*x*0.000590) - 0.258544*cos(3*x*0.000590) -
    0.132340*sin(3*x*0.000590)

end if

if ((alt < 230400) .AND. (alt > 200100)) then
  ta = 230400
  ba = 200100

  t = 10.626446 - 0.793770*cos(x*0.000590) + 1.503455*sin(x
    *0.000590) + 0.855377*cos(2*x*0.000590) &
+ 0.256204*sin(2*x*0.000590) - 0.258544*cos(3*x*0.000590) -
    0.132340*sin(3*x*0.000590)

```

```

b = 9.968571 - 1.423545*cos(x*0.000764) - 0.614223*sin(x
    *0.000764) - 0.379327*cos(2*x*0.000764) &
- 0.143922*sin(2*x*0.000764) - 0.199100*cos(3*x*0.000764) +
    0.061089*sin(3*x*0.000764) &
- 0.011690*cos(4*x*0.000764) + 0.057147*sin(4*x*0.000764)

```

```

end if

```

```

if ((alt < 200100) .AND. (alt > 173500)) then

```

```

    ta = 200100

```

```

    ba = 173500

```

```

t = 9.968571 - 1.423545*cos(x*0.000764) - 0.614223*sin(x
    *0.000764) - 0.379327*cos(2*x*0.000764) &
- 0.143922*sin(2*x*0.000764) - 0.199100*cos(3*x*0.000764) +
    0.061089*sin(3*x*0.000764) &
- 0.011690*cos(4*x*0.000764) + 0.057147*sin(4*x*0.000764)

```

```

b = 16555925.601673*exp(-((x-44312.453328)/7624.331579)**2)
    + 4.406157*exp(-((x-13345.324839)/2720.798936)**2) &
+ 8.112626*exp(-((x-10055.443410)/4919.805342)**2)+
    0.166632*exp(-((x-13440.263640)/1159.929220)**2)

```

```

end if

```

```

IF ((alt < 173500) .AND. (alt > 154800)) then

```

```

    ta = 173500

```

```

    ba = 154800

```

```

t = 16555925.601673*exp(-((x-44312.453328)/7624.331579)**2)
    + 4.406157*exp(-((x-13345.324839)/2720.798936)**2) &
+ 8.112626*exp(-((x-10055.443410)/4919.805342)**2)+
    0.166632*exp(-((x-13440.263640)/1159.929220)**2)

```

```

b = 7.538234 - 0.251024*cos(x*0.000360) - 3.129732*sin(x
    *0.000360) &
- 0.093967*cos(2*x*0.000360) - 0.223799*sin(2*x*0.000360)

```

```

end if

```

```

if ((alt <154800) .AND. (alt >120300)) then

```

```

    ta = 154800

```

```

    ba = 120300

```

```

t = 7.538234 - 0.251024*cos(x*0.000360) - 3.129732*sin(x
    *0.000360) &
- 0.093967*cos(2*x*0.000360) - 0.223799*sin(2*x*0.000360)

```

```

b = 6.162598*exp(-((x-14812.671443)/4992.077943)**2) +
    6.183455*exp(-((x-9540.950660)/7716.790931)**2)

```

```

end if

```

```

if ((alt <120300) .AND. (alt >100000)) then

```

```

ta = 120300
ba = 100000

t = 6.162598*exp(-((x-14812.671443)/4992.077943)**2) +
    6.183455*exp(-((x-9540.950660)/7716.790931)**2)

b = 8.697589*exp(-((x-15064.093185)/8083.951163)**2) +
    2.866774*exp(-((x-5774.419138)/11275.450358)**2)

end if

if ((alt < 100000) .AND. (alt > 82200)) then
    ta = 100000
    ba = 82200

    t = 8.697589*exp(-((x-15064.093185)/8083.951163)**2) +
        2.866774*exp(-((x-5774.419138)/11275.450358)**2)

    b = 8.739264*exp(-((x-14890.189854)/6959.335431)**2) +
        4.401245*exp(-((x-3570.935198)/9816.292681)**2)

end if

if ((alt < 82200) .AND. (alt > 59800)) then
    ta = 82200
    ba = 59800

```

```

t = 8.739264*exp(-((x-14890.189854)/6959.335431)**2) +
    4.401245*exp(-((x-3570.935198)/9816.292681)**2)

b = 8.262919*exp(-((x-15157.349515)/7361.484781)**2) +
    4.346447*exp(-((x-3131.978378)/11163.533748)**2)

end if

if ((alt < 59800) .AND. (alt > 35900)) then
    ta = 59800
    ba = 35900

    t = 8.262919*exp(-((x-15157.349515)/7361.484781)**2) +
        4.346447*exp(-((x-3131.978378)/11163.533748)**2)

    b = 7.401731*exp(-((x-15986.230140)/7982.714111)**2) +
        4.286769*exp(-((x-5391.763808)/11698.184829)**2)

end if

if ((alt < 35900) .AND. (alt >= 0)) then
!write(*,*) 'NEW FIT'
    ta = 35900
    ba = 0

    t = 7.401731*exp(-((x-15986.230140)/7982.714111)**2) +
        4.286769*exp(-((x-5391.763808)/11698.184829)**2)

```

```

      b = 9.200852*exp(-((x-15782.991029)/10767.778653)**2) +
          1.493206*exp(-((x-7656.903077)/3624.069550)**2)
end if

if (t>b) then
  percdiff = (alt-ba)/((ta-alt)+(alt-ba))
  densdiff = t-b
  densratio = densdiff*percdiff+b
end if

if (b>t) then
  percdiff = (ta-alt)/((alt-ba)+(ta-alt))
  densdiff = b-t
  densratio = densdiff*percdiff+t
end if

!write(*,*) t,b,densratio,'Greater than 9k'
! END > 9000
end if

return
end Subroutine

!*****

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! Radiative heating based on Sutton &
Hartung tables!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

```


!!

!*****

```
subroutine qfit(v,rn,rho1,newqr)
double precision v,rn,rho1,newqr, x, rhodiff, qnoselow,
    qnosehigh, rndiff, qnose1, qnose2, nose
newqr = 0.0
x = v
rho = rho1
nose = rn

IF((rho<=0.0184100000).AND.(rho>=0.0072579000)) THEN
    rhodiff = (0.0184100000-rho)/(0.0184100000-0.0072579000)
    rndiff = 1.0-(0.100000-nose)/(0.100000-0.049999)
    IF((nose>0.0499990).AND.(nose<=0.10)) THEN
        IF((x>=0).AND.(x<8000))THEN
            qnoselow = (1.66500000000000001e-04)*(x) +
                (0.00000000000000000e+00)
        END IF
        IF((x>=8000).AND.(x<8500))THEN
            qnoselow = (1.6341085271317830e-09)*(x-8000)**3 +
                (-6.0716279069767404e-07)*(x-8000)**2 &
                + (1.6470542635658916e-03)*(x-8000) + (1.33200000000000001
                    e+00)
        END IF
        IF((x>=8500).AND.(x<9000))THEN
```

```

qnoselow = (1.6341085271317919e-09)*(x-8500)**3 +
(1.8439999999999950e-06)*(x-8500)**2 &
+ (2.2654728682170548e-03)*(x-8500) + (2.2080000000000002
e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnoselow = (1.2549457364341073e-08)*(x-9000)**3 +
(4.2951627906976789e-06)*(x-9000)**2 &
+ (5.3350542635658936e-03)*(x-9000) + (4.0060000000000002
e+00)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnoselow = (1.6504062015503873e-08)*(x-9500)**3 +
(2.3119348837209302e-05)*(x-9500)**2 &
+ (1.9042310077519376e-02)*(x-9500) + (9.3160000000000007
e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnoselow = (-2.1951472868217450e-09)*(x-10000)**3 +
(4.7875441860465166e-05)*(x-10000)**2 &
+ (5.4539705426356583e-02)*(x-10000) +
(2.6680000000000000e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnoselow = (-2.1951472868216569e-09)*(x-11000)**3 +
(4.1289999999999966e-05)*(x-11000)**2 &
+ (1.4370514728682168e-01)*(x-11000) +
(1.2690000000000001e+02)

```

```

END IF
if ((x>=0).AND.(x<8000))THEN
qnosehigh = (3.0125000000000003e-04)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
qnosehigh = (1.9036279069767454e-09)*(x-8000)**3 +
(5.6558139534883410e-08)*(x-8000)**2 &
+ (2.3058139534883715e-03)*(x-8000) + (2.4100000000000001
e+00)
END IF
IF ((x>=8500).AND.(x<9000))THEN
qnosehigh = (1.9036279069767425e-09)*(x-8500)**3 +
(2.91200000000000019e-06)*(x-8500)**2 &
+ (3.7900930232558140e-03)*(x-8500) + (3.814999999999999
e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnosehigh = (1.7897860465116284e-08)*(x-9000)**3 +
(5.7674418604651116e-06)*(x-9000)**2 &
+ (8.1298139534883729e-03)*(x-9000) + (6.6760000000000002
e+00)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnosehigh = (1.9568930232558179e-08)*(x-9500)**3 +
(3.2614232558139504e-05)*(x-9500)**2 &
+ (2.7320651162790699e-02)*(x-9500) + (1.4420000000000000
e+01)

```

```

END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (-4.9592093023255567e-09)*(x-10000)**3 +
    (6.1967627906976737e-05)*(x-10000)**2 &
+ (7.4611581395348833e-02)*(x-10000) +
    (3.8680000000000000e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (-4.9592093023257131e-09)*(x-11000)**3 +
    (4.70900000000000079e-05)*(x-11000)**2 &
+ (1.8366920930232564e-01)*(x-11000) +
    (1.70300000000000001e+02)
END IF
  qnosel = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<8000))THEN
  qnoselow = (3.1162499999999999e-05)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
  qnoselow = (3.2539085941380988e-10)*(x-8000)**3 +
    (9.5713710879285063e-08)*(x-8000)**2 &
+ (2.4119542970690500e-04)*(x-8000) + (2.4929999999999999
    e-01)
END IF
IF ((x>=8500).AND.(x<9000))THEN
  qnoselow = (3.2539085941380937e-10)*(x-8500)**3 +
    (5.83800000000000013e-07)*(x-8500)**2 &

```

```

+ (5.8095228514654746e-04)*(x-8500) + (4.3450000000000000
e-01)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnoselow = (6.2002457029309555e-09)*(x-9000)**3 +
    (1.0718862891207128e-06)*(x-9000)**2 &
+ (1.4087954297069046e-03)*(x-9000) + (9.115999999999997
e-01)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnoselow = (4.5320263288623894e-09)*(x-9500)**3 +
    (1.0372254843517140e-05)*(x-9500)**2 &
+ (7.1308659960258340e-03)*(x-9500) + (2.6589999999999998
e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (-2.9643492300048195e-10)*(x-10000)**3 +
    (1.7170294336810713e-05)*(x-10000)**2 &
+ (2.0902140586189766e-02)*(x-10000) +
    (9.3840000000000003e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (5.5859413810108040e-12)*(x-11000)**3 +
    (1.6280989567809240e-05)*(x-11000)**2 &
+ (5.4353424490809744e-02)*(x-11000) +
    (4.7159999999999997e+01)
END IF
if ((x>=0).AND.(x<8000))THEN

```

```

qnosehigh = (5.684999999999999e-05)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
  qnosehigh = (5.8671038251366123e-10)*(x-8000)**3 +
(-8.1665573770492560e-08)*(x-8000)**2 &
+ (5.1615519125683114e-04)*(x-8000) + (4.5479999999999998
e-01)
END IF
IF ((x>=8500).AND.(x<9000))THEN
  qnosehigh = (5.8671038251366113e-10)*(x-8500)**3 +
(7.9840000000000006e-07)*(x-8500)**2 &
+ (8.7452240437158448e-04)*(x-8500) + (7.6580000000000004
e-01)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (8.0872480874316866e-09)*(x-9000)**3 +
(1.6784655737704964e-06)*(x-9000)**2 &
+ (2.1129551912568304e-03)*(x-9000) + (1.4760000000000000
e+00)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (4.9298972677595492e-09)*(x-9500)**3 +
(1.3809337704918042e-05)*(x-9500)**2 &
+ (9.8568568306010920e-03)*(x-9500) + (3.9630000000000001
e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN

```

```

      qnosehigh = (-8.7780109289614607e-10)*(x-10000)**3 +
      (2.1204183606557345e-05)*(x-10000)**2 &
+ (2.7363617486338795e-02)*(x-10000) +
      (1.29600000000000001e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
      qnosehigh = (-4.5936174863396767e-10)*(x-11000)**3 +
      (1.8570780327868924e-05)*(x-11000)**2 &
+ (6.7138581420765051e-02)*(x-11000) +
      (6.0649999999999999e+01)
END IF
      qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
      newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
      rndiff = 1.0-(0.230000-nose)/(0.230000-0.100000)
IF ((nose>0.1000000).AND.(nose<=0.23)) THEN
      IF ((x>=0).AND.(x<8000))THEN
      qnoselow = (3.01250000000000003e-04)*(x) +
      (0.0000000000000000e+00)
END IF
      IF ((x>=8000).AND.(x<8500))THEN
      qnoselow = (1.9036279069767454e-09)*(x-8000)**3 +
      (5.6558139534883410e-08)*(x-8000)**2 &
+ (2.3058139534883715e-03)*(x-8000) + (2.41000000000000001
      e+00)
END IF
      IF ((x>=8500).AND.(x<9000))THEN

```

```

qnoselow = (1.9036279069767425e-09)*(x-8500)**3 +
            (2.91200000000000019e-06)*(x-8500)**2 &
+ (3.7900930232558140e-03)*(x-8500) + (3.8149999999999999
            e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnoselow = (1.7897860465116284e-08)*(x-9000)**3 +
            (5.7674418604651116e-06)*(x-9000)**2 &
+ (8.1298139534883729e-03)*(x-9000) + (6.67600000000000002
            e+00)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnoselow = (1.9568930232558179e-08)*(x-9500)**3 +
            (3.2614232558139504e-05)*(x-9500)**2 &
+ (2.7320651162790699e-02)*(x-9500) + (1.44200000000000000
            e+01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnoselow = (-4.9592093023255567e-09)*(x-10000)**3 +
            (6.1967627906976737e-05)*(x-10000)**2 &
+ (7.4611581395348833e-02)*(x-10000) +
            (3.86800000000000000e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnoselow = (-4.9592093023257131e-09)*(x-11000)**3 +
            (4.70900000000000079e-05)*(x-11000)**2 &
+ (1.8366920930232564e-01)*(x-11000) +
            (1.70300000000000001e+02)

```



```

END IF
if ((x>=0).AND.(x<8000))THEN
qnosehigh = (6.1050000000000004e-04)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
qnosehigh = (2.9592713178294424e-09)*(x-8000)**3 +
(5.5309302325583957e-07)*(x-8000)**2 &
+ (4.4336356589147188e-03)*(x-8000) + (4.8840000000000003
e+00)
END IF
IF ((x>=8500).AND.(x<9000))THEN
qnosehigh = (2.9592713178294449e-09)*(x-8500)**3 +
(4.9919999999999973e-06)*(x-8500)**2 &
+ (7.2061821705426401e-03)*(x-8500) + (7.6090000000000000
e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnosehigh = (2.7067643410852716e-08)*(x-9000)**3 +
(9.4309069767442062e-06)*(x-9000)**2 &
+ (1.4417635658914721e-02)*(x-9000) + (1.2830000000000000
e+01)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnosehigh = (2.1498155038759680e-08)*(x-9500)**3 +
(5.0032372093023221e-05)*(x-9500)**2 &
+ (4.4149275193798465e-02)*(x-9500) + (2.5780000000000001
e+01)

```

```

END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (-8.8348682170542163e-09)*(x-10000)**3 +
    (8.2279604651162773e-05)*(x-10000)**2 &
+ (1.1030526356589145e-01)*(x-10000) +
    (6.304999999999997e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (-8.8348682170543933e-09)*(x-11000)**3 +
    (5.5775000000000068e-05)*(x-11000)**2 &
+ (2.4835986821705433e-01)*(x-11000) +
    (2.4680000000000001e+02)
END IF
  qnosel = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<8000))THEN
  qnoselow = (5.684999999999999e-05)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
  qnoselow = (5.8671038251366123e-10)*(x-8000)**3 +
    (-8.1665573770492560e-08)*(x-8000)**2 &
+ (5.1615519125683114e-04)*(x-8000) + (4.5479999999999998
    e-01)
END IF
IF ((x>=8500).AND.(x<9000))THEN
  qnoselow = (5.8671038251366113e-10)*(x-8500)**3 +
    (7.9840000000000006e-07)*(x-8500)**2 &

```

```

+ (8.7452240437158448e-04)*(x-8500) + (7.6580000000000004
e-01)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnoselow = (8.0872480874316866e-09)*(x-9000)**3 +
    (1.6784655737704964e-06)*(x-9000)**2 &
+ (2.1129551912568304e-03)*(x-9000) + (1.4760000000000000
e+00)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnoselow = (4.9298972677595492e-09)*(x-9500)**3 +
    (1.3809337704918042e-05)*(x-9500)**2 &
+ (9.8568568306010920e-03)*(x-9500) + (3.9630000000000001
e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (-8.7780109289614607e-10)*(x-10000)**3 +
    (2.1204183606557345e-05)*(x-10000)**2 &
+ (2.7363617486338795e-02)*(x-10000) +
    (1.2960000000000001e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (-4.5936174863396767e-10)*(x-11000)**3 +
    (1.8570780327868924e-05)*(x-11000)**2 &
+ (6.7138581420765051e-02)*(x-11000) +
    (6.064999999999999e+01)
END IF
if ((x>=0).AND.(x<8000))THEN

```

```

qnosehigh = (1.1882500000000000e-04)*(x) +
              (0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
    qnosehigh = (1.2172065573770535e-09)*(x-8000)**3 +
                 (-5.7060983606557971e-07)*(x-8000)**2 &
                 + (1.1138032786885263e-03)*(x-8000) + (9.5060000000000000
                 e-01)
END IF
IF ((x>=8500).AND.(x<9000))THEN
    qnosehigh = (1.2172065573770481e-09)*(x-8500)**3 +
                 (1.25520000000000024e-06)*(x-8500)**2 &
                 + (1.4560983606557367e-03)*(x-8500) + (1.5169999999999999
                 e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
    qnosehigh = (1.0773167213114740e-08)*(x-9000)**3 +
                 (3.0810098360655810e-06)*(x-9000)**2 &
                 + (3.6242032786885253e-03)*(x-9000) + (2.7109999999999999
                 e+00)
END IF
IF ((x>=9500).AND.(x<10000))THEN
    qnosehigh = (6.2981245901639163e-09)*(x-9500)**3 +
                 (1.9240760655737713e-05)*(x-9500)**2 &
                 + (1.4785088524590161e-02)*(x-9500) + (6.6399999999999997
                 e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN

```

```

      qnosehigh = (-2.2673901639344319e-09)*(x-10000)**3 +
      (2.8687947540983628e-05)*(x-10000)**2 &
+ (3.8749442622950811e-02)*(x-10000) +
      (1.9629999999999999e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
      qnosehigh = (-9.0894426229519492e-10)*(x-11000)**3 +
      (2.1885777049180420e-05)*(x-11000)**2 &
+ (8.9323167213114771e-02)*(x-11000) +
      (8.4799999999999997e+01)
END IF
      qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
      newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
      rndiff = 1.0-(0.500000-nose)/(0.500000-0.230000)
IF ((nose>0.2300000).AND.(nose<=0.50)) THEN
      IF ((x>=0).AND.(x<8000))THEN
      qnoselow = (6.10500000000000004e-04)*(x) +
      (0.0000000000000000e+00)
END IF
      IF ((x>=8000).AND.(x<8500))THEN
      qnoselow = (2.9592713178294424e-09)*(x-8000)**3 +
      (5.5309302325583957e-07)*(x-8000)**2 &
+ (4.4336356589147188e-03)*(x-8000) + (4.88400000000000003
      e+00)
END IF
      IF ((x>=8500).AND.(x<9000))THEN

```

```

qnoselow = (2.9592713178294449e-09)*(x-8500)**3 +
(4.9919999999999973e-06)*(x-8500)**2 &
+ (7.2061821705426401e-03)*(x-8500) + (7.6090000000000000
e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnoselow = (2.7067643410852716e-08)*(x-9000)**3 +
(9.4309069767442062e-06)*(x-9000)**2 &
+ (1.4417635658914721e-02)*(x-9000) + (1.2830000000000000
e+01)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnoselow = (2.1498155038759680e-08)*(x-9500)**3 +
(5.0032372093023221e-05)*(x-9500)**2 &
+ (4.4149275193798465e-02)*(x-9500) + (2.5780000000000001
e+01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnoselow = (-8.8348682170542163e-09)*(x-10000)**3 +
(8.2279604651162773e-05)*(x-10000)**2 &
+ (1.1030526356589145e-01)*(x-10000) +
(6.304999999999997e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnoselow = (-8.8348682170543933e-09)*(x-11000)**3 +
(5.57750000000000068e-05)*(x-11000)**2 &
+ (2.4835986821705433e-01)*(x-11000) +
(2.4680000000000001e+02)

```

```

END IF
if ((x>=0).AND.(x<8000))THEN
qnosehigh = (1.1852499999999999e-03)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
qnosehigh = (4.4727441860465187e-09)*(x-8000)**3 +
(1.6348837209302222e-06)*(x-8000)**2 &
+ (8.3003720930232595e-03)*(x-8000) + (9.4819999999999993
e+00)
END IF
IF ((x>=8500).AND.(x<9000))THEN
qnosehigh = (4.4727441860465071e-09)*(x-8500)**3 +
(8.34400000000000102e-06)*(x-8500)**2 &
+ (1.3289813953488371e-02)*(x-8500) + (1.4600000000000000
e+01)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnosehigh = (4.0180279069767443e-08)*(x-9000)**3 +
(1.5053116279069760e-05)*(x-9000)**2 &
+ (2.4988372093023261e-02)*(x-9000) + (2.3890000000000001
e+01)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnosehigh = (1.6086139534883982e-08)*(x-9500)**3 +
(7.5323534883720781e-05)*(x-9500)**2 &
+ (7.0176697674418601e-02)*(x-9500) + (4.5170000000000002
e+01)

```

```

END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (-1.2117581395348863e-08)*(x-10000)**3 +
    (9.9452744186046509e-05)*(x-10000)**2 &
+ (1.5756483720930237e-01)*(x-10000) +
    (1.0109999999999999e+02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (-1.2117581395348856e-08)*(x-11000)**3 +
    (6.31000000000000097e-05)*(x-11000)**2 &
+ (3.2011758139534879e-01)*(x-11000) +
    (3.4600000000000000e+02)
END IF
  qnosel = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<8000))THEN
  qnoselow = (1.1882500000000000e-04)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
  qnoselow = (1.2172065573770535e-09)*(x-8000)**3 +
    (-5.7060983606557971e-07)*(x-8000)**2 &
+ (1.1138032786885263e-03)*(x-8000) + (9.5060000000000000
    e-01)
END IF
IF ((x>=8500).AND.(x<9000))THEN
  qnoselow = (1.2172065573770481e-09)*(x-8500)**3 +
    (1.25520000000000024e-06)*(x-8500)**2 &

```



```

+ (1.4560983606557367e-03)*(x-8500) + (1.5169999999999999
e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnoselow = (1.0773167213114740e-08)*(x-9000)**3 +
    (3.0810098360655810e-06)*(x-9000)**2 &
+ (3.6242032786885253e-03)*(x-9000) + (2.7109999999999999
e+00)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnoselow = (6.2981245901639163e-09)*(x-9500)**3 +
    (1.9240760655737713e-05)*(x-9500)**2 &
+ (1.4785088524590161e-02)*(x-9500) + (6.6399999999999997
e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (-2.2673901639344319e-09)*(x-10000)**3 +
    (2.8687947540983628e-05)*(x-10000)**2 &
+ (3.8749442622950811e-02)*(x-10000) +
    (1.9629999999999999e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (-9.0894426229519492e-10)*(x-11000)**3 +
    (2.1885777049180420e-05)*(x-11000)**2 &
+ (8.9323167213114771e-02)*(x-11000) +
    (8.4799999999999997e+01)
END IF
if ((x>=0).AND.(x<8000))THEN

```

```

qnosehigh = (2.3387500000000001e-04)*(x) +
              (0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
    qnosehigh = (1.6420059612518657e-09)*(x-8000)**3 +
                 (-3.8100894187779747e-07)*(x-8000)**2 &
                 + (1.9000029806259323e-03)*(x-8000) + (1.8710000000000000
                 e+00)
END IF
IF ((x>=8500).AND.(x<9000))THEN
    qnosehigh = (1.6420059612518700e-09)*(x-8500)**3 +
                 (2.0819999999999972e-06)*(x-8500)**2 &
                 + (2.7504985096870340e-03)*(x-8500) + (2.9310000000000000
                 e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
    qnosehigh = (1.5597970193740697e-08)*(x-9000)**3 +
                 (4.5450089418777843e-06)*(x-9000)**2 &
                 + (6.0640029806259335e-03)*(x-9000) + (5.0320000000000000
                 e+00)
END IF
IF ((x>=9500).AND.(x<10000))THEN
    qnosehigh = (6.4861132637854139e-09)*(x-9500)**3 +
                 (2.7941964232488819e-05)*(x-9500)**2 &
                 + (2.2307489567809240e-02)*(x-9500) + (1.1150000000000000
                 e+01)
END IF
IF ((x>=10000).AND.(x<11000))THEN

```

```

      qnosehigh = (-4.0851728763040063e-09)*(x-10000)**3 +
      (3.7671134128166878e-05)*(x-10000)**2 &
+ (5.5114038748137119e-02)*(x-10000) +
      (3.01000000000000001e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
      qnosehigh = (-1.3164038748137286e-09)*(x-11000)**3 +
      (2.5415615499254889e-05)*(x-11000)**2 &
+ (1.1820078837555885e-01)*(x-11000) +
      (1.18800000000000000e+02)
END IF
      qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
      newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
      rndiff = 1.0-(1.000000-nose)/(1.000000-0.500000)
IF ((nose>0.5000000).AND.(nose<=1.00)) THEN
      IF ((x>=0).AND.(x<8000))THEN
      qnoselow = (1.1852499999999999e-03)*(x) +
      (0.00000000000000000e+00)
END IF
      IF ((x>=8000).AND.(x<8500))THEN
      qnoselow = (4.4727441860465187e-09)*(x-8000)**3 +
      (1.6348837209302222e-06)*(x-8000)**2 &
+ (8.3003720930232595e-03)*(x-8000) + (9.4819999999999993
      e+00)
END IF
      IF ((x>=8500).AND.(x<9000))THEN

```

```

      qnoselow = (4.4727441860465071e-09)*(x-8500)**3 +
      (8.34400000000000102e-06)*(x-8500)**2 &
+ (1.3289813953488371e-02)*(x-8500) + (1.4600000000000000
      e+01)
END IF
IF ((x>=9000) .AND. (x<9500)) THEN
      qnoselow = (4.0180279069767443e-08)*(x-9000)**3 +
      (1.5053116279069760e-05)*(x-9000)**2 &
+ (2.4988372093023261e-02)*(x-9000) + (2.38900000000000001
      e+01)
END IF
IF ((x>=9500) .AND. (x<10000)) THEN
      qnoselow = (1.6086139534883982e-08)*(x-9500)**3 +
      (7.5323534883720781e-05)*(x-9500)**2 &
+ (7.0176697674418601e-02)*(x-9500) + (4.51700000000000002
      e+01)
END IF
IF ((x>=10000) .AND. (x<11000)) THEN
      qnoselow = (-1.2117581395348863e-08)*(x-10000)**3 +
      (9.9452744186046509e-05)*(x-10000)**2 &
+ (1.5756483720930237e-01)*(x-10000) +
      (1.0109999999999999e+02)
END IF
IF ((x>=11000) .AND. (x<12000)) THEN
      qnoselow = (-1.2117581395348856e-08)*(x-11000)**3 +
      (6.31000000000000097e-05)*(x-11000)**2 &
+ (3.2011758139534879e-01)*(x-11000) +
      (3.4600000000000000e+02)

```

```

END IF
if ((x>=0).AND.(x<8000))THEN
qnosehigh = (2.0800000000000003e-03)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
qnosehigh = (6.3516279069767094e-09)*(x-8000)**3 +
(4.3725581395349370e-06)*(x-8000)**2 &
+ (1.3565813953488352e-02)*(x-8000) + (1.6640000000000001
e+01)
END IF
IF ((x>=8500).AND.(x<9000))THEN
qnosehigh = (6.3516279069767739e-09)*(x-8500)**3 +
(1.3899999999999972e-05)*(x-8500)**2 &
+ (2.2702093023255820e-02)*(x-8500) + (2.530999999999999
e+01)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnosehigh = (5.0321860465116334e-08)*(x-9000)**3 +
(2.3427441860465105e-05)*(x-9000)**2 &
+ (4.1365813953488373e-02)*(x-9000) + (4.0930000000000000
e+01)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnosehigh = (8.3609302325577930e-09)*(x-9500)**3 +
(9.8910232558139684e-05)*(x-9500)**2 &
+ (1.0253465116279073e-01)*(x-9500) + (7.3760000000000005
e+01)

```

```

END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (-1.4967209302325453e-08)*(x-10000)**3 +
    (1.1145162790697666e-04)*(x-10000)**2 &
+ (2.0771558139534876e-01)*(x-10000) +
    (1.50800000000000001e+02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (-1.4967209302325701e-08)*(x-11000)**3 +
    (6.6549999999999937e-05)*(x-11000)**2 &
+ (3.8571720930232573e-01)*(x-11000) +
    (4.55000000000000000e+02)
END IF
  qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<8000))THEN
  qnoselow = (2.33875000000000001e-04)*(x) +
    (0.00000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
  qnoselow = (1.6420059612518657e-09)*(x-8000)**3 +
    (-3.8100894187779747e-07)*(x-8000)**2 &
+ (1.9000029806259323e-03)*(x-8000) + (1.8710000000000000
    e+00)
END IF
IF ((x>=8500).AND.(x<9000))THEN
  qnoselow = (1.6420059612518700e-09)*(x-8500)**3 +
    (2.0819999999999972e-06)*(x-8500)**2 &

```

```

+ (2.7504985096870340e-03)*(x-8500) + (2.9310000000000000
e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnoselow = (1.5597970193740697e-08)*(x-9000)**3 +
    (4.5450089418777843e-06)*(x-9000)**2 &
+ (6.0640029806259335e-03)*(x-9000) + (5.0320000000000000
e+00)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnoselow = (6.4861132637854139e-09)*(x-9500)**3 +
    (2.7941964232488819e-05)*(x-9500)**2 &
+ (2.2307489567809240e-02)*(x-9500) + (1.1150000000000000
e+01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (-4.0851728763040063e-09)*(x-10000)**3 +
    (3.7671134128166878e-05)*(x-10000)**2 &
+ (5.5114038748137119e-02)*(x-10000) +
    (3.01000000000000001e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (-1.3164038748137286e-09)*(x-11000)**3 +
    (2.5415615499254889e-05)*(x-11000)**2 &
+ (1.1820078837555885e-01)*(x-11000) +
    (1.18800000000000000e+02)
END IF
if ((x>=0).AND.(x<8000))THEN

```

```

qnosehigh = (4.3525000000000004e-04)*(x) +
            (0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
    qnosehigh = (2.4438638847491100e-09)*(x-8000)**3 +
                (-3.5579582712366623e-07)*(x-8000)**2 &
                + (3.3189319423745545e-03)*(x-8000) + (3.4820000000000002
                e+00)
END IF
IF ((x>=8500).AND.(x<9000))THEN
    qnosehigh = (2.4438638847491129e-09)*(x-8500)**3 +
                (3.31000000000000013e-06)*(x-8500)**2 &
                + (4.7960340288127207e-03)*(x-8500) + (5.3579999999999997
                e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
    qnosehigh = (2.1100680576254380e-08)*(x-9000)**3 +
                (6.9757958271236969e-06)*(x-9000)**2 &
                + (9.9389319423745567e-03)*(x-9000) + (8.8889999999999993
                e+00)
END IF
IF ((x>=9500).AND.(x<10000))THEN
    qnosehigh = (5.2254138102334666e-09)*(x-9500)**3 +
                (3.8626816691505201e-05)*(x-9500)**2 &
                + (3.2740238201689038e-02)*(x-9500) + (1.8239999999999998
                e+01)
END IF
IF ((x>=10000).AND.(x<11000))THEN

```



```

      qnosehigh = (-6.2710526577247938e-09)*(x-10000)**3 +
      (4.6464937406855455e-05)*(x-10000)**2 &
+ (7.5286115250869340e-02)*(x-10000) +
      (4.4920000000000002e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
      qnosehigh = (-1.3546115250870630e-09)*(x-11000)**3 +
      (2.7651779433681219e-05)*(x-11000)**2 &
+ (1.4940283209140587e-01)*(x-11000) +
      (1.6040000000000001e+02)
END IF
      qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
      newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
      rndiff = 1.0-(2.300000-nose)/(2.300000-1.000000)
IF ((nose>1.0000000).AND.(nose<=2.30)) THEN
      IF ((x>=0).AND.(x<8000))THEN
      qnoselow = (2.0800000000000003e-03)*(x) +
      (0.0000000000000000e+00)
END IF
      IF ((x>=8000).AND.(x<8500))THEN
      qnoselow = (6.3516279069767094e-09)*(x-8000)**3 +
      (4.3725581395349370e-06)*(x-8000)**2 &
+ (1.3565813953488352e-02)*(x-8000) + (1.6640000000000001
      e+01)
END IF
      IF ((x>=8500).AND.(x<9000))THEN

```

```

qnoselow = (6.3516279069767739e-09)*(x-8500)**3 +
            (1.3899999999999972e-05)*(x-8500)**2 &
+ (2.2702093023255820e-02)*(x-8500) + (2.5309999999999999
            e+01)
END IF
IF ((x>=9000) .AND. (x<9500)) THEN
    qnoselow = (5.0321860465116334e-08)*(x-9000)**3 +
                (2.3427441860465105e-05)*(x-9000)**2 &
+ (4.1365813953488373e-02)*(x-9000) + (4.0930000000000000
            e+01)
END IF
IF ((x>=9500) .AND. (x<10000)) THEN
    qnoselow = (8.3609302325577930e-09)*(x-9500)**3 +
                (9.8910232558139684e-05)*(x-9500)**2 &
+ (1.0253465116279073e-01)*(x-9500) + (7.3760000000000005
            e+01)
END IF
IF ((x>=10000) .AND. (x<11000)) THEN
    qnoselow = (-1.4967209302325453e-08)*(x-10000)**3 +
                (1.1145162790697666e-04)*(x-10000)**2 &
+ (2.0771558139534876e-01)*(x-10000) +
            (1.50800000000000001e+02)
END IF
IF ((x>=11000) .AND. (x<12000)) THEN
    qnoselow = (-1.4967209302325701e-08)*(x-11000)**3 +
                (6.6549999999999937e-05)*(x-11000)**2 &
+ (3.8571720930232573e-01)*(x-11000) +
            (4.5500000000000000e+02)

```

```

END IF
if ((x>=0).AND.(x<8000))THEN
qnosehigh = (3.8425000000000000e-03)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
qnosehigh = (1.0606976744185979e-08)*(x-8000)**3 +
(6.0095348837210108e-06)*(x-8000)**2 &
+ (2.6503488372093001e-02)*(x-8000) + (3.0739999999999998
e+01)
END IF
IF ((x>=8500).AND.(x<9000))THEN
qnosehigh = (1.0606976744186148e-08)*(x-8500)**3 +
(2.1919999999999926e-05)*(x-8500)**2 &
+ (4.0468255813953496e-02)*(x-8500) + (4.6820000000000000
e+01)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnosehigh = (5.9285116279069717e-08)*(x-9000)**3 +
(3.7830465116279091e-05)*(x-9000)**2 &
+ (7.0343488372093033e-02)*(x-9000) + (7.3859999999999999
e+01)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnosehigh = (-1.1267441860465235e-08)*(x-9500)**3 +
(1.2675813953488379e-04)*(x-9500)**2 &
+ (1.5263779069767441e-01)*(x-9500) + (1.2590000000000001
e+02)

```

```

END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (-1.7102325581395190e-08)*(x-10000)**3 +
    (1.0985697674418594e-04)*(x-10000)**2 &
+ (2.7094534883720928e-01)*(x-10000) +
    (2.3250000000000000e+02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (-1.7102325581395793e-08)*(x-11000)**3 +
    (5.85500000000000156e-05)*(x-11000)**2 &
+ (4.3935232558139559e-01)*(x-11000) +
    (5.9620000000000005e+02)
END IF
  qnosel = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<8000))THEN
  qnoselow = (4.3525000000000004e-04)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
  qnoselow = (2.4438638847491100e-09)*(x-8000)**3 +
    (-3.5579582712366623e-07)*(x-8000)**2 &
+ (3.3189319423745545e-03)*(x-8000) + (3.4820000000000002
    e+00)
END IF
IF ((x>=8500).AND.(x<9000))THEN
  qnoselow = (2.4438638847491129e-09)*(x-8500)**3 +
    (3.31000000000000013e-06)*(x-8500)**2 &

```

```

+ (4.7960340288127207e-03)*(x-8500) + (5.357999999999997
e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnoselow = (2.1100680576254380e-08)*(x-9000)**3 +
    (6.9757958271236969e-06)*(x-9000)**2 &
+ (9.9389319423745567e-03)*(x-9000) + (8.888999999999993
e+00)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnoselow = (5.2254138102334666e-09)*(x-9500)**3 +
    (3.8626816691505201e-05)*(x-9500)**2 &
+ (3.2740238201689038e-02)*(x-9500) + (1.8239999999999998
e+01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (-6.2710526577247938e-09)*(x-10000)**3 +
    (4.6464937406855455e-05)*(x-10000)**2 &
+ (7.5286115250869340e-02)*(x-10000) +
    (4.4920000000000002e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (-1.3546115250870630e-09)*(x-11000)**3 +
    (2.7651779433681219e-05)*(x-11000)**2 &
+ (1.4940283209140587e-01)*(x-11000) +
    (1.6040000000000001e+02)
END IF
if ((x>=0).AND.(x<8000))THEN

```

```

qnosehigh = (8.9000000000000006e-04)*(x) +
            (0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
    qnosehigh = (4.1891306507700197e-09)*(x-8000)**3 +
                (-4.2369597615502490e-07)*(x-8000)**2 &
                + (6.6045653253850069e-03)*(x-8000) + (7.1200000000000001
                e+00)
END IF
IF ((x>=8500).AND.(x<9000))THEN
    qnosehigh = (4.1891306507699709e-09)*(x-8500)**3 +
                (5.86000000000000150e-06)*(x-8500)**2 &
                + (9.3227173373074974e-03)*(x-8500) + (1.0840000000000000
                e+01)
END IF
IF ((x>=9000).AND.(x<9500))THEN
    qnosehigh = (2.7454346746149986e-08)*(x-9000)**3 +
                (1.2143695976155019e-05)*(x-9000)**2 &
                + (1.8324565325384991e-02)*(x-9000) + (1.7489999999999998
                e+01)
END IF
IF ((x>=9500).AND.(x<10000))THEN
    qnosehigh = (1.1934823646299829e-09)*(x-9500)**3 +
                (5.3325216095379995e-05)*(x-9500)**2 &
                + (5.1059021361152500e-02)*(x-9500) + (3.3119999999999997
                e+01)
END IF
IF ((x>=10000).AND.(x<11000))THEN

```

```

      qnosehigh = (-9.3247888723297924e-09)*(x-10000)**3 +
      (5.5115439642324794e-05)*(x-10000)**2 &
+ (1.0527934923000498e-01)*(x-10000) +
      (7.2129999999999995e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
      qnosehigh = (-1.0769349230005941e-09)*(x-11000)**3 +
      (2.7141073025335416e-05)*(x-11000)**2 &
+ (1.8753586189766519e-01)*(x-11000) +
      (2.2319999999999999e+02)
END IF
      qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
      newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
      rndiff = 1.0-(5.000000-nose)/(5.000000-2.300000)
IF ((nose>2.3000000).AND.(nose<=5.00)) THEN
      IF ((x>=0).AND.(x<8000))THEN
      qnoselow = (3.8425000000000000e-03)*(x) +
      (0.0000000000000000e+00)
END IF
      IF ((x>=8000).AND.(x<8500))THEN
      qnoselow = (1.0606976744185979e-08)*(x-8000)**3 +
      (6.0095348837210108e-06)*(x-8000)**2 &
+ (2.6503488372093001e-02)*(x-8000) + (3.0739999999999998
      e+01)
END IF
      IF ((x>=8500).AND.(x<9000))THEN

```

```

qnoselow = (1.0606976744186148e-08)*(x-8500)**3 +
(2.1919999999999926e-05)*(x-8500)**2 &
+ (4.0468255813953496e-02)*(x-8500) + (4.6820000000000000
e+01)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnoselow = (5.9285116279069717e-08)*(x-9000)**3 +
(3.7830465116279091e-05)*(x-9000)**2 &
+ (7.0343488372093033e-02)*(x-9000) + (7.385999999999999
e+01)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnoselow = (-1.1267441860465235e-08)*(x-9500)**3 +
(1.2675813953488379e-04)*(x-9500)**2 &
+ (1.5263779069767441e-01)*(x-9500) + (1.2590000000000001
e+02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnoselow = (-1.7102325581395190e-08)*(x-10000)**3 +
(1.0985697674418594e-04)*(x-10000)**2 &
+ (2.7094534883720928e-01)*(x-10000) +
(2.3250000000000000e+02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnoselow = (-1.7102325581395793e-08)*(x-11000)**3 +
(5.85500000000000156e-05)*(x-11000)**2 &
+ (4.3935232558139559e-01)*(x-11000) +
(5.9620000000000005e+02)

```



```

END IF
if ((x>=0).AND.(x<8000))THEN
qnosehigh = (6.4437499999999998e-03)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
qnosehigh = (1.3293023255813998e-08)*(x-8000)**3 +
(1.1960465116279029e-05)*(x-8000)**2 &
+ (4.1596511627906993e-02)*(x-8000) + (5.154999999999997
e+01)
END IF
IF ((x>=8500).AND.(x<9000))THEN
qnosehigh = (1.3293023255813761e-08)*(x-8500)**3 +
(3.19000000000000105e-05)*(x-8500)**2 &
+ (6.3526744186046522e-02)*(x-8500) + (7.7000000000000000
e+01)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnosehigh = (5.6334883720930315e-08)*(x-9000)**3 +
(5.1839534883720896e-05)*(x-9000)**2 &
+ (1.0539651162790695e-01)*(x-9000) + (1.1840000000000001
e+02)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnosehigh = (-3.8632558139535136e-08)*(x-9500)**3 +
(1.3634186046511653e-04)*(x-9500)**2 &
+ (1.9948720930232558e-01)*(x-9500) + (1.910999999999999
e+02)

```

```

END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (-4.0476744186044828e-09)*(x-10000)**3 +
    (7.8393023255813650e-05)*(x-10000)**2 &
+ (3.0685465116279076e-01)*(x-10000) +
    (3.20100000000000002e+02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (-4.0476744186048153e-09)*(x-11000)**3 +
    (6.62500000000000255e-05)*(x-11000)**2 &
+ (4.5149767441860461e-01)*(x-11000) +
    (7.0129999999999995e+02)
END IF
  qnosel = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<8000))THEN
  qnoselow = (8.9000000000000006e-04)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
  qnoselow = (4.1891306507700197e-09)*(x-8000)**3 +
    (-4.2369597615502490e-07)*(x-8000)**2 &
+ (6.6045653253850069e-03)*(x-8000) + (7.1200000000000001
    e+00)
END IF
IF ((x>=8500).AND.(x<9000))THEN
  qnoselow = (4.1891306507699709e-09)*(x-8500)**3 +
    (5.86000000000000150e-06)*(x-8500)**2 &

```

```

+ (9.3227173373074974e-03)*(x-8500) + (1.0840000000000000
e+01)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnoselow = (2.7454346746149986e-08)*(x-9000)**3 +
    (1.2143695976155019e-05)*(x-9000)**2 &
+ (1.8324565325384991e-02)*(x-9000) + (1.7489999999999998
e+01)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnoselow = (1.1934823646299829e-09)*(x-9500)**3 +
    (5.3325216095379995e-05)*(x-9500)**2 &
+ (5.1059021361152500e-02)*(x-9500) + (3.3119999999999997
e+01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (-9.3247888723297924e-09)*(x-10000)**3 +
    (5.5115439642324794e-05)*(x-10000)**2 &
+ (1.0527934923000498e-01)*(x-10000) +
    (7.2129999999999995e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (-1.0769349230005941e-09)*(x-11000)**3 +
    (2.7141073025335416e-05)*(x-11000)**2 &
+ (1.8753586189766519e-01)*(x-11000) +
    (2.2319999999999999e+02)
END IF
if ((x>=0).AND.(x<8000))THEN

```

```

qnosehigh = (1.6462499999999999e-03)*(x) +
            (0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
    qnosehigh = (6.3360357675111756e-09)*(x-8000)**3 +
                (9.5946348733236820e-08)*(x-8000)**2 &
                + (1.1908017883755591e-02)*(x-8000) + (1.3170000000000000
                e+01)
END IF
IF ((x>=8500).AND.(x<9000))THEN
    qnosehigh = (6.3360357675111839e-09)*(x-8500)**3 +
                (9.5999999999999894e-06)*(x-8500)**2 &
                + (1.6755991058122210e-02)*(x-8500) + (1.9940000000000001
                e+01)
END IF
IF ((x>=9000).AND.(x<9500))THEN
    qnosehigh = (3.1039821162444155e-08)*(x-9000)**3 +
                (1.9104053651266744e-05)*(x-9000)**2 &
                + (3.1108017883755588e-02)*(x-9000) + (3.1510000000000002
                e+01)
END IF
IF ((x>=9500).AND.(x<10000))THEN
    qnosehigh = (-5.4553204172876946e-09)*(x-9500)**3 +
                (6.5663785394932980e-05)*(x-9500)**2 &
                + (7.3491937406855445e-02)*(x-9500) + (5.571999999999999
                e+01)
END IF
IF ((x>=10000).AND.(x<11000))THEN

```

```

      qnosehigh = (-1.2045037257824157e-08)*(x-10000)**3 +
      (5.7480804769001492e-05)*(x-10000)**2 &
+ (1.3506423248882266e-01)*(x-10000) +
      (1.0820000000000000e+02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
      qnosehigh = (2.4635767511177204e-09)*(x-11000)**3 +
      (2.1345692995529101e-05)*(x-11000)**2 &
+ (2.1389073025335317e-01)*(x-11000) +
      (2.8869999999999999e+02)
END IF
      qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
      newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
      rndiff = 1.0-(10.000000-nose)/(10.000000-5.000000)
IF ((nose>5.0000000).AND.(nose<=10.00)) THEN
      IF ((x>=0).AND.(x<8000))THEN
      qnoselow = (6.4437499999999998e-03)*(x) +
      (0.0000000000000000e+00)
END IF
      IF ((x>=8000).AND.(x<8500))THEN
      qnoselow = (1.3293023255813998e-08)*(x-8000)**3 +
      (1.1960465116279029e-05)*(x-8000)**2 &
+ (4.1596511627906993e-02)*(x-8000) + (5.1549999999999997
      e+01)
END IF
      IF ((x>=8500).AND.(x<9000))THEN

```

```

qnoselow = (1.3293023255813761e-08)*(x-8500)**3 +
(3.19000000000000105e-05)*(x-8500)**2 &
+ (6.3526744186046522e-02)*(x-8500) + (7.7000000000000000
e+01)
END IF
IF ((x>=9000) .AND. (x<9500)) THEN
qnoselow = (5.6334883720930315e-08)*(x-9000)**3 +
(5.1839534883720896e-05)*(x-9000)**2 &
+ (1.0539651162790695e-01)*(x-9000) + (1.1840000000000001
e+02)
END IF
IF ((x>=9500) .AND. (x<10000)) THEN
qnoselow = (-3.8632558139535136e-08)*(x-9500)**3 +
(1.3634186046511653e-04)*(x-9500)**2 &
+ (1.9948720930232558e-01)*(x-9500) + (1.9109999999999999
e+02)
END IF
IF ((x>=10000) .AND. (x<11000)) THEN
qnoselow = (-4.0476744186044828e-09)*(x-10000)**3 +
(7.8393023255813650e-05)*(x-10000)**2 &
+ (3.0685465116279076e-01)*(x-10000) +
(3.20100000000000002e+02)
END IF
IF ((x>=11000) .AND. (x<12000)) THEN
qnoselow = (-4.0476744186048153e-09)*(x-11000)**3 +
(6.62500000000000255e-05)*(x-11000)**2 &
+ (4.5149767441860461e-01)*(x-11000) +
(7.0129999999999995e+02)

```

```

END IF
if ((x>=0).AND.(x<8000))THEN
qnosehigh = (9.6062499999999985e-03)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
qnosehigh = (2.1894573643410741e-08)*(x-8000)**3 +
(1.4581395348838290e-06)*(x-8000)**2 &
+ (6.4697286821705405e-02)*(x-8000) + (7.684999999999994
e+01)
END IF
IF ((x>=8500).AND.(x<9000))THEN
qnosehigh = (2.1894573643410907e-08)*(x-8500)**3 +
(3.4300000000000000e-05)*(x-8500)**2 &
+ (8.2576356589147290e-02)*(x-8500) + (1.1230000000000000
e+02)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnosehigh = (2.4527131782945846e-08)*(x-9000)**3 +
(6.7141860465116110e-05)*(x-9000)**2 &
+ (1.3329728682170547e-01)*(x-9000) + (1.6490000000000001
e+02)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnosehigh = (-5.6031007751936579e-09)*(x-9500)**3 +
(1.0393255813953495e-04)*(x-9500)**2 &
+ (2.1883449612403097e-01)*(x-9500) + (2.5140000000000001
e+02)

```

```

END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (-2.9292635658914823e-08)*(x-10000)**3 +
    (9.5527906976744124e-05)*(x-10000)**2 &
+ (3.1856472868217067e-01)*(x-10000) +
    (3.8610000000000002e+02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (-2.9292635658914320e-08)*(x-11000)**3 +
    (7.6499999999998742e-06)*(x-11000)**2 &
+ (4.2174263565891446e-01)*(x-11000) +
    (7.7089999999999998e+02)
END IF
  qnosel = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<8000))THEN
  qnoselow = (1.6462499999999999e-03)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
  qnoselow = (6.3360357675111756e-09)*(x-8000)**3 +
    (9.5946348733236820e-08)*(x-8000)**2 &
+ (1.1908017883755591e-02)*(x-8000) + (1.3170000000000000
    e+01)
END IF
IF ((x>=8500).AND.(x<9000))THEN
  qnoselow = (6.3360357675111839e-09)*(x-8500)**3 +
    (9.5999999999999894e-06)*(x-8500)**2 &

```



```

+ (1.6755991058122210e-02)*(x-8500) + (1.9940000000000001
    e+01)
END IF
IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (3.1039821162444155e-08)*(x-9000)**3 +
        (1.9104053651266744e-05)*(x-9000)**2 &
+ (3.1108017883755588e-02)*(x-9000) + (3.1510000000000002
    e+01)
END IF
IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (-5.4553204172876946e-09)*(x-9500)**3 +
        (6.5663785394932980e-05)*(x-9500)**2 &
+ (7.3491937406855445e-02)*(x-9500) + (5.571999999999999
    e+01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
    qnoselow = (-1.2045037257824157e-08)*(x-10000)**3 +
        (5.7480804769001492e-05)*(x-10000)**2 &
+ (1.3506423248882266e-01)*(x-10000) +
        (1.0820000000000000e+02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
    qnoselow = (2.4635767511177204e-09)*(x-11000)**3 +
        (2.1345692995529101e-05)*(x-11000)**2 &
+ (2.1389073025335317e-01)*(x-11000) +
        (2.8869999999999999e+02)
END IF
if ((x>=0).AND.(x<8000))THEN

```

```

qnosehigh = (2.7425000000000001e-03)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
  qnosehigh = (9.4904918032787956e-09)*(x-8000)**3 +
(-1.4757377049181707e-06)*(x-8000)**2 &
+ (2.0205245901639381e-02)*(x-8000) + (2.1940000000000001
e+01)
END IF
IF ((x>=8500).AND.(x<9000))THEN
  qnosehigh = (9.4904918032787311e-09)*(x-8500)**3 +
(1.27600000000000003e-05)*(x-8500)**2 &
+ (2.5847377049180307e-02)*(x-8500) + (3.2859999999999999
e+01)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (2.7907540983606425e-08)*(x-9000)**3 +
(2.6995737704918116e-05)*(x-9000)**2 &
+ (4.5725245901639358e-02)*(x-9000) + (5.0159999999999997
e+01)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (-2.4006557377049708e-09)*(x-9500)**3 +
(6.8857049180327855e-05)*(x-9500)**2 &
+ (9.3651639344262294e-02)*(x-9500) + (8.32600000000000005
e+01)
END IF
IF ((x>=10000).AND.(x<11000))THEN

```

```

      qnosehigh = (-1.5564262295081924e-08)*(x-10000)**3 +
      (6.5256065573770480e-05)*(x-10000)**2 &
+ (1.6070819672131142e-01)*(x-10000) +
      (1.4700000000000000e+02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
      qnosehigh = (-8.9081967213114380e-10)*(x-11000)**3 +
      (1.8563278688524587e-05)*(x-11000)**2 &
+ (2.4452754098360660e-01)*(x-11000) +
      (3.5739999999999998e+02)
END IF
      qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
      newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if

END IF !density if
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!NEW ALTITUDE RANGE!!!!!!!!!!!!!!!!!!!!!!!!

IF ((rho<=0.0072579000).AND.(rho>=0.0029948000)) THEN
      rhodiff = (0.0072579000-rho)/(0.0072579000-0.0029948000)
      rndiff = 1.0-(0.100000-nose)/(0.100000-0.049999)
      IF ((nose>0.0499990).AND.(nose<=0.10)) THEN
            IF ((x>=0).AND.(x<8000))THEN
                  qnoselow = (3.1162499999999999e-05)*(x) +
                  (0.0000000000000000e+00)
            END IF
            IF ((x>=8000).AND.(x<8500))THEN

```

```

qnoselow = (3.2539085941380988e-10)*(x-8000)**3 +
            (9.5713710879285063e-08)*(x-8000)**2 &
+ (2.4119542970690500e-04)*(x-8000) + (2.4929999999999999
            e-01)
END IF
IF ((x>=8500).AND.(x<9000))THEN
    qnoselow = (3.2539085941380937e-10)*(x-8500)**3 +
                (5.83800000000000013e-07)*(x-8500)**2 &
+ (5.8095228514654746e-04)*(x-8500) + (4.34500000000000000
            e-01)
END IF
IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (6.2002457029309555e-09)*(x-9000)**3 +
                (1.0718862891207128e-06)*(x-9000)**2 &
+ (1.4087954297069046e-03)*(x-9000) + (9.1159999999999997
            e-01)
END IF
IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (4.5320263288623894e-09)*(x-9500)**3 +
                (1.0372254843517140e-05)*(x-9500)**2 &
+ (7.1308659960258340e-03)*(x-9500) + (2.6589999999999998
            e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN
    qnoselow = (-2.9643492300048195e-10)*(x-10000)**3 +
                (1.7170294336810713e-05)*(x-10000)**2 &
+ (2.0902140586189766e-02)*(x-10000) +
            (9.38400000000000003e+00)

```

```

END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (5.5859413810108040e-12)*(x-11000)**3 +
    (1.6280989567809240e-05)*(x-11000)**2 &
+ (5.4353424490809744e-02)*(x-11000) +
    (4.7159999999999997e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (5.5859413810650142e-12)*(x-12000)**3 +
    (1.6297747391952238e-05)*(x-12000)**2 &
+ (8.6932161450571255e-02)*(x-12000) +
    (1.1780000000000000e+02)
END IF
if ((x>=0).AND.(x<8000))THEN
  qnosehigh = (5.684999999999999e-05)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
  qnosehigh = (5.8671038251366123e-10)*(x-8000)**3 +
    (-8.1665573770492560e-08)*(x-8000)**2 &
+ (5.1615519125683114e-04)*(x-8000) + (4.5479999999999998
    e-01)
END IF
IF ((x>=8500).AND.(x<9000))THEN
  qnosehigh = (5.8671038251366113e-10)*(x-8500)**3 +
    (7.9840000000000006e-07)*(x-8500)**2 &
+ (8.7452240437158448e-04)*(x-8500) + (7.6580000000000004
    e-01)

```

```

END IF
IF ((x>=9000).AND.(x<9500))THEN
    qnosehigh = (8.0872480874316866e-09)*(x-9000)**3 +
        (1.6784655737704964e-06)*(x-9000)**2 &
+ (2.1129551912568304e-03)*(x-9000) + (1.4760000000000000
    e+00)
END IF
IF ((x>=9500).AND.(x<10000))THEN
    qnosehigh = (4.9298972677595492e-09)*(x-9500)**3 +
        (1.3809337704918042e-05)*(x-9500)**2 &
+ (9.8568568306010920e-03)*(x-9500) + (3.9630000000000001
    e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN
    qnosehigh = (-8.7780109289614607e-10)*(x-10000)**3 +
        (2.1204183606557345e-05)*(x-10000)**2 &
+ (2.7363617486338795e-02)*(x-10000) +
        (1.29600000000000001e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
    qnosehigh = (-4.5936174863396767e-10)*(x-11000)**3 +
        (1.8570780327868924e-05)*(x-11000)**2 &
+ (6.7138581420765051e-02)*(x-11000) +
        (6.0649999999999999e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
    qnosehigh = (-4.5936174863382027e-10)*(x-12000)**3 +
        (1.7192695081967136e-05)*(x-12000)**2 &

```

```

+ (1.0290205683060100e-01)*(x-12000) +
    (1.45900000000000001e+02)
END IF

qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<8000))THEN
qnoselow = (6.353749999999999e-06)*(x) +
    (0.0000000000000000e+00)
END IF

IF ((x>=8000).AND.(x<8500))THEN
    qnoselow = (1.6041177347242870e-10)*(x-8000)**3 +
        (-8.6477660208643074e-08)*(x-8000)**2 &
+ (8.7635886736214360e-05)*(x-8000) + (5.0830000000000000
    e-02)
END IF

IF ((x>=8500).AND.(x<9000))THEN
    qnoselow = (1.6041177347242826e-10)*(x-8500)**3 +
        (1.54140000000000029e-07)*(x-8500)**2 &
+ (1.2146705663189280e-04)*(x-8500) + (9.3079999999999996
    e-02)
END IF

IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (2.4436211326378547e-09)*(x-9000)**3 +
        (3.9475766020864386e-07)*(x-9000)**2 &
+ (3.9591588673621429e-04)*(x-9000) + (2.1240000000000001
    e-01)
END IF

IF ((x>=9500).AND.(x<10000))THEN

```

```

qnoselow = (1.5580636959761644e-09)*(x-9500)**3 +
(4.0601893591654176e-06)*(x-9500)**2 &
+ (2.6233893964232491e-03)*(x-9500) + (8.1450000000000000
e-01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnoselow = (8.4588569299548476e-11)*(x-10000)**3 +
(6.3972849031296627e-06)*(x-10000)**2 &
+ (7.8521265275707901e-03)*(x-10000) +
(3.3359999999999999e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnoselow = (1.2848734724291853e-10)*(x-11000)**3 +
(6.6510506110283199e-06)*(x-11000)**2 &
+ (2.0900462041728761e-02)*(x-11000) +
(1.76700000000000002e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
qnoselow = (1.2848734724292742e-10)*(x-12000)**3 +
(7.0365126527570645e-06)*(x-12000)**2 &
+ (3.4588025305514157e-02)*(x-12000) +
(4.53500000000000001e+01)
END IF
if ((x>=0).AND.(x<8000))THEN
qnosehigh = (1.1611250000000000e-05)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN

```



```

qnosehigh = (2.4094228514654682e-10)*(x-8000)**3 +
              (-1.2583342771982022e-07)*(x-8000)**2 &
+ (1.4350114257327340e-04)*(x-8000) + (9.2890000000000000
              e-02)
END IF
IF ((x>=8500).AND.(x<9000))THEN
  qnosehigh = (2.4094228514654801e-10)*(x-8500)**3 +
              (2.3557999999999925e-07)*(x-8500)**2 &
+ (1.9837442871336331e-04)*(x-8500) + (1.6330000000000000
              e-01)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (3.3833685742672636e-09)*(x-9000)**3 +
              (5.9699342771982167e-07)*(x-9000)**2 &
+ (6.1466114257327357e-04)*(x-9000) + (3.5149999999999998
              e-01)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (1.5711834177843861e-09)*(x-9500)**3 +
              (5.6720462891207214e-06)*(x-9500)**2 &
+ (3.7491810009935428e-03)*(x-9500) + (1.2310000000000001
              e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (-1.9843626924986609e-10)*(x-10000)**3 +
              (8.0288214157973108e-06)*(x-10000)**2 &
+ (1.0599614853452554e-02)*(x-10000) +
              (4.7199999999999998e+00)

```

```

END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (2.4538514654746398e-11)*(x-11000)**3 +
    (7.4335126080476787e-06)*(x-11000)**2 &
+ (2.6061948877297576e-02)*(x-11000) +
    (2.3149999999999999e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (2.4538514654750634e-11)*(x-12000)**3 +
    (7.5071281520119145e-06)*(x-12000)**2 &
+ (4.1002589637357173e-02)*(x-12000) +
    (5.66700000000000002e+01)
END IF
  qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
  newqnr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
rndiff = 1.0-(0.230000-nose)/(0.230000-0.100000)
IF ((nose>0.1000000).AND.(nose<=0.23)) THEN
  IF ((x>=0).AND.(x<8000))THEN
    qnoselow = (5.684999999999999e-05)*(x) +
      (0.0000000000000000e+00)
  END IF
  IF ((x>=8000).AND.(x<8500))THEN
    qnoselow = (5.8671038251366123e-10)*(x-8000)**3 +
      (-8.1665573770492560e-08)*(x-8000)**2 &
+ (5.1615519125683114e-04)*(x-8000) + (4.5479999999999998
      e-01)
  END IF

```

```

IF ((x>=8500).AND.(x<9000))THEN
  qnoselow = (5.8671038251366113e-10)*(x-8500)**3 +
    (7.9840000000000006e-07)*(x-8500)**2 &
+ (8.7452240437158448e-04)*(x-8500) + (7.6580000000000004
  e-01)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnoselow = (8.0872480874316866e-09)*(x-9000)**3 +
    (1.6784655737704964e-06)*(x-9000)**2 &
+ (2.1129551912568304e-03)*(x-9000) + (1.4760000000000000
  e+00)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnoselow = (4.9298972677595492e-09)*(x-9500)**3 +
    (1.3809337704918042e-05)*(x-9500)**2 &
+ (9.8568568306010920e-03)*(x-9500) + (3.9630000000000001
  e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (-8.7780109289614607e-10)*(x-10000)**3 +
    (2.1204183606557345e-05)*(x-10000)**2 &
+ (2.7363617486338795e-02)*(x-10000) +
    (1.2960000000000001e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (-4.5936174863396767e-10)*(x-11000)**3 +
    (1.8570780327868924e-05)*(x-11000)**2 &

```

```

+ (6.7138581420765051e-02)*(x-11000) +
    (6.064999999999999e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
    qnoselow = (-4.5936174863382027e-10)*(x-12000)**3 +
        (1.7192695081967136e-05)*(x-12000)**2 &
+ (1.0290205683060100e-01)*(x-12000) +
    (1.45900000000000001e+02)
END IF
if ((x>=0).AND.(x<8000))THEN
    qnosehigh = (1.1882500000000000e-04)*(x) +
        (0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
    qnosehigh = (1.2172065573770535e-09)*(x-8000)**3 +
        (-5.7060983606557971e-07)*(x-8000)**2 &
+ (1.1138032786885263e-03)*(x-8000) + (9.5060000000000000
    e-01)
END IF
IF ((x>=8500).AND.(x<9000))THEN
    qnosehigh = (1.2172065573770481e-09)*(x-8500)**3 +
        (1.255200000000000024e-06)*(x-8500)**2 &
+ (1.4560983606557367e-03)*(x-8500) + (1.5169999999999999
    e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
    qnosehigh = (1.0773167213114740e-08)*(x-9000)**3 +
        (3.0810098360655810e-06)*(x-9000)**2 &

```

```

+ (3.6242032786885253e-03)*(x-9000) + (2.7109999999999999
e+00)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (6.2981245901639163e-09)*(x-9500)**3 +
    (1.9240760655737713e-05)*(x-9500)**2 &
+ (1.4785088524590161e-02)*(x-9500) + (6.6399999999999997
e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (-2.2673901639344319e-09)*(x-10000)**3 +
    (2.8687947540983628e-05)*(x-10000)**2 &
+ (3.8749442622950811e-02)*(x-10000) +
    (1.9629999999999999e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (-9.0894426229519492e-10)*(x-11000)**3 +
    (2.1885777049180420e-05)*(x-11000)**2 &
+ (8.9323167213114771e-02)*(x-11000) +
    (8.4799999999999997e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (-9.0894426229499329e-10)*(x-12000)**3 +
    (1.9158944262294974e-05)*(x-12000)**2 &
+ (1.3036788852459003e-01)*(x-12000) +
    (1.9509999999999999e+02)
END IF
qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)

```

```

if ((x>=0).AND.(x<8000))THEN
qnoselow = (1.1611250000000000e-05)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
qnoselow = (2.4094228514654682e-10)*(x-8000)**3 +
(-1.2583342771982022e-07)*(x-8000)**2 &
+ (1.4350114257327340e-04)*(x-8000) + (9.2890000000000000
e-02)
END IF
IF ((x>=8500).AND.(x<9000))THEN
qnoselow = (2.4094228514654801e-10)*(x-8500)**3 +
(2.3557999999999925e-07)*(x-8500)**2 &
+ (1.9837442871336331e-04)*(x-8500) + (1.6330000000000000
e-01)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnoselow = (3.3833685742672636e-09)*(x-9000)**3 +
(5.9699342771982167e-07)*(x-9000)**2 &
+ (6.1466114257327357e-04)*(x-9000) + (3.5149999999999998
e-01)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnoselow = (1.5711834177843861e-09)*(x-9500)**3 +
(5.6720462891207214e-06)*(x-9500)**2 &
+ (3.7491810009935428e-03)*(x-9500) + (1.2310000000000001
e+00)
END IF

```

```

IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (-1.9843626924986609e-10)*(x-10000)**3 +
    (8.0288214157973108e-06)*(x-10000)**2 &
  + (1.0599614853452554e-02)*(x-10000) +
    (4.7199999999999998e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (2.4538514654746398e-11)*(x-11000)**3 +
    (7.4335126080476787e-06)*(x-11000)**2 &
  + (2.6061948877297576e-02)*(x-11000) +
    (2.3149999999999999e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (2.4538514654750634e-11)*(x-12000)**3 +
    (7.5071281520119145e-06)*(x-12000)**2 &
  + (4.1002589637357173e-02)*(x-12000) +
    (5.66700000000000002e+01)
END IF
if ((x>=0).AND.(x<8000))THEN
  qnosehigh = (2.4324999999999998e-05)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
  qnosehigh = (3.8542583209140470e-10)*(x-8000)**3 +
    (-2.0273874813710691e-07)*(x-8000)**2 &
  + (2.6941291604570225e-04)*(x-8000) + (1.9460000000000000
    e-01)
END IF

```

```

IF ((x>=8500).AND.(x<9000))THEN
  qnosehigh = (3.8542583209140357e-10)*(x-8500)**3 +
    (3.75400000000000078e-07)*(x-8500)**2 &
  + (3.5574354197714887e-04)*(x-8500) + (3.2679999999999998
    e-01)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (4.8544708395429733e-09)*(x-9000)**3 +
    (9.5353874813710818e-07)*(x-9000)**2 &
  + (1.0202129160457023e-03)*(x-9000) + (6.4670000000000005
    e-01)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (1.2630908097367017e-09)*(x-9500)**3 +
    (8.2352450074515675e-06)*(x-9500)**2 &
  + (5.6146047938400402e-03)*(x-9500) + (2.0019999999999998
    e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (-6.0304913065077303e-10)*(x-10000)**3 +
    (1.0129881222056640e-05)*(x-10000)**2 &
  + (1.4797167908594134e-02)*(x-10000) +
    (7.0259999999999998e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (-5.8516790859421312e-11)*(x-11000)**3 +
    (8.3207338301043230e-06)*(x-11000)**2 &

```



```

+ (3.3247782960755097e-02)*(x-11000) +
    (3.1350000000000001e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
    qnosehigh = (-5.8516790859393356e-11)*(x-12000)**3 +
        (8.1451834575260523e-06)*(x-12000)**2 &
+ (4.9713700248385478e-02)*(x-12000) +
    (7.2859999999999999e+01)
END IF
    qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
    newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
    rndiff = 1.0-(0.500000-nose)/(0.500000-0.230000)
IF ((nose>0.2300000).AND.(nose<=0.50)) THEN
    IF ((x>=0).AND.(x<8000))THEN
        qnoselow = (1.1882500000000000e-04)*(x) +
            (0.0000000000000000e+00)
    END IF
    IF ((x>=8000).AND.(x<8500))THEN
        qnoselow = (1.2172065573770535e-09)*(x-8000)**3 +
            (-5.7060983606557971e-07)*(x-8000)**2 &
+ (1.1138032786885263e-03)*(x-8000) + (9.5060000000000000
            e-01)
    END IF
    IF ((x>=8500).AND.(x<9000))THEN
        qnoselow = (1.2172065573770481e-09)*(x-8500)**3 +
            (1.25520000000000024e-06)*(x-8500)**2 &

```

```

+ (1.4560983606557367e-03)*(x-8500) + (1.5169999999999999
e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnoselow = (1.0773167213114740e-08)*(x-9000)**3 +
    (3.0810098360655810e-06)*(x-9000)**2 &
+ (3.6242032786885253e-03)*(x-9000) + (2.7109999999999999
e+00)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnoselow = (6.2981245901639163e-09)*(x-9500)**3 +
    (1.9240760655737713e-05)*(x-9500)**2 &
+ (1.4785088524590161e-02)*(x-9500) + (6.6399999999999997
e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (-2.2673901639344319e-09)*(x-10000)**3 +
    (2.8687947540983628e-05)*(x-10000)**2 &
+ (3.8749442622950811e-02)*(x-10000) +
    (1.9629999999999999e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (-9.0894426229519492e-10)*(x-11000)**3 +
    (2.1885777049180420e-05)*(x-11000)**2 &
+ (8.9323167213114771e-02)*(x-11000) +
    (8.4799999999999997e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN

```

```

      qnoselow =  (-9.0894426229499329e-10)*(x-12000)**3 +
                  (1.9158944262294974e-05)*(x-12000)**2 &
+  (1.3036788852459003e-01)*(x-12000) +
                  (1.9509999999999999e+02)
END IF
if ((x>=0).AND.(x<8000))THEN
qnosehigh = (2.3387500000000001e-04)*(x) +
              (0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
      qnosehigh =  (1.6420059612518657e-09)*(x-8000)**3 +
                   (-3.8100894187779747e-07)*(x-8000)**2 &
+  (1.9000029806259323e-03)*(x-8000) + (1.8710000000000000
      e+00)
END IF
IF ((x>=8500).AND.(x<9000))THEN
      qnosehigh =  (1.6420059612518700e-09)*(x-8500)**3 +
                   (2.0819999999999972e-06)*(x-8500)**2 &
+  (2.7504985096870340e-03)*(x-8500) + (2.9310000000000000
      e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
      qnosehigh =  (1.5597970193740697e-08)*(x-9000)**3 +
                   (4.5450089418777843e-06)*(x-9000)**2 &
+  (6.0640029806259335e-03)*(x-9000) + (5.0320000000000000
      e+00)
END IF
IF ((x>=9500).AND.(x<10000))THEN

```

```

    qnosehigh = (6.4861132637854139e-09)*(x-9500)**3 +
        (2.7941964232488819e-05)*(x-9500)**2 &
+ (2.2307489567809240e-02)*(x-9500) + (1.1150000000000000
    e+01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
    qnosehigh = (-4.0851728763040063e-09)*(x-10000)**3 +
        (3.7671134128166878e-05)*(x-10000)**2 &
+ (5.5114038748137119e-02)*(x-10000) +
        (3.01000000000000001e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
    qnosehigh = (-1.3164038748137286e-09)*(x-11000)**3 +
        (2.5415615499254889e-05)*(x-11000)**2 &
+ (1.1820078837555885e-01)*(x-11000) +
        (1.18800000000000000e+02)
END IF
IF ((x>=12000).AND.(x<14000))THEN
    qnosehigh = (-1.3164038748137302e-09)*(x-12000)**3 +
        (2.1466403874813747e-05)*(x-12000)**2 &
+ (1.6508280774962744e-01)*(x-12000) +
        (2.61100000000000002e+02)
END IF
    qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<8000))THEN
    qnoselow = (2.4324999999999998e-05)*(x) +
        (0.00000000000000000e+00)
END IF

```

```

IF ((x>=8000).AND.(x<8500))THEN
  qnoselow = (3.8542583209140470e-10)*(x-8000)**3 +
    (-2.0273874813710691e-07)*(x-8000)**2 &
  + (2.6941291604570225e-04)*(x-8000) + (1.9460000000000000
    e-01)
END IF

IF ((x>=8500).AND.(x<9000))THEN
  qnoselow = (3.8542583209140357e-10)*(x-8500)**3 +
    (3.75400000000000078e-07)*(x-8500)**2 &
  + (3.5574354197714887e-04)*(x-8500) + (3.2679999999999998
    e-01)
END IF

IF ((x>=9000).AND.(x<9500))THEN
  qnoselow = (4.8544708395429733e-09)*(x-9000)**3 +
    (9.5353874813710818e-07)*(x-9000)**2 &
  + (1.0202129160457023e-03)*(x-9000) + (6.4670000000000005
    e-01)
END IF

IF ((x>=9500).AND.(x<10000))THEN
  qnoselow = (1.2630908097367017e-09)*(x-9500)**3 +
    (8.2352450074515675e-06)*(x-9500)**2 &
  + (5.6146047938400402e-03)*(x-9500) + (2.0019999999999998
    e+00)
END IF

IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (-6.0304913065077303e-10)*(x-10000)**3 +
    (1.0129881222056640e-05)*(x-10000)**2 &

```

```

+ (1.4797167908594134e-02)*(x-10000) +
(7.0259999999999998e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (-5.8516790859421312e-11)*(x-11000)**3 +
(8.3207338301043230e-06)*(x-11000)**2 &
+ (3.3247782960755097e-02)*(x-11000) +
(3.13500000000000001e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (-5.8516790859393356e-11)*(x-12000)**3 +
(8.1451834575260523e-06)*(x-12000)**2 &
+ (4.9713700248385478e-02)*(x-12000) +
(7.2859999999999999e+01)
END IF
if ((x>=0).AND.(x<8000))THEN
  qnosehigh = (4.8974999999999998e-05)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
  qnosehigh = (7.4204649776452812e-10)*(x-8000)**3 +
(-5.4186974664679266e-07)*(x-8000)**2 &
+ (5.6102324888226442e-04)*(x-8000) + (3.9179999999999998
e-01)
END IF
IF ((x>=8500).AND.(x<9000))THEN
  qnosehigh = (7.4204649776453153e-10)*(x-8500)**3 +
(5.71199999999999823e-07)*(x-8500)**2 &

```

```

+ (5.7568837555886784e-04)*(x-8500) + (6.2960000000000005
    e-01)
END IF
IF ((x>=9000).AND.(x<9500))THEN
    qnosehigh = (6.2897675111773582e-09)*(x-9000)**3 +
        (1.6842697466467921e-06)*(x-9000)**2 &
+ (1.7034232488822648e-03)*(x-9000) + (1.1530000000000000
    e+00)
END IF
IF ((x>=9500).AND.(x<10000))THEN
    qnosehigh = (1.0860834575260694e-09)*(x-9500)**3 +
        (1.1118921013412814e-05)*(x-9500)**2 &
+ (8.1050186289120755e-03)*(x-9500) + (3.2120000000000002
    e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN
    qnosehigh = (-9.5654843517137231e-10)*(x-10000)**3 +
        (1.2748046199701927e-05)*(x-10000)**2 &
+ (2.0038502235469442e-02)*(x-10000) +
        (1.0180000000000000e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
    qnosehigh = (-2.6335022354699040e-10)*(x-11000)**3 +
        (9.8784008941878188e-06)*(x-11000)**2 &
+ (4.2664949329359179e-02)*(x-11000) +
        (4.2009999999999998e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN

```

```

      qnosehigh = (-2.6335022354692014e-10)*(x-12000)**3 +
      (9.0883502235469136e-06)*(x-12000)**2 &
+ (6.1631700447093846e-02)*(x-12000) +
      (9.4290000000000006e+01)
END IF

      qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
      newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if

      rndiff = 1.0-(1.000000-nose)/(1.000000-0.500000)
IF ((nose>0.5000000).AND.(nose<=1.00)) THEN
  IF ((x>=0).AND.(x<8000))THEN
    qnoselow = (2.3387500000000001e-04)*(x) +
      (0.0000000000000000e+00)
  END IF
  IF ((x>=8000).AND.(x<8500))THEN
    qnoselow = (1.6420059612518657e-09)*(x-8000)**3 +
      (-3.8100894187779747e-07)*(x-8000)**2 &
+ (1.9000029806259323e-03)*(x-8000) + (1.8710000000000000
      e+00)
  END IF
  IF ((x>=8500).AND.(x<9000))THEN
    qnoselow = (1.6420059612518700e-09)*(x-8500)**3 +
      (2.0819999999999972e-06)*(x-8500)**2 &
+ (2.7504985096870340e-03)*(x-8500) + (2.9310000000000000
      e+00)
  END IF
  IF ((x>=9000).AND.(x<9500))THEN

```



```

qnoselow = (1.5597970193740697e-08)*(x-9000)**3 +
            (4.5450089418777843e-06)*(x-9000)**2 &
+ (6.0640029806259335e-03)*(x-9000) + (5.0320000000000000
            e+00)
END IF
IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (6.4861132637854139e-09)*(x-9500)**3 +
                (2.7941964232488819e-05)*(x-9500)**2 &
+ (2.2307489567809240e-02)*(x-9500) + (1.1150000000000000
            e+01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
    qnoselow = (-4.0851728763040063e-09)*(x-10000)**3 +
                (3.7671134128166878e-05)*(x-10000)**2 &
+ (5.5114038748137119e-02)*(x-10000) +
            (3.01000000000000001e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
    qnoselow = (-1.3164038748137286e-09)*(x-11000)**3 +
                (2.5415615499254889e-05)*(x-11000)**2 &
+ (1.1820078837555885e-01)*(x-11000) +
            (1.18800000000000000e+02)
END IF
IF ((x>=12000).AND.(x<14000))THEN
    qnoselow = (-1.3164038748137302e-09)*(x-12000)**3 +
                (2.1466403874813747e-05)*(x-12000)**2 &
+ (1.6508280774962744e-01)*(x-12000) +
            (2.61100000000000002e+02)

```

```

END IF
if ((x>=0).AND.(x<8000))THEN
qnosehigh = (4.3525000000000004e-04)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
qnosehigh = (2.4438638847491100e-09)*(x-8000)**3 +
(-3.5579582712366623e-07)*(x-8000)**2 &
+ (3.3189319423745545e-03)*(x-8000) + (3.4820000000000002
e+00)
END IF
IF ((x>=8500).AND.(x<9000))THEN
qnosehigh = (2.4438638847491129e-09)*(x-8500)**3 +
(3.31000000000000013e-06)*(x-8500)**2 &
+ (4.7960340288127207e-03)*(x-8500) + (5.357999999999997
e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnosehigh = (2.1100680576254380e-08)*(x-9000)**3 +
(6.9757958271236969e-06)*(x-9000)**2 &
+ (9.9389319423745567e-03)*(x-9000) + (8.888999999999993
e+00)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnosehigh = (5.2254138102334666e-09)*(x-9500)**3 +
(3.8626816691505201e-05)*(x-9500)**2 &
+ (3.2740238201689038e-02)*(x-9500) + (1.823999999999998
e+01)

```

```

END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (-6.2710526577247938e-09)*(x-10000)**3 +
    (4.6464937406855455e-05)*(x-10000)**2 &
+ (7.5286115250869340e-02)*(x-10000) +
    (4.4920000000000002e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (-1.3546115250870630e-09)*(x-11000)**3 +
    (2.7651779433681219e-05)*(x-11000)**2 &
+ (1.4940283209140587e-01)*(x-11000) +
    (1.6040000000000001e+02)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (-1.3546115250868767e-09)*(x-12000)**3 +
    (2.3587944858420186e-05)*(x-12000)**2 &
+ (2.0064255638350711e-01)*(x-12000) +
    (3.3610000000000002e+02)
END IF
  qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
  if ((x>=0).AND.(x<8000))THEN
    qnoselow = (4.897499999999998e-05)*(x) +
      (0.0000000000000000e+00)
  END IF
  IF ((x>=8000).AND.(x<8500))THEN
    qnoselow = (7.4204649776452812e-10)*(x-8000)**3 +
      (-5.4186974664679266e-07)*(x-8000)**2 &

```

```

+ (5.6102324888226442e-04)*(x-8000) + (3.9179999999999998
    e-01)
END IF
IF ((x>=8500).AND.(x<9000))THEN
    qnoselow = (7.4204649776453153e-10)*(x-8500)**3 +
        (5.71199999999999823e-07)*(x-8500)**2 &
+ (5.7568837555886784e-04)*(x-8500) + (6.2960000000000005
    e-01)
END IF
IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (6.2897675111773582e-09)*(x-9000)**3 +
        (1.6842697466467921e-06)*(x-9000)**2 &
+ (1.7034232488822648e-03)*(x-9000) + (1.1530000000000000
    e+00)
END IF
IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (1.0860834575260694e-09)*(x-9500)**3 +
        (1.1118921013412814e-05)*(x-9500)**2 &
+ (8.1050186289120755e-03)*(x-9500) + (3.2120000000000002
    e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN
    qnoselow = (-9.5654843517137231e-10)*(x-10000)**3 +
        (1.2748046199701927e-05)*(x-10000)**2 &
+ (2.0038502235469442e-02)*(x-10000) +
        (1.0180000000000000e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN

```

```

      qnoselow =  (-2.6335022354699040e-10)*(x-11000)**3 +
                  (9.8784008941878188e-06)*(x-11000)**2 &
+   (4.2664949329359179e-02)*(x-11000) +
      (4.2009999999999998e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
      qnoselow =  (-2.6335022354692014e-10)*(x-12000)**3 +
                  (9.0883502235469136e-06)*(x-12000)**2 &
+   (6.1631700447093846e-02)*(x-12000) +
      (9.42900000000000006e+01)
END IF
if ((x>=0).AND.(x<8000))THEN
      qnosehigh = (9.06375000000000001e-05)*(x) +
                  (0.00000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
      qnosehigh =  (1.0391702930948753e-09)*(x-8000)**3 +
                  (-6.3655543964231260e-07)*(x-8000)**2 &
+   (8.8028514654743745e-04)*(x-8000) + (7.2509999999999997
      e-01)
END IF
IF ((x>=8500).AND.(x<9000))THEN
      qnosehigh =  (1.0391702930948777e-09)*(x-8500)**3 +
                  (9.22199999999999940e-07)*(x-8500)**2 &
+   (1.0231074267262811e-03)*(x-8500) + (1.1359999999999999
      e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN

```

```

qnosehigh = (8.1713485345255934e-09)*(x-9000)**3 +
(2.4809554396423249e-06)*(x-9000)**2 &
+ (2.7246851465474388e-03)*(x-9000) + (2.0080000000000000
e+00)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnosehigh = (7.7143556880279347e-10)*(x-9500)**3 +
(1.4737978241430685e-05)*(x-9500)**2 &
+ (1.1334151987083959e-02)*(x-9500) + (5.0119999999999996
e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnosehigh = (-1.4358384997516119e-09)*(x-10000)**3 +
(1.5895131594634874e-05)*(x-10000)**2 &
+ (2.6650706905116738e-02)*(x-10000) +
(1.44600000000000001e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnosehigh = (-4.9107069051169983e-10)*(x-11000)**3 +
(1.1587616095380046e-05)*(x-11000)**2 &
+ (5.4133454595131650e-02)*(x-11000) +
(5.55700000000000000e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
qnosehigh = (-4.9107069051164731e-10)*(x-12000)**3 +
(1.0114404023844967e-05)*(x-12000)**2 &
+ (7.5835474714356643e-02)*(x-12000) +
(1.20800000000000000e+02)

```

```

END IF

  qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
  newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
rndiff = 1.0-(2.300000-nose)/(2.300000-1.000000)
IF ((nose>1.0000000).AND.(nose<=2.30)) THEN
  IF ((x>=0).AND.(x<8000))THEN
    qnoselow = (4.3525000000000004e-04)*(x) +
      (0.0000000000000000e+00)
  END IF
  IF ((x>=8000).AND.(x<8500))THEN
    qnoselow = (2.4438638847491100e-09)*(x-8000)**3 +
      (-3.5579582712366623e-07)*(x-8000)**2 &
      + (3.3189319423745545e-03)*(x-8000) + (3.4820000000000002
        e+00)
  END IF
  IF ((x>=8500).AND.(x<9000))THEN
    qnoselow = (2.4438638847491129e-09)*(x-8500)**3 +
      (3.31000000000000013e-06)*(x-8500)**2 &
      + (4.7960340288127207e-03)*(x-8500) + (5.357999999999997
        e+00)
  END IF
  IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (2.1100680576254380e-08)*(x-9000)**3 +
      (6.9757958271236969e-06)*(x-9000)**2 &
      + (9.9389319423745567e-03)*(x-9000) + (8.888999999999993
        e+00)
  END IF

```

```

IF ((x>=9500).AND.(x<10000))THEN
  qnoselow = (5.2254138102334666e-09)*(x-9500)**3 +
    (3.8626816691505201e-05)*(x-9500)**2 &
+ (3.2740238201689038e-02)*(x-9500) + (1.8239999999999998
    e+01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (-6.2710526577247938e-09)*(x-10000)**3 +
    (4.6464937406855455e-05)*(x-10000)**2 &
+ (7.5286115250869340e-02)*(x-10000) +
    (4.4920000000000002e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (-1.3546115250870630e-09)*(x-11000)**3 +
    (2.7651779433681219e-05)*(x-11000)**2 &
+ (1.4940283209140587e-01)*(x-11000) +
    (1.6040000000000001e+02)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (-1.3546115250868767e-09)*(x-12000)**3 +
    (2.3587944858420186e-05)*(x-12000)**2 &
+ (2.0064255638350711e-01)*(x-12000) +
    (3.3610000000000002e+02)
END IF
if ((x>=0).AND.(x<8000))THEN
  qnosehigh = (8.9000000000000006e-04)*(x) +
    (0.0000000000000000e+00)
END IF

```



```

IF ((x>=8000).AND.(x<8500))THEN
  qnosehigh = (4.1891306507700197e-09)*(x-8000)**3 +
    (-4.2369597615502490e-07)*(x-8000)**2 &
  + (6.6045653253850069e-03)*(x-8000) + (7.1200000000000001
    e+00)
END IF
IF ((x>=8500).AND.(x<9000))THEN
  qnosehigh = (4.1891306507699709e-09)*(x-8500)**3 +
    (5.86000000000000150e-06)*(x-8500)**2 &
  + (9.3227173373074974e-03)*(x-8500) + (1.0840000000000000
    e+01)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (2.7454346746149986e-08)*(x-9000)**3 +
    (1.2143695976155019e-05)*(x-9000)**2 &
  + (1.8324565325384991e-02)*(x-9000) + (1.7489999999999998
    e+01)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (1.1934823646299829e-09)*(x-9500)**3 +
    (5.3325216095379995e-05)*(x-9500)**2 &
  + (5.1059021361152500e-02)*(x-9500) + (3.3119999999999997
    e+01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (-9.3247888723297924e-09)*(x-10000)**3 +
    (5.5115439642324794e-05)*(x-10000)**2 &

```

```

+ (1.0527934923000498e-01)*(x-10000) +
(7.2129999999999995e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (-1.0769349230005941e-09)*(x-11000)**3 +
(2.7141073025335416e-05)*(x-11000)**2 &
+ (1.8753586189766519e-01)*(x-11000) +
(2.2319999999999999e+02)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (-1.0769349230004824e-09)*(x-12000)**3 +
(2.3910268256333853e-05)*(x-12000)**2 &
+ (2.3858720317933424e-01)*(x-12000) +
(4.36800000000000001e+02)
END IF
  qnosel = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<8000))THEN
  qnoselow = (9.06375000000000001e-05)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
  qnoselow = (1.0391702930948753e-09)*(x-8000)**3 +
(-6.3655543964231260e-07)*(x-8000)**2 &
+ (8.8028514654743745e-04)*(x-8000) + (7.2509999999999997
e-01)
END IF
IF ((x>=8500).AND.(x<9000))THEN

```

```

qnoselow = (1.0391702930948777e-09)*(x-8500)**3 +
(9.2219999999999940e-07)*(x-8500)**2 &
+ (1.0231074267262811e-03)*(x-8500) + (1.1359999999999999
e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnoselow = (8.1713485345255934e-09)*(x-9000)**3 +
(2.4809554396423249e-06)*(x-9000)**2 &
+ (2.7246851465474388e-03)*(x-9000) + (2.0080000000000000
e+00)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnoselow = (7.7143556880279347e-10)*(x-9500)**3 +
(1.4737978241430685e-05)*(x-9500)**2 &
+ (1.1334151987083959e-02)*(x-9500) + (5.0119999999999996
e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnoselow = (-1.4358384997516119e-09)*(x-10000)**3 +
(1.5895131594634874e-05)*(x-10000)**2 &
+ (2.6650706905116738e-02)*(x-10000) +
(1.44600000000000001e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnoselow = (-4.9107069051169983e-10)*(x-11000)**3 +
(1.1587616095380046e-05)*(x-11000)**2 &
+ (5.4133454595131650e-02)*(x-11000) +
(5.5570000000000000e+01)

```

```

END IF
IF ((x>=12000).AND.(x<14000))THEN
    qnoselow = (-4.9107069051164731e-10)*(x-12000)**3 +
        (1.0114404023844967e-05)*(x-12000)**2 &
+ (7.5835474714356643e-02)*(x-12000) +
        (1.2080000000000000e+02)
END IF
if ((x>=0).AND.(x<8000))THEN
    qnosehigh = (1.9374999999999999e-04)*(x) +
        (0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
    qnosehigh = (1.6502533532041672e-09)*(x-8000)**3 +
        (-8.7938002980625326e-07)*(x-8000)**2 &
+ (1.6871266766020846e-03)*(x-8000) + (1.5500000000000000
        e+00)
END IF
IF ((x>=8500).AND.(x<9000))THEN
    qnosehigh = (1.6502533532041805e-09)*(x-8500)**3 +
        (1.5959999999999967e-06)*(x-8500)**2 &
+ (2.0454366616989568e-03)*(x-8500) + (2.3799999999999999
        e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
    qnosehigh = (1.1084733233979133e-08)*(x-9000)**3 +
        (4.0713800298062593e-06)*(x-9000)**2 &
+ (4.8791266766020887e-03)*(x-9000) + (4.0080000000000000
        e+00)

```

```

END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (1.8813710879254615e-11)*(x-9500)**3 +
    (2.0698479880774977e-05)*(x-9500)**2 &
+ (1.7264056631892698e-02)*(x-9500) + (8.8510000000000009
  e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (-2.8733472429209877e-09)*(x-10000)**3 +
    (2.0726700447093872e-05)*(x-10000)**2 &
+ (3.7976646795827115e-02)*(x-10000) +
    (2.2660000000000000e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (3.9333532041725724e-10)*(x-11000)**3 +
    (1.2106658718330872e-05)*(x-11000)**2 &
+ (7.0810005961251893e-02)*(x-11000) +
    (7.8489999999999995e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (3.9333532041730212e-10)*(x-12000)**3 +
    (1.3286664679582685e-05)*(x-12000)**2 &
+ (9.6203329359165410e-02)*(x-12000) +
    (1.6180000000000001e+02)
END IF
  qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
  newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if

```

```

rndiff = 1.0-(5.000000-nose)/(5.000000-2.300000)
IF ((nose>2.300000).AND.(nose<=5.00)) THEN
  IF ((x>=0).AND.(x<8000))THEN
    qnoselow = (8.90000000000000006e-04)*(x) +
      (0.00000000000000000e+00)
  END IF
  IF ((x>=8000).AND.(x<8500))THEN
    qnoselow = (4.1891306507700197e-09)*(x-8000)**3 +
      (-4.2369597615502490e-07)*(x-8000)**2 &
      + (6.6045653253850069e-03)*(x-8000) + (7.12000000000000001
        e+00)
  END IF
  IF ((x>=8500).AND.(x<9000))THEN
    qnoselow = (4.1891306507699709e-09)*(x-8500)**3 +
      (5.860000000000000150e-06)*(x-8500)**2 &
      + (9.3227173373074974e-03)*(x-8500) + (1.0840000000000000
        e+01)
  END IF
  IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (2.7454346746149986e-08)*(x-9000)**3 +
      (1.2143695976155019e-05)*(x-9000)**2 &
      + (1.8324565325384991e-02)*(x-9000) + (1.7489999999999998
        e+01)
  END IF
  IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (1.1934823646299829e-09)*(x-9500)**3 +
      (5.3325216095379995e-05)*(x-9500)**2 &

```

```

+ (5.1059021361152500e-02)*(x-9500) + (3.311999999999997
e+01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (-9.3247888723297924e-09)*(x-10000)**3 +
    (5.5115439642324794e-05)*(x-10000)**2 &
+ (1.0527934923000498e-01)*(x-10000) +
    (7.2129999999999995e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (-1.0769349230005941e-09)*(x-11000)**3 +
    (2.7141073025335416e-05)*(x-11000)**2 &
+ (1.8753586189766519e-01)*(x-11000) +
    (2.2319999999999999e+02)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (-1.0769349230004824e-09)*(x-12000)**3 +
    (2.3910268256333853e-05)*(x-12000)**2 &
+ (2.3858720317933424e-01)*(x-12000) +
    (4.36800000000000001e+02)
END IF
if ((x>=0).AND.(x<8000))THEN
  qnosehigh = (1.6462499999999999e-03)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
  qnosehigh = (6.3360357675111756e-09)*(x-8000)**3 +
    (9.5946348733236820e-08)*(x-8000)**2 &

```

```

+ (1.1908017883755591e-02)*(x-8000) + (1.3170000000000000
e+01)
END IF
IF ((x>=8500).AND.(x<9000))THEN
  qnosehigh = (6.3360357675111839e-09)*(x-8500)**3 +
    (9.5999999999999894e-06)*(x-8500)**2 &
+ (1.6755991058122210e-02)*(x-8500) + (1.9940000000000001
e+01)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (3.1039821162444155e-08)*(x-9000)**3 +
    (1.9104053651266744e-05)*(x-9000)**2 &
+ (3.1108017883755588e-02)*(x-9000) + (3.1510000000000002
e+01)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (-5.4553204172876946e-09)*(x-9500)**3 +
    (6.5663785394932980e-05)*(x-9500)**2 &
+ (7.3491937406855445e-02)*(x-9500) + (5.571999999999999
e+01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (-1.2045037257824157e-08)*(x-10000)**3 +
    (5.7480804769001492e-05)*(x-10000)**2 &
+ (1.3506423248882266e-01)*(x-10000) +
    (1.0820000000000000e+02)
END IF
IF ((x>=11000).AND.(x<12000))THEN

```



```

      qnosehigh = (2.4635767511177204e-09)*(x-11000)**3 +
        (2.1345692995529101e-05)*(x-11000)**2 &
+ (2.1389073025335317e-01)*(x-11000) +
        (2.8869999999999999e+02)
END IF
IF ((x>=12000).AND.(x<14000))THEN
      qnosehigh = (2.4635767511177610e-09)*(x-12000)**3 +
        (2.8736423248882228e-05)*(x-12000)**2 &
+ (2.6397284649776454e-01)*(x-12000) +
        (5.2639999999999998e+02)
END IF
      qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<8000))THEN
      qnoselow = (1.9374999999999999e-04)*(x) +
        (0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
      qnoselow = (1.6502533532041672e-09)*(x-8000)**3 +
        (-8.7938002980625326e-07)*(x-8000)**2 &
+ (1.6871266766020846e-03)*(x-8000) + (1.5500000000000000
        e+00)
END IF
IF ((x>=8500).AND.(x<9000))THEN
      qnoselow = (1.6502533532041805e-09)*(x-8500)**3 +
        (1.5959999999999967e-06)*(x-8500)**2 &
+ (2.0454366616989568e-03)*(x-8500) + (2.3799999999999999
        e+00)
END IF

```

```

IF ((x>=9000).AND.(x<9500))THEN
  qnoselow = (1.1084733233979133e-08)*(x-9000)**3 +
    (4.0713800298062593e-06)*(x-9000)**2 &
  + (4.8791266766020887e-03)*(x-9000) + (4.0080000000000000
    e+00)
END IF

IF ((x>=9500).AND.(x<10000))THEN
  qnoselow = (1.8813710879254615e-11)*(x-9500)**3 +
    (2.0698479880774977e-05)*(x-9500)**2 &
  + (1.7264056631892698e-02)*(x-9500) + (8.8510000000000009
    e+00)
END IF

IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (-2.8733472429209877e-09)*(x-10000)**3 +
    (2.0726700447093872e-05)*(x-10000)**2 &
  + (3.7976646795827115e-02)*(x-10000) +
    (2.2660000000000000e+01)
END IF

IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (3.9333532041725724e-10)*(x-11000)**3 +
    (1.2106658718330872e-05)*(x-11000)**2 &
  + (7.0810005961251893e-02)*(x-11000) +
    (7.8489999999999995e+01)
END IF

IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (3.9333532041730212e-10)*(x-12000)**3 +
    (1.3286664679582685e-05)*(x-12000)**2 &

```

```

+ (9.6203329359165410e-02)*(x-12000) +
  (1.61800000000000001e+02)
END IF
if ((x>=0).AND.(x<8000))THEN
qnosehigh = (3.8624999999999998e-04)*(x) +
  (0.00000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
  qnosehigh = (2.4307352210630859e-09)*(x-8000)**3 +
    (-9.5610283159462924e-07)*(x-8000)**2 &
+ (3.0923676105315428e-03)*(x-8000) + (3.0899999999999999
  e+00)
END IF
IF ((x>=8500).AND.(x<9000))THEN
  qnosehigh = (2.4307352210630880e-09)*(x-8500)**3 +
    (2.690000000000000013e-06)*(x-8500)**2 &
+ (3.9593161947342279e-03)*(x-8500) + (4.7009999999999996
  e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (1.4422323894684546e-08)*(x-9000)**3 +
    (6.3361028315946341e-06)*(x-9000)**2 &
+ (8.4723676105315452e-03)*(x-9000) + (7.6570000000000000
  e+00)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (-3.0000307998013198e-09)*(x-9500)**3 +
    (2.7969588673621479e-05)*(x-9500)**2 &

```

```

+ (2.5625213363139589e-02)*(x-9500) + (1.5279999999999999
    e+01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
    qnosehigh = (-3.9243214108296074e-09)*(x-10000)**3 +
        (2.3469542473919509e-05)*(x-10000)**2 &
+ (5.1344778936910078e-02)*(x-10000) +
        (3.4710000000000001e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
    qnosehigh = (8.9252210630903731e-10)*(x-11000)**3 +
        (1.1696578241430681e-05)*(x-11000)**2 &
+ (8.6510899652260276e-02)*(x-11000) +
        (1.0559999999999999e+02)
END IF
IF ((x>=12000).AND.(x<14000))THEN
    qnosehigh = (8.9252210630896442e-10)*(x-12000)**3 +
        (1.43741444560357698e-05)*(x-12000)**2 &
+ (1.1258162245404875e-01)*(x-12000) +
        (2.0469999999999999e+02)
END IF
    qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
    newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
rndiff = 1.0-(10.000000-nose)/(10.000000-5.000000)
IF ((nose>5.0000000).AND.(nose<=10.00)) THEN
    IF ((x>=0).AND.(x<8000))THEN

```

```

qnoselow = (1.6462499999999999e-03)*(x) +
            (0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
    qnoselow = (6.3360357675111756e-09)*(x-8000)**3 +
                (9.5946348733236820e-08)*(x-8000)**2 &
                + (1.1908017883755591e-02)*(x-8000) + (1.3170000000000000
                e+01)
END IF
IF ((x>=8500).AND.(x<9000))THEN
    qnoselow = (6.3360357675111839e-09)*(x-8500)**3 +
                (9.5999999999999894e-06)*(x-8500)**2 &
                + (1.6755991058122210e-02)*(x-8500) + (1.9940000000000001
                e+01)
END IF
IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (3.1039821162444155e-08)*(x-9000)**3 +
                (1.9104053651266744e-05)*(x-9000)**2 &
                + (3.1108017883755588e-02)*(x-9000) + (3.1510000000000002
                e+01)
END IF
IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (-5.4553204172876946e-09)*(x-9500)**3 +
                (6.5663785394932980e-05)*(x-9500)**2 &
                + (7.3491937406855445e-02)*(x-9500) + (5.571999999999999
                e+01)
END IF
IF ((x>=10000).AND.(x<11000))THEN

```

```

      qnoselow =  (-1.2045037257824157e-08)*(x-10000)**3 +
      (5.7480804769001492e-05)*(x-10000)**2 &
+ (1.3506423248882266e-01)*(x-10000) +
      (1.0820000000000000e+02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
      qnoselow =  (2.4635767511177204e-09)*(x-11000)**3 +
      (2.1345692995529101e-05)*(x-11000)**2 &
+ (2.1389073025335317e-01)*(x-11000) +
      (2.8869999999999999e+02)
END IF
IF ((x>=12000).AND.(x<14000))THEN
      qnoselow =  (2.4635767511177610e-09)*(x-12000)**3 +
      (2.8736423248882228e-05)*(x-12000)**2 &
+ (2.6397284649776454e-01)*(x-12000) +
      (5.2639999999999998e+02)
END IF
if ((x>=0).AND.(x<8000))THEN
      qnosehigh = (2.7425000000000001e-03)*(x) +
      (0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
      qnosehigh =  (9.4904918032787956e-09)*(x-8000)**3 +
      (-1.4757377049181707e-06)*(x-8000)**2 &
+ (2.0205245901639381e-02)*(x-8000) + (2.1940000000000001
      e+01)
END IF
IF ((x>=8500).AND.(x<9000))THEN

```

```

qnosehigh = (9.4904918032787311e-09)*(x-8500)**3 +
(1.27600000000000003e-05)*(x-8500)**2 &
+ (2.5847377049180307e-02)*(x-8500) + (3.2859999999999999
e+01)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnosehigh = (2.7907540983606425e-08)*(x-9000)**3 +
(2.6995737704918116e-05)*(x-9000)**2 &
+ (4.5725245901639358e-02)*(x-9000) + (5.0159999999999997
e+01)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnosehigh = (-2.4006557377049708e-09)*(x-9500)**3 +
(6.8857049180327855e-05)*(x-9500)**2 &
+ (9.3651639344262294e-02)*(x-9500) + (8.3260000000000005
e+01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnosehigh = (-1.5564262295081924e-08)*(x-10000)**3 +
(6.5256065573770480e-05)*(x-10000)**2 &
+ (1.6070819672131142e-01)*(x-10000) +
(1.4700000000000000e+02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnosehigh = (-8.9081967213114380e-10)*(x-11000)**3 +
(1.8563278688524587e-05)*(x-11000)**2 &
+ (2.4452754098360660e-01)*(x-11000) +
(3.5739999999999998e+02)

```

```

END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (-8.9081967213114380e-10)*(x-12000)**3 +
    (1.5890819672131102e-05)*(x-12000)**2 &
+ (2.7898163934426234e-01)*(x-12000) +
    (6.1960000000000002e+02)
END IF
  qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<8000))THEN
qnoselow = (3.862499999999998e-04)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
  qnoselow = (2.4307352210630859e-09)*(x-8000)**3 +
    (-9.5610283159462924e-07)*(x-8000)**2 &
+ (3.0923676105315428e-03)*(x-8000) + (3.089999999999999
    e+00)
END IF
IF ((x>=8500).AND.(x<9000))THEN
  qnoselow = (2.4307352210630880e-09)*(x-8500)**3 +
    (2.69000000000000013e-06)*(x-8500)**2 &
+ (3.9593161947342279e-03)*(x-8500) + (4.7009999999999996
    e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnoselow = (1.4422323894684546e-08)*(x-9000)**3 +
    (6.3361028315946341e-06)*(x-9000)**2 &

```



```

+ (8.4723676105315452e-03)*(x-9000) + (7.6570000000000000
e+00)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnoselow = (-3.0000307998013198e-09)*(x-9500)**3 +
    (2.7969588673621479e-05)*(x-9500)**2 &
+ (2.5625213363139589e-02)*(x-9500) + (1.5279999999999999
e+01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (-3.9243214108296074e-09)*(x-10000)**3 +
    (2.3469542473919509e-05)*(x-10000)**2 &
+ (5.1344778936910078e-02)*(x-10000) +
    (3.47100000000000001e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (8.9252210630903731e-10)*(x-11000)**3 +
    (1.1696578241430681e-05)*(x-11000)**2 &
+ (8.6510899652260276e-02)*(x-11000) +
    (1.0559999999999999e+02)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (8.9252210630896442e-10)*(x-12000)**3 +
    (1.4374144560357698e-05)*(x-12000)**2 &
+ (1.1258162245404875e-01)*(x-12000) +
    (2.0469999999999999e+02)
END IF
if ((x>=0).AND.(x<8000))THEN

```

```

qnosehigh = (7.0037499999999993e-04)*(x) +
            (0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
    qnosehigh = (3.3314585196224343e-09)*(x-8000)**3 +
                (-9.7518777943365189e-07)*(x-8000)**2 &
                + (5.1707292598112192e-03)*(x-8000) + (5.6029999999999998
                e+00)
END IF
IF ((x>=8500).AND.(x<9000))THEN
    qnosehigh = (3.3314585196224380e-09)*(x-8500)**3 +
                (4.0219999999999938e-06)*(x-8500)**2 &
                + (6.6941353700943930e-03)*(x-8500) + (8.3610000000000007
                e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
    qnosehigh = (1.7262707401887758e-08)*(x-9000)**3 +
                (9.0191877794336849e-06)*(x-9000)**2 &
                + (1.3214729259811216e-02)*(x-9000) + (1.3130000000000001
                e+01)
END IF
IF ((x>=9500).AND.(x<10000))THEN
    qnosehigh = (-1.0630288127173397e-08)*(x-9500)**3 +
                (3.4913248882265270e-05)*(x-9500)**2 &
                + (3.5180947590660719e-02)*(x-9500) + (2.4149999999999999
                e+01)
END IF
IF ((x>=10000).AND.(x<11000))THEN

```

```

      qnosehigh = (1.3707029309488527e-09)*(x-10000)**3 +
        (1.8967816691505201e-05)*(x-10000)**2 &
+ (6.2121480377545939e-02)*(x-10000) +
        (4.9140000000000001e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
      qnosehigh = (-3.5491480377546478e-09)*(x-11000)**3 +
        (2.3079925484351766e-05)*(x-11000)**2 &
+ (1.0416922255340290e-01)*(x-11000) +
        (1.3159999999999999e+02)
END IF
IF ((x>=12000).AND.(x<14000))THEN
      qnosehigh = (-3.5491480377545928e-09)*(x-12000)**3 +
        (1.2432481371087934e-05)*(x-12000)**2 &
+ (1.3968162940884249e-01)*(x-12000) +
        (2.5530000000000001e+02)
END IF
      qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
      newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if

```

```

END IF !density if

```

```

!!!!!!!!!!!!!!!!!!!!NEW ALTITUDE RANGE!!!!!!!!!!!!!!!!!!!!

```

```

IF ((rho<=0.0029948000).AND.(rho>=0.0013167000)) THEN
      rhodiff = (0.0029948000-rho)/(0.0029948000-0.0013167000)
      rndiff = 1.0-(0.100000-nose)/(0.100000-0.049999)

```

```

IF ((nose > 0.0499990) .AND. (nose <= 0.10)) THEN
  IF ((x >= 0) .AND. (x < 8000)) THEN
    qnoselow = (6.353749999999999e-06)*(x) +
      (0.0000000000000000e+00)
  END IF
  IF ((x >= 8000) .AND. (x < 8500)) THEN
    qnoselow = (1.6041177347242870e-10)*(x-8000)**3 +
      (-8.6477660208643074e-08)*(x-8000)**2 &
      + (8.7635886736214360e-05)*(x-8000) + (5.0830000000000000
        e-02)
  END IF
  IF ((x >= 8500) .AND. (x < 9000)) THEN
    qnoselow = (1.6041177347242826e-10)*(x-8500)**3 +
      (1.54140000000000029e-07)*(x-8500)**2 &
      + (1.2146705663189280e-04)*(x-8500) + (9.3079999999999996
        e-02)
  END IF
  IF ((x >= 9000) .AND. (x < 9500)) THEN
    qnoselow = (2.4436211326378547e-09)*(x-9000)**3 +
      (3.9475766020864386e-07)*(x-9000)**2 &
      + (3.9591588673621429e-04)*(x-9000) + (2.1240000000000001
        e-01)
  END IF
  IF ((x >= 9500) .AND. (x < 10000)) THEN
    qnoselow = (1.5580636959761644e-09)*(x-9500)**3 +
      (4.0601893591654176e-06)*(x-9500)**2 &
      + (2.6233893964232491e-03)*(x-9500) + (8.1450000000000000
        e-01)
  END IF

```

```

END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (8.4588569299548476e-11)*(x-10000)**3 +
    (6.3972849031296627e-06)*(x-10000)**2 &
+ (7.8521265275707901e-03)*(x-10000) +
    (3.3359999999999999e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (1.2848734724291853e-10)*(x-11000)**3 +
    (6.6510506110283199e-06)*(x-11000)**2 &
+ (2.0900462041728761e-02)*(x-11000) +
    (1.76700000000000002e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (1.2848734724292742e-10)*(x-12000)**3 +
    (7.0365126527570645e-06)*(x-12000)**2 &
+ (3.4588025305514157e-02)*(x-12000) +
    (4.53500000000000001e+01)
END IF
if ((x>=0).AND.(x<8000))THEN
  qnosehigh = (1.1611250000000000e-05)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
  qnosehigh = (2.4094228514654682e-10)*(x-8000)**3 +
    (-1.2583342771982022e-07)*(x-8000)**2 &
+ (1.4350114257327340e-04)*(x-8000) + (9.2890000000000000
e-02)

```

```

END IF
IF ((x>=8500).AND.(x<9000))THEN
  qnosehigh = (2.4094228514654801e-10)*(x-8500)**3 +
    (2.3557999999999925e-07)*(x-8500)**2 &
+ (1.9837442871336331e-04)*(x-8500) + (1.6330000000000000
  e-01)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (3.3833685742672636e-09)*(x-9000)**3 +
    (5.9699342771982167e-07)*(x-9000)**2 &
+ (6.1466114257327357e-04)*(x-9000) + (3.5149999999999998
  e-01)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (1.5711834177843861e-09)*(x-9500)**3 +
    (5.6720462891207214e-06)*(x-9500)**2 &
+ (3.7491810009935428e-03)*(x-9500) + (1.2310000000000001
  e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (-1.9843626924986609e-10)*(x-10000)**3 +
    (8.0288214157973108e-06)*(x-10000)**2 &
+ (1.0599614853452554e-02)*(x-10000) +
    (4.7199999999999998e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (2.4538514654746398e-11)*(x-11000)**3 +
    (7.4335126080476787e-06)*(x-11000)**2 &

```

```

+ (2.6061948877297576e-02)*(x-11000) +
    (2.3149999999999999e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
    qnosehigh = (2.4538514654750634e-11)*(x-12000)**3 +
        (7.5071281520119145e-06)*(x-12000)**2 &
+ (4.1002589637357173e-02)*(x-12000) +
    (5.6670000000000002e+01)
END IF
    qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<9000))THEN
    qnoselow = (6.0855555555555558e-06)*(x) +
        (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (6.1879650793650763e-10)*(x-9000)**3 +
        (4.6534523809523827e-07)*(x-9000)**2 &
+ (3.5088253968253993e-05)*(x-9000) + (5.4769999999999999
    e-02)
END IF
IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (6.1879650793650639e-10)*(x-9500)**3 +
        (1.3935400000000006e-06)*(x-9500)**2 &
+ (9.6453087301587304e-04)*(x-9500) + (2.6600000000000001
    e-01)
END IF
IF ((x>=10000).AND.(x<11000))THEN

```

```

      qnoselow = (8.6096984126984615e-11)*(x-10000)**3 +
      (2.3217347619047624e-06)*(x-10000)**2 &
+ (2.8221682539682534e-03)*(x-10000) +
      (1.1739999999999999e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
      qnoselow = (1.32045555555555699e-10)*(x-11000)**3 +
      (2.5800257142857103e-06)*(x-11000)**2 &
+ (7.7239287301587318e-03)*(x-11000) +
      (6.4039999999999999e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
      qnoselow = (8.3139603174605556e-11)*(x-12000)**3 +
      (2.9761623809523766e-06)*(x-12000)**2 &
+ (1.3280116825396824e-02)*(x-12000) +
      (1.6840000000000000e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
      qnosehigh = (1.0395555555555555e-05)*(x) +
      (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
      qnosehigh = (6.8667788359788124e-10)*(x-9000)**3 +
      (9.9310317460317866e-07)*(x-9000)**2 &
+ (-3.9341058201059818e-05)*(x-9000) +
      (9.3560000000000000e-02)
END IF
IF ((x>=9500).AND.(x<10000))THEN

```



```

    qnosehigh = (6.8667788359788082e-10)*(x-9500)**3 +
        (2.0231200000000007e-06)*(x-9500)**2 &
+ (1.4687705291005298e-03)*(x-9500) + (4.0799999999999997
    e-01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
    qnosehigh = (2.9964232804234473e-11)*(x-10000)**3 +
        (3.0531368253968245e-06)*(x-10000)**2 &
+ (4.0068989417989410e-03)*(x-10000) +
        (1.7340000000000000e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
    qnosehigh = (9.9905185185187399e-11)*(x-11000)**3 +
        (3.1430295238095186e-06)*(x-11000)**2 &
+ (1.0203065291005294e-02)*(x-11000) +
        (8.823999999999998e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
    qnosehigh = (6.3917486772489647e-11)*(x-12000)**3 +
        (3.4427450793650740e-06)*(x-12000)**2 &
+ (1.6788839894179893e-02)*(x-12000) +
        (2.2270000000000000e+01)
END IF
    qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
    newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
    rndiff = 1.0-(0.230000-nose)/(0.230000-0.100000)
    IF ((nose>0.1000000).AND.(nose<=0.23)) THEN

```

```

IF ((x>=0).AND.(x<8000))THEN
  qnoselow = (1.1611250000000000e-05)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
  qnoselow = (2.4094228514654682e-10)*(x-8000)**3 +
    (-1.2583342771982022e-07)*(x-8000)**2 &
    + (1.4350114257327340e-04)*(x-8000) + (9.2890000000000000
      e-02)
END IF
IF ((x>=8500).AND.(x<9000))THEN
  qnoselow = (2.4094228514654801e-10)*(x-8500)**3 +
    (2.3557999999999925e-07)*(x-8500)**2 &
    + (1.9837442871336331e-04)*(x-8500) + (1.6330000000000000
      e-01)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnoselow = (3.3833685742672636e-09)*(x-9000)**3 +
    (5.9699342771982167e-07)*(x-9000)**2 &
    + (6.1466114257327357e-04)*(x-9000) + (3.5149999999999998
      e-01)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnoselow = (1.5711834177843861e-09)*(x-9500)**3 +
    (5.6720462891207214e-06)*(x-9500)**2 &
    + (3.7491810009935428e-03)*(x-9500) + (1.2310000000000001
      e+00)
END IF

```

```

IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (-1.9843626924986609e-10)*(x-10000)**3 +
    (8.0288214157973108e-06)*(x-10000)**2 &
  + (1.0599614853452554e-02)*(x-10000) +
    (4.7199999999999998e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (2.4538514654746398e-11)*(x-11000)**3 +
    (7.4335126080476787e-06)*(x-11000)**2 &
  + (2.6061948877297576e-02)*(x-11000) +
    (2.3149999999999999e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (2.4538514654750634e-11)*(x-12000)**3 +
    (7.5071281520119145e-06)*(x-12000)**2 &
  + (4.1002589637357173e-02)*(x-12000) +
    (5.66700000000000002e+01)
END IF
if ((x>=0).AND.(x<8000))THEN
  qnosehigh = (2.4324999999999998e-05)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
  qnosehigh = (3.8542583209140470e-10)*(x-8000)**3 +
    (-2.0273874813710691e-07)*(x-8000)**2 &
  + (2.6941291604570225e-04)*(x-8000) + (1.9460000000000000
    e-01)
END IF

```

```

IF ((x>=8500).AND.(x<9000))THEN
  qnosehigh = (3.8542583209140357e-10)*(x-8500)**3 +
    (3.75400000000000078e-07)*(x-8500)**2 &
+ (3.5574354197714887e-04)*(x-8500) + (3.2679999999999998
  e-01)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (4.8544708395429733e-09)*(x-9000)**3 +
    (9.5353874813710818e-07)*(x-9000)**2 &
+ (1.0202129160457023e-03)*(x-9000) + (6.4670000000000005
  e-01)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (1.2630908097367017e-09)*(x-9500)**3 +
    (8.2352450074515675e-06)*(x-9500)**2 &
+ (5.6146047938400402e-03)*(x-9500) + (2.0019999999999998
  e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (-6.0304913065077303e-10)*(x-10000)**3 +
    (1.0129881222056640e-05)*(x-10000)**2 &
+ (1.4797167908594134e-02)*(x-10000) +
    (7.0259999999999998e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (-5.8516790859421312e-11)*(x-11000)**3 +
    (8.3207338301043230e-06)*(x-11000)**2 &

```

```

+ (3.3247782960755097e-02)*(x-11000) +
    (3.1350000000000001e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
    qnosehigh = (-5.8516790859393356e-11)*(x-12000)**3 +
        (8.1451834575260523e-06)*(x-12000)**2 &
+ (4.9713700248385478e-02)*(x-12000) +
    (7.285999999999999e+01)
END IF
    qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<9000))THEN
    qnoselow = (1.0395555555555555e-05)*(x) +
        (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (6.8667788359788124e-10)*(x-9000)**3 +
        (9.9310317460317866e-07)*(x-9000)**2 &
+ (-3.9341058201059818e-05)*(x-9000) +
    (9.3560000000000004e-02)
END IF
IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (6.8667788359788082e-10)*(x-9500)**3 +
        (2.0231200000000007e-06)*(x-9500)**2 &
+ (1.4687705291005298e-03)*(x-9500) + (4.0799999999999997
    e-01)
END IF
IF ((x>=10000).AND.(x<11000))THEN

```

```

      qnoselow = (2.9964232804234473e-11)*(x-10000)**3 +
      (3.0531368253968245e-06)*(x-10000)**2 &
+ (4.0068989417989410e-03)*(x-10000) +
      (1.7340000000000000e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
      qnoselow = (9.9905185185187399e-11)*(x-11000)**3 +
      (3.1430295238095186e-06)*(x-11000)**2 &
+ (1.0203065291005294e-02)*(x-11000) +
      (8.8239999999999998e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
      qnoselow = (6.3917486772489647e-11)*(x-12000)**3 +
      (3.4427450793650740e-06)*(x-12000)**2 &
+ (1.6788839894179893e-02)*(x-12000) +
      (2.2270000000000000e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
      qnosehigh = (1.93444444444444446e-05)*(x) +
      (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
      qnosehigh = (6.8506772486772668e-10)*(x-9000)**3 +
      (1.9473984126984121e-06)*(x-9000)**2 &
+ (-1.3256613756613777e-04)*(x-9000) +
      (1.7410000000000000e-01)
END IF
IF ((x>=9500).AND.(x<10000))THEN

```

```

    qnosehigh = (6.8506772486773092e-10)*(x-9500)**3 +
        (2.9749999999999969e-06)*(x-9500)**2 &
+ (2.3286330687830692e-03)*(x-9500) + (6.8030000000000002
    e-01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
    qnosehigh = (-1.0403544973545408e-10)*(x-10000)**3 +
        (4.0026015873015907e-06)*(x-10000)**2 &
+ (5.8174338624338642e-03)*(x-10000) +
        (2.6739999999999999e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
    qnosehigh = (-1.1025925925925492e-11)*(x-11000)**3 +
        (3.6904952380952395e-06)*(x-11000)**2 &
+ (1.3510530687830683e-02)*(x-11000) +
        (1.23900000000000001e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
    qnosehigh = (1.6680423280421707e-11)*(x-12000)**3 +
        (3.6574174603174647e-06)*(x-12000)**2 &
+ (2.0858443386243385e-02)*(x-12000) +
        (2.9579999999999998e+01)
END IF
    qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
    newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
    rndiff = 1.0-(0.500000-nose)/(0.500000-0.230000)
IF ((nose>0.2300000).AND.(nose<=0.50)) THEN

```

```

IF ((x>=0).AND.(x<8000))THEN
qnoselow = (2.4324999999999998e-05)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
qnoselow = (3.8542583209140470e-10)*(x-8000)**3 +
(-2.0273874813710691e-07)*(x-8000)**2 &
+ (2.6941291604570225e-04)*(x-8000) + (1.9460000000000000
e-01)
END IF
IF ((x>=8500).AND.(x<9000))THEN
qnoselow = (3.8542583209140357e-10)*(x-8500)**3 +
(3.75400000000000078e-07)*(x-8500)**2 &
+ (3.5574354197714887e-04)*(x-8500) + (3.2679999999999998
e-01)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnoselow = (4.8544708395429733e-09)*(x-9000)**3 +
(9.5353874813710818e-07)*(x-9000)**2 &
+ (1.0202129160457023e-03)*(x-9000) + (6.4670000000000005
e-01)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnoselow = (1.2630908097367017e-09)*(x-9500)**3 +
(8.2352450074515675e-06)*(x-9500)**2 &
+ (5.6146047938400402e-03)*(x-9500) + (2.0019999999999998
e+00)
END IF

```



```

IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (-6.0304913065077303e-10)*(x-10000)**3 +
    (1.0129881222056640e-05)*(x-10000)**2 &
  + (1.4797167908594134e-02)*(x-10000) +
    (7.0259999999999998e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (-5.8516790859421312e-11)*(x-11000)**3 +
    (8.3207338301043230e-06)*(x-11000)**2 &
  + (3.3247782960755097e-02)*(x-11000) +
    (3.13500000000000001e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (-5.8516790859393356e-11)*(x-12000)**3 +
    (8.1451834575260523e-06)*(x-12000)**2 &
  + (4.9713700248385478e-02)*(x-12000) +
    (7.2859999999999999e+01)
END IF
if ((x>=0).AND.(x<8000))THEN
  qnosehigh = (4.8974999999999998e-05)*(x) +
    (0.00000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
  qnosehigh = (7.4204649776452812e-10)*(x-8000)**3 +
    (-5.4186974664679266e-07)*(x-8000)**2 &
  + (5.6102324888226442e-04)*(x-8000) + (3.9179999999999998
    e-01)
END IF

```

```

IF ((x>=8500).AND.(x<9000))THEN
  qnosehigh = (7.4204649776453153e-10)*(x-8500)**3 +
    (5.7119999999999823e-07)*(x-8500)**2 &
  + (5.7568837555886784e-04)*(x-8500) + (6.2960000000000005
    e-01)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (6.2897675111773582e-09)*(x-9000)**3 +
    (1.6842697466467921e-06)*(x-9000)**2 &
  + (1.7034232488822648e-03)*(x-9000) + (1.1530000000000000
    e+00)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (1.0860834575260694e-09)*(x-9500)**3 +
    (1.1118921013412814e-05)*(x-9500)**2 &
  + (8.1050186289120755e-03)*(x-9500) + (3.2120000000000002
    e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (-9.5654843517137231e-10)*(x-10000)**3 +
    (1.2748046199701927e-05)*(x-10000)**2 &
  + (2.0038502235469442e-02)*(x-10000) +
    (1.0180000000000000e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (-2.6335022354699040e-10)*(x-11000)**3 +
    (9.8784008941878188e-06)*(x-11000)**2 &

```

```

+ (4.2664949329359179e-02)*(x-11000) +
    (4.2009999999999998e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
    qnosehigh = (-2.6335022354692014e-10)*(x-12000)**3 +
        (9.0883502235469136e-06)*(x-12000)**2 &
+ (6.1631700447093846e-02)*(x-12000) +
    (9.42900000000000006e+01)
END IF
    qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<9000))THEN
    qnoselow = (1.9344444444444446e-05)*(x) +
        (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (6.8506772486772668e-10)*(x-9000)**3 +
        (1.9473984126984121e-06)*(x-9000)**2 &
+ (-1.3256613756613777e-04)*(x-9000) +
    (1.7410000000000000e-01)
END IF
IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (6.8506772486773092e-10)*(x-9500)**3 +
        (2.9749999999999969e-06)*(x-9500)**2 &
+ (2.3286330687830692e-03)*(x-9500) + (6.8030000000000002
    e-01)
END IF
IF ((x>=10000).AND.(x<11000))THEN

```

```

      qnoselow =  (-1.0403544973545408e-10)*(x-10000)**3 +
      (4.0026015873015907e-06)*(x-10000)**2 &
+ (5.8174338624338642e-03)*(x-10000) +
      (2.6739999999999999e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
      qnoselow =  (-1.1025925925925492e-11)*(x-11000)**3 +
      (3.6904952380952395e-06)*(x-11000)**2 &
+ (1.3510530687830683e-02)*(x-11000) +
      (1.23900000000000001e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
      qnoselow =  (1.6680423280421707e-11)*(x-12000)**3 +
      (3.6574174603174647e-06)*(x-12000)**2 &
+ (2.0858443386243385e-02)*(x-12000) +
      (2.9579999999999998e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
      qnosehigh = (3.4188888888888885e-05)*(x) +
      (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
      qnosehigh =  (6.4466878306878408e-10)*(x-9000)**3 +
      (3.0763968253968224e-06)*(x-9000)**2 &
+ (-1.4676560846560704e-04)*(x-9000) +
      (3.0769999999999997e-01)
END IF
IF ((x>=9500).AND.(x<10000))THEN

```

```

      qnosehigh = (6.4466878306877137e-10)*(x-9500)**3 +
                  (4.0434000000000086e-06)*(x-9500)**2 &
+ (3.4131328042328033e-03)*(x-9500) + (1.0840000000000001
      e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN
      qnosehigh = (-3.3243756613756546e-10)*(x-10000)**3 +
                  (5.0104031746031758e-06)*(x-10000)**2 &
+ (7.9400343915343903e-03)*(x-10000) +
                  (3.8820000000000001e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
      qnosehigh = (-2.6618518518515803e-11)*(x-11000)**3 +
                  (4.0130904761904735e-06)*(x-11000)**2 &
+ (1.6963528042328046e-02)*(x-11000) +
                  (1.6500000000000000e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
      qnosehigh = (-6.2830820105814792e-11)*(x-12000)**3 +
                  (3.9332349206349058e-06)*(x-12000)**2 &
+ (2.4909853439153445e-02)*(x-12000) +
                  (3.7450000000000003e+01)
END IF
      qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
      newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
      rndiff = 1.0-(1.000000-nose)/(1.000000-0.500000)
      IF ((nose>0.5000000).AND.(nose<=1.00)) THEN

```

```

IF ((x>=0).AND.(x<8000))THEN
qnoselow = (4.897499999999998e-05)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
qnoselow = (7.4204649776452812e-10)*(x-8000)**3 +
(-5.4186974664679266e-07)*(x-8000)**2 &
+ (5.6102324888226442e-04)*(x-8000) + (3.917999999999998
e-01)
END IF
IF ((x>=8500).AND.(x<9000))THEN
qnoselow = (7.4204649776453153e-10)*(x-8500)**3 +
(5.71199999999999823e-07)*(x-8500)**2 &
+ (5.7568837555886784e-04)*(x-8500) + (6.2960000000000005
e-01)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnoselow = (6.2897675111773582e-09)*(x-9000)**3 +
(1.6842697466467921e-06)*(x-9000)**2 &
+ (1.7034232488822648e-03)*(x-9000) + (1.1530000000000000
e+00)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnoselow = (1.0860834575260694e-09)*(x-9500)**3 +
(1.1118921013412814e-05)*(x-9500)**2 &
+ (8.1050186289120755e-03)*(x-9500) + (3.2120000000000002
e+00)
END IF

```

```

IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (-9.5654843517137231e-10)*(x-10000)**3 +
    (1.2748046199701927e-05)*(x-10000)**2 &
+ (2.0038502235469442e-02)*(x-10000) +
    (1.0180000000000000e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (-2.6335022354699040e-10)*(x-11000)**3 +
    (9.8784008941878188e-06)*(x-11000)**2 &
+ (4.2664949329359179e-02)*(x-11000) +
    (4.2009999999999998e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (-2.6335022354692014e-10)*(x-12000)**3 +
    (9.0883502235469136e-06)*(x-12000)**2 &
+ (6.1631700447093846e-02)*(x-12000) +
    (9.4290000000000000e+01)
END IF
if ((x>=0).AND.(x<8000))THEN
  qnosehigh = (9.0637500000000001e-05)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
  qnosehigh = (1.0391702930948753e-09)*(x-8000)**3 +
    (-6.3655543964231260e-07)*(x-8000)**2 &
+ (8.8028514654743745e-04)*(x-8000) + (7.2509999999999997
    e-01)
END IF

```

```

IF ((x>=8500).AND.(x<9000))THEN
  qnosehigh = (1.0391702930948777e-09)*(x-8500)**3 +
    (9.2219999999999940e-07)*(x-8500)**2 &
+ (1.0231074267262811e-03)*(x-8500) + (1.1359999999999999
  e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (8.1713485345255934e-09)*(x-9000)**3 +
    (2.4809554396423249e-06)*(x-9000)**2 &
+ (2.7246851465474388e-03)*(x-9000) + (2.0080000000000000
  e+00)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (7.7143556880279347e-10)*(x-9500)**3 +
    (1.4737978241430685e-05)*(x-9500)**2 &
+ (1.1334151987083959e-02)*(x-9500) + (5.0119999999999996
  e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (-1.4358384997516119e-09)*(x-10000)**3 +
    (1.5895131594634874e-05)*(x-10000)**2 &
+ (2.6650706905116738e-02)*(x-10000) +
    (1.44600000000000001e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (-4.9107069051169983e-10)*(x-11000)**3 +
    (1.1587616095380046e-05)*(x-11000)**2 &

```



```

+ (5.4133454595131650e-02)*(x-11000) +
    (5.5570000000000000e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
    qnosehigh = (-4.9107069051164731e-10)*(x-12000)**3 +
        (1.0114404023844967e-05)*(x-12000)**2 &
+ (7.5835474714356643e-02)*(x-12000) +
    (1.2080000000000000e+02)
END IF
    qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<9000))THEN
    qnoselow = (3.4188888888888885e-05)*(x) +
        (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (6.4466878306878408e-10)*(x-9000)**3 +
        (3.0763968253968224e-06)*(x-9000)**2 &
+ (-1.4676560846560704e-04)*(x-9000) +
    (3.0769999999999997e-01)
END IF
IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (6.4466878306877137e-10)*(x-9500)**3 +
        (4.04340000000000086e-06)*(x-9500)**2 &
+ (3.4131328042328033e-03)*(x-9500) + (1.0840000000000001
    e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN

```

```

      qnoselow =  (-3.3243756613756546e-10)*(x-10000)**3 +
                  (5.0104031746031758e-06)*(x-10000)**2 &
+ (7.9400343915343903e-03)*(x-10000) +
                  (3.8820000000000001e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
      qnoselow =  (-2.6618518518515803e-11)*(x-11000)**3 +
                  (4.0130904761904735e-06)*(x-11000)**2 &
+ (1.6963528042328046e-02)*(x-11000) +
                  (1.6500000000000000e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
      qnoselow =  (-6.2830820105814792e-11)*(x-12000)**3 +
                  (3.9332349206349058e-06)*(x-12000)**2 &
+ (2.4909853439153445e-02)*(x-12000) +
                  (3.7450000000000003e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
      qnosehigh = (5.7700000000000000e-05)*(x) +
                  (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
      qnosehigh =  (6.04844444444445382e-10)*(x-9000)**3 +
                  (4.3093333333333228e-06)*(x-9000)**2 &
+ (-5.64777777777775631e-05)*(x-9000) +
                  (5.1929999999999998e-01)
END IF
IF ((x>=9500).AND.(x<10000))THEN

```

```

    qnosehigh = (6.04844444444443180e-10)*(x-9500)**3 +
        (5.21660000000000078e-06)*(x-9500)**2 &
+ (4.7064888888888878e-03)*(x-9500) + (1.6439999999999999
    e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN
    qnosehigh = (-6.8758888888889111e-10)*(x-10000)**3 +
        (6.12386666666666725e-06)*(x-10000)**2 &
+ (1.0376722222222219e-02)*(x-10000) +
        (5.3769999999999998e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
    qnosehigh = (5.77211111111110053e-10)*(x-11000)**3 +
        (4.06110000000000084e-06)*(x-11000)**2 &
+ (2.0561688888888891e-02)*(x-11000) +
        (2.11900000000000001e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
    qnosehigh = (-3.23997222222222003e-10)*(x-12000)**3 +
        (5.7927333333333339e-06)*(x-12000)**2 &
+ (3.0415522222222210e-02)*(x-12000) +
        (4.63900000000000001e+01)
END IF
    qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
    newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
    rndiff = 1.0-(2.300000-nose)/(2.300000-1.000000)
IF ((nose>1.0000000).AND.(nose<=2.30)) THEN

```

```

IF ((x>=0).AND.(x<8000))THEN
  qnoselow = (9.0637500000000001e-05)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
  qnoselow = (1.0391702930948753e-09)*(x-8000)**3 +
    (-6.3655543964231260e-07)*(x-8000)**2 &
    + (8.8028514654743745e-04)*(x-8000) + (7.2509999999999997
      e-01)
END IF
IF ((x>=8500).AND.(x<9000))THEN
  qnoselow = (1.0391702930948777e-09)*(x-8500)**3 +
    (9.22199999999999940e-07)*(x-8500)**2 &
    + (1.0231074267262811e-03)*(x-8500) + (1.1359999999999999
      e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnoselow = (8.1713485345255934e-09)*(x-9000)**3 +
    (2.4809554396423249e-06)*(x-9000)**2 &
    + (2.7246851465474388e-03)*(x-9000) + (2.0080000000000000
      e+00)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnoselow = (7.7143556880279347e-10)*(x-9500)**3 +
    (1.4737978241430685e-05)*(x-9500)**2 &
    + (1.1334151987083959e-02)*(x-9500) + (5.0119999999999996
      e+00)
END IF

```

```

IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (-1.4358384997516119e-09)*(x-10000)**3 +
    (1.5895131594634874e-05)*(x-10000)**2 &
  + (2.6650706905116738e-02)*(x-10000) +
    (1.44600000000000001e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (-4.9107069051169983e-10)*(x-11000)**3 +
    (1.1587616095380046e-05)*(x-11000)**2 &
  + (5.4133454595131650e-02)*(x-11000) +
    (5.5570000000000000e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (-4.9107069051164731e-10)*(x-12000)**3 +
    (1.0114404023844967e-05)*(x-12000)**2 &
  + (7.5835474714356643e-02)*(x-12000) +
    (1.2080000000000000e+02)
END IF
if ((x>=0).AND.(x<8000))THEN
  qnosehigh = (1.9374999999999999e-04)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
  qnosehigh = (1.6502533532041672e-09)*(x-8000)**3 +
    (-8.7938002980625326e-07)*(x-8000)**2 &
  + (1.6871266766020846e-03)*(x-8000) + (1.5500000000000000
    e+00)
END IF

```

```

IF ((x>=8500).AND.(x<9000))THEN
  qnosehigh = (1.6502533532041805e-09)*(x-8500)**3 +
    (1.5959999999999967e-06)*(x-8500)**2 &
+ (2.0454366616989568e-03)*(x-8500) + (2.3799999999999999
  e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (1.1084733233979133e-08)*(x-9000)**3 +
    (4.0713800298062593e-06)*(x-9000)**2 &
+ (4.8791266766020887e-03)*(x-9000) + (4.0080000000000000
  e+00)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (1.8813710879254615e-11)*(x-9500)**3 +
    (2.0698479880774977e-05)*(x-9500)**2 &
+ (1.7264056631892698e-02)*(x-9500) + (8.8510000000000009
  e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (-2.8733472429209877e-09)*(x-10000)**3 +
    (2.0726700447093872e-05)*(x-10000)**2 &
+ (3.7976646795827115e-02)*(x-10000) +
    (2.2660000000000000e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (3.9333532041725724e-10)*(x-11000)**3 +
    (1.2106658718330872e-05)*(x-11000)**2 &

```

```

+ (7.0810005961251893e-02)*(x-11000) +
  (7.8489999999999995e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (3.9333532041730212e-10)*(x-12000)**3 +
    (1.3286664679582685e-05)*(x-12000)**2 &
+ (9.6203329359165410e-02)*(x-12000) +
  (1.61800000000000001e+02)
END IF
  qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<9000))THEN
  qnoselow = (5.7700000000000000e-05)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnoselow = (6.04844444444445382e-10)*(x-9000)**3 +
    (4.30933333333333228e-06)*(x-9000)**2 &
+ (-5.64777777777775631e-05)*(x-9000) +
  (5.1929999999999998e-01)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnoselow = (6.04844444444443180e-10)*(x-9500)**3 +
    (5.21660000000000078e-06)*(x-9500)**2 &
+ (4.7064888888888878e-03)*(x-9500) + (1.6439999999999999
  e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN

```

```

qnoselow = (-6.8758888888889111e-10)*(x-10000)**3 +
(6.1238666666666725e-06)*(x-10000)**2 &
+ (1.0376722222222219e-02)*(x-10000) +
(5.3769999999999998e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnoselow = (5.77211111111110053e-10)*(x-11000)**3 +
(4.06110000000000084e-06)*(x-11000)**2 &
+ (2.0561688888888891e-02)*(x-11000) +
(2.11900000000000001e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
qnoselow = (-3.2399722222222003e-10)*(x-12000)**3 +
(5.7927333333333339e-06)*(x-12000)**2 &
+ (3.0415522222222210e-02)*(x-12000) +
(4.63900000000000001e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
qnosehigh = (1.12444444444444445e-04)*(x) +
(0.00000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnosehigh = (4.6965079365081316e-10)*(x-9000)**3 +
(6.3215238095237838e-06)*(x-9000)**2 &
+ (2.1382539682540447e-04)*(x-9000) + (1.0120000000000000
e+00)
END IF
IF ((x>=9500).AND.(x<10000))THEN

```



```

      qnosehigh = (4.6965079365078265e-10)*(x-9500)**3 +
                  (7.02600000000000112e-06)*(x-9500)**2 &
+ (6.8875873015872982e-03)*(x-9500) + (2.7580000000000000
      e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN
      qnosehigh = (-1.0433015873015943e-09)*(x-10000)**3 +
                  (7.7304761904761962e-06)*(x-10000)**2 &
+ (1.4265825396825397e-02)*(x-10000) +
                  (8.0169999999999995e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
      qnosehigh = (8.12555555555555568e-10)*(x-11000)**3 +
                  (4.6005714285714336e-06)*(x-11000)**2 &
+ (2.6596873015873008e-02)*(x-11000) +
                  (2.8969999999999999e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
      qnosehigh = (-4.8803968253968532e-10)*(x-12000)**3 +
                  (7.0382380952381006e-06)*(x-12000)**2 &
+ (3.8235682539682542e-02)*(x-12000) +
                  (6.0979999999999997e+01)
END IF
      qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
      newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
      rndiff = 1.0-(5.000000-nose)/(5.000000-2.300000)
      IF ((nose>2.3000000).AND.(nose<=5.00)) THEN

```

```

IF ((x>=0).AND.(x<8000))THEN
qnoselow = (1.9374999999999999e-04)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
qnoselow = (1.6502533532041672e-09)*(x-8000)**3 +
(-8.7938002980625326e-07)*(x-8000)**2 &
+ (1.6871266766020846e-03)*(x-8000) + (1.5500000000000000
e+00)
END IF
IF ((x>=8500).AND.(x<9000))THEN
qnoselow = (1.6502533532041805e-09)*(x-8500)**3 +
(1.59599999999999967e-06)*(x-8500)**2 &
+ (2.0454366616989568e-03)*(x-8500) + (2.3799999999999999
e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnoselow = (1.1084733233979133e-08)*(x-9000)**3 +
(4.0713800298062593e-06)*(x-9000)**2 &
+ (4.8791266766020887e-03)*(x-9000) + (4.0080000000000000
e+00)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnoselow = (1.8813710879254615e-11)*(x-9500)**3 +
(2.0698479880774977e-05)*(x-9500)**2 &
+ (1.7264056631892698e-02)*(x-9500) + (8.8510000000000009
e+00)
END IF

```

```

IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (-2.8733472429209877e-09)*(x-10000)**3 +
    (2.0726700447093872e-05)*(x-10000)**2 &
  + (3.7976646795827115e-02)*(x-10000) +
    (2.2660000000000000e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (3.9333532041725724e-10)*(x-11000)**3 +
    (1.2106658718330872e-05)*(x-11000)**2 &
  + (7.0810005961251893e-02)*(x-11000) +
    (7.8489999999999995e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (3.9333532041730212e-10)*(x-12000)**3 +
    (1.3286664679582685e-05)*(x-12000)**2 &
  + (9.6203329359165410e-02)*(x-12000) +
    (1.61800000000000001e+02)
END IF
if ((x>=0).AND.(x<8000))THEN
  qnosehigh = (3.8624999999999998e-04)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
  qnosehigh = (2.4307352210630859e-09)*(x-8000)**3 +
    (-9.5610283159462924e-07)*(x-8000)**2 &
  + (3.0923676105315428e-03)*(x-8000) + (3.0899999999999999
    e+00)
END IF

```

```

IF ((x>=8500).AND.(x<9000))THEN
  qnosehigh = (2.4307352210630880e-09)*(x-8500)**3 +
    (2.69000000000000013e-06)*(x-8500)**2 &
+ (3.9593161947342279e-03)*(x-8500) + (4.7009999999999996
  e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (1.4422323894684546e-08)*(x-9000)**3 +
    (6.3361028315946341e-06)*(x-9000)**2 &
+ (8.4723676105315452e-03)*(x-9000) + (7.6570000000000000
  e+00)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (-3.0000307998013198e-09)*(x-9500)**3 +
    (2.7969588673621479e-05)*(x-9500)**2 &
+ (2.5625213363139589e-02)*(x-9500) + (1.5279999999999999
  e+01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (-3.9243214108296074e-09)*(x-10000)**3 +
    (2.3469542473919509e-05)*(x-10000)**2 &
+ (5.1344778936910078e-02)*(x-10000) +
    (3.47100000000000001e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (8.9252210630903731e-10)*(x-11000)**3 +
    (1.1696578241430681e-05)*(x-11000)**2 &

```

```

+ (8.6510899652260276e-02)*(x-11000) +
  (1.0559999999999999e+02)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (8.9252210630896442e-10)*(x-12000)**3 +
    (1.4374144560357698e-05)*(x-12000)**2 &
+ (1.1258162245404875e-01)*(x-12000) +
  (2.0469999999999999e+02)
END IF
  qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<9000))THEN
  qnoselow = (1.1244444444444445e-04)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnoselow = (4.6965079365081316e-10)*(x-9000)**3 +
    (6.3215238095237838e-06)*(x-9000)**2 &
+ (2.1382539682540447e-04)*(x-9000) + (1.0120000000000000
  e+00)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnoselow = (4.6965079365078265e-10)*(x-9500)**3 +
    (7.02600000000000112e-06)*(x-9500)**2 &
+ (6.8875873015872982e-03)*(x-9500) + (2.7580000000000000
  e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN

```

```

      qnoselow =  (-1.0433015873015943e-09)*(x-10000)**3 +
                  (7.7304761904761962e-06)*(x-10000)**2 &
+ (1.4265825396825397e-02)*(x-10000) +
                  (8.016999999999995e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
      qnoselow =  (8.1255555555555568e-10)*(x-11000)**3 +
                  (4.6005714285714336e-06)*(x-11000)**2 &
+ (2.6596873015873008e-02)*(x-11000) +
                  (2.8969999999999999e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
      qnoselow =  (-4.8803968253968532e-10)*(x-12000)**3 +
                  (7.0382380952381006e-06)*(x-12000)**2 &
+ (3.8235682539682542e-02)*(x-12000) +
                  (6.0979999999999997e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
      qnosehigh = (2.1511111111111110e-04)*(x) +
                  (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
      qnosehigh =  (2.1707936507934286e-10)*(x-9000)**3 +
                  (8.8943809523809754e-06)*(x-9000)**2 &
+ (8.0253968253967756e-04)*(x-9000) + (1.9359999999999999
      e+00)
END IF
IF ((x>=9500).AND.(x<10000))THEN

```

```

      qnosehigh = (2.1707936507935640e-10)*(x-9500)**3 +
                  (9.2199999999999998e-06)*(x-9500)**2 &
+ (9.8597301587301598e-03)*(x-9500) + (4.58800000000000001
      e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN
      qnosehigh = (-1.5881587301587148e-09)*(x-10000)**3 +
                  (9.5456190476190344e-06)*(x-10000)**2 &
+ (1.9242539682539677e-02)*(x-10000) +
                  (1.1850000000000000e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
      qnosehigh = (1.15955555555555525e-09)*(x-11000)**3 +
                  (4.7811428571428519e-06)*(x-11000)**2 &
+ (3.3569301587301599e-02)*(x-11000) +
                  (3.9049999999999997e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
      qnosehigh = (-7.0246825396825559e-10)*(x-12000)**3 +
                  (8.2598095238095280e-06)*(x-12000)**2 &
+ (4.6610253968253960e-02)*(x-12000) +
                  (7.85600000000000002e+01)
END IF
      qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
      newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
      rndiff = 1.0-(10.000000-nose)/(10.000000-5.000000)
      IF ((nose>5.0000000).AND.(nose<=10.00)) THEN

```

```

IF ((x>=0).AND.(x<8000))THEN
qnoselow = (3.862499999999998e-04)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
qnoselow = (2.4307352210630859e-09)*(x-8000)**3 +
(-9.5610283159462924e-07)*(x-8000)**2 &
+ (3.0923676105315428e-03)*(x-8000) + (3.089999999999999
e+00)
END IF
IF ((x>=8500).AND.(x<9000))THEN
qnoselow = (2.4307352210630880e-09)*(x-8500)**3 +
(2.69000000000000013e-06)*(x-8500)**2 &
+ (3.9593161947342279e-03)*(x-8500) + (4.7009999999999996
e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnoselow = (1.4422323894684546e-08)*(x-9000)**3 +
(6.3361028315946341e-06)*(x-9000)**2 &
+ (8.4723676105315452e-03)*(x-9000) + (7.6570000000000000
e+00)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnoselow = (-3.0000307998013198e-09)*(x-9500)**3 +
(2.7969588673621479e-05)*(x-9500)**2 &
+ (2.5625213363139589e-02)*(x-9500) + (1.527999999999999
e+01)
END IF

```



```

IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (-3.9243214108296074e-09)*(x-10000)**3 +
    (2.3469542473919509e-05)*(x-10000)**2 &
  + (5.1344778936910078e-02)*(x-10000) +
    (3.4710000000000001e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (8.9252210630903731e-10)*(x-11000)**3 +
    (1.1696578241430681e-05)*(x-11000)**2 &
  + (8.6510899652260276e-02)*(x-11000) +
    (1.0559999999999999e+02)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (8.9252210630896442e-10)*(x-12000)**3 +
    (1.4374144560357698e-05)*(x-12000)**2 &
  + (1.1258162245404875e-01)*(x-12000) +
    (2.0469999999999999e+02)
END IF
if ((x>=0).AND.(x<8000))THEN
  qnosehigh = (7.0037499999999993e-04)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=8000).AND.(x<8500))THEN
  qnosehigh = (3.3314585196224343e-09)*(x-8000)**3 +
    (-9.7518777943365189e-07)*(x-8000)**2 &
  + (5.1707292598112192e-03)*(x-8000) + (5.6029999999999998
    e+00)
END IF

```

```

IF ((x>=8500).AND.(x<9000))THEN
  qnosehigh = (3.3314585196224380e-09)*(x-8500)**3 +
    (4.0219999999999938e-06)*(x-8500)**2 &
+ (6.6941353700943930e-03)*(x-8500) + (8.3610000000000007
  e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (1.7262707401887758e-08)*(x-9000)**3 +
    (9.0191877794336849e-06)*(x-9000)**2 &
+ (1.3214729259811216e-02)*(x-9000) + (1.3130000000000001
  e+01)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (-1.0630288127173397e-08)*(x-9500)**3 +
    (3.4913248882265270e-05)*(x-9500)**2 &
+ (3.5180947590660719e-02)*(x-9500) + (2.4149999999999999
  e+01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (1.3707029309488527e-09)*(x-10000)**3 +
    (1.8967816691505201e-05)*(x-10000)**2 &
+ (6.2121480377545939e-02)*(x-10000) +
    (4.9140000000000001e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (-3.5491480377546478e-09)*(x-11000)**3 +
    (2.3079925484351766e-05)*(x-11000)**2 &

```

```

+ (1.0416922255340290e-01)*(x-11000) +
    (1.3159999999999999e+02)
END IF
IF ((x>=12000).AND.(x<14000))THEN
    qnosehigh = (-3.5491480377545928e-09)*(x-12000)**3 +
        (1.2432481371087934e-05)*(x-12000)**2 &
+ (1.3968162940884249e-01)*(x-12000) +
    (2.55300000000000001e+02)
END IF
    qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<9000))THEN
    qnoselow = (2.1511111111111110e-04)*(x) +
        (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (2.1707936507934286e-10)*(x-9000)**3 +
        (8.8943809523809754e-06)*(x-9000)**2 &
+ (8.0253968253967756e-04)*(x-9000) + (1.9359999999999999
    e+00)
END IF
IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (2.1707936507935640e-10)*(x-9500)**3 +
        (9.2199999999999998e-06)*(x-9500)**2 &
+ (9.8597301587301598e-03)*(x-9500) + (4.58800000000000001
    e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN

```

```

qnoselow = (-1.5881587301587148e-09)*(x-10000)**3 +
(9.5456190476190344e-06)*(x-10000)**2 &
+ (1.9242539682539677e-02)*(x-10000) +
(1.1850000000000000e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnoselow = (1.1595555555555525e-09)*(x-11000)**3 +
(4.7811428571428519e-06)*(x-11000)**2 &
+ (3.3569301587301599e-02)*(x-11000) +
(3.9049999999999997e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
qnoselow = (-7.0246825396825559e-10)*(x-12000)**3 +
(8.2598095238095280e-06)*(x-12000)**2 &
+ (4.6610253968253960e-02)*(x-12000) +
(7.8560000000000002e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
qnosehigh = (3.8166666666666666e-04)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnosehigh = (-2.1270370370370099e-09)*(x-9000)**3 +
(1.4720555555555513e-05)*(x-9000)**2 &
+ (8.5148148148149613e-04)*(x-9000) + (3.4350000000000001
e+00)
END IF
IF ((x>=9500).AND.(x<10000))THEN

```

```

      qnosehigh = (-2.1270370370370372e-09)*(x-9500)**3 +
      (1.15300000000000012e-05)*(x-9500)**2 &
+ (1.3976759259259251e-02)*(x-9500) + (7.2750000000000004
      e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN
      qnosehigh = (8.2907407407407901e-10)*(x-10000)**3 +
      (8.33944444444444340e-06)*(x-10000)**2 &
+ (2.3911481481481485e-02)*(x-10000) +
      (1.6879999999999999e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
      qnosehigh = (-2.3642592592592600e-09)*(x-11000)**3 +
      (1.0826666666666671e-05)*(x-11000)**2 &
+ (4.3077592592592591e-02)*(x-11000) +
      (4.99600000000000001e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
      qnosehigh = (8.6018518518521542e-11)*(x-12000)**3 +
      (3.7338888888888843e-06)*(x-12000)**2 &
+ (5.7638148148148154e-02)*(x-12000) +
      (1.01500000000000000e+02)
END IF
      qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
      newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if

END IF !density if

```

!!!!!!!!!!!!!!!!!!!!NEW ALTITUDE RANGE!!!!!!!!!!!!!!!!!!!!

```
IF ((rho <= 0.0013167000) .AND. (rho >= 0.0006313700)) THEN
  rhodiff = (0.0013167000 - rho) / (0.0013167000 - 0.0006313700)
  rndiff = 1.0 - (0.100000 - nose) / (0.100000 - 0.049999)
  IF ((nose > 0.049999) .AND. (nose <= 0.10)) THEN
    IF ((x >= 0) .AND. (x < 9000)) THEN
      qnoselow = (6.0855555555555558e-06) * (x) +
        (0.0000000000000000e+00)
    END IF
    IF ((x >= 9000) .AND. (x < 9500)) THEN
      qnoselow = (6.1879650793650763e-10) * (x - 9000)**3 +
        (4.6534523809523827e-07) * (x - 9000)**2 &
        + (3.5088253968253993e-05) * (x - 9000) + (5.476999999999999
          e-02)
    END IF
    IF ((x >= 9500) .AND. (x < 10000)) THEN
      qnoselow = (6.1879650793650639e-10) * (x - 9500)**3 +
        (1.39354000000000006e-06) * (x - 9500)**2 &
        + (9.6453087301587304e-04) * (x - 9500) + (2.6600000000000001
          e-01)
    END IF
    IF ((x >= 10000) .AND. (x < 11000)) THEN
      qnoselow = (8.6096984126984615e-11) * (x - 10000)**3 +
        (2.3217347619047624e-06) * (x - 10000)**2 &
        + (2.8221682539682534e-03) * (x - 10000) +
        (1.1739999999999999e+00)
```

```

END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (1.3204555555555699e-10)*(x-11000)**3 +
    (2.5800257142857103e-06)*(x-11000)**2 &
+ (7.7239287301587318e-03)*(x-11000) +
    (6.4039999999999999e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (8.3139603174605556e-11)*(x-12000)**3 +
    (2.9761623809523766e-06)*(x-12000)**2 &
+ (1.3280116825396824e-02)*(x-12000) +
    (1.6840000000000000e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnoselow = (8.3139603174600050e-11)*(x-14000)**3 +
    (3.47500000000000023e-06)*(x-14000)**2 &
+ (2.6182441587301597e-02)*(x-14000) +
    (5.5969999999999999e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
  qnosehigh = (1.0395555555555555e-05)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (6.8667788359788124e-10)*(x-9000)**3 +
    (9.9310317460317866e-07)*(x-9000)**2 &
+ (-3.9341058201059818e-05)*(x-9000) +
    (9.3560000000000004e-02)

```

```

END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (6.8667788359788082e-10)*(x-9500)**3 +
    (2.02312000000000007e-06)*(x-9500)**2 &
+ (1.4687705291005298e-03)*(x-9500) + (4.0799999999999997
  e-01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (2.9964232804234473e-11)*(x-10000)**3 +
    (3.0531368253968245e-06)*(x-10000)**2 &
+ (4.0068989417989410e-03)*(x-10000) +
    (1.7340000000000000e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (9.9905185185187399e-11)*(x-11000)**3 +
    (3.1430295238095186e-06)*(x-11000)**2 &
+ (1.0203065291005294e-02)*(x-11000) +
    (8.8239999999999998e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (6.3917486772489647e-11)*(x-12000)**3 +
    (3.4427450793650740e-06)*(x-12000)**2 &
+ (1.6788839894179893e-02)*(x-12000) +
    (2.2270000000000000e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (6.3917486772485201e-11)*(x-14000)**3 +
    (3.8262499999999966e-06)*(x-14000)**2 &

```



```

+ (3.1326830052910065e-02)*(x-14000) +
(7.0129999999999995e+01)
END IF

qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<9000))THEN
qnoselow = (1.6955555555555555e-06)*(x) +
(0.0000000000000000e+00)
END IF

IF ((x>=9000).AND.(x<9500))THEN
qnoselow = (2.0969739130434884e-10)*(x-9000)**3 +
(1.9549391304347674e-07)*(x-9000)**2 &
+ (1.2448695652174435e-05)*(x-9000) + (1.5259999999999999
e-02)
END IF

IF ((x>=9500).AND.(x<10000))THEN
qnoselow = (2.0969739130434809e-10)*(x-9500)**3 +
(5.10040000000000046e-07)*(x-9500)**2 &
+ (3.6521565217391281e-04)*(x-9500) + (9.6570000000000003
e-02)
END IF

IF ((x>=10000).AND.(x<11000))THEN
qnoselow = (6.4985217391303241e-11)*(x-10000)**3 +
(8.2458608695652238e-07)*(x-10000)**2 &
+ (1.0325286956521743e-03)*(x-10000) +
(4.3290000000000001e-01)
END IF

IF ((x>=11000).AND.(x<12000))THEN

```

```

      qnoselow = (4.5801739130434293e-11)*(x-11000)**3 +
      (1.0195417391304361e-06)*(x-11000)**2 &
+ (2.8766565217391289e-03)*(x-11000) +
      (2.3550000000000000e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
      qnoselow = (5.6115217391305519e-11)*(x-12000)**3 +
      (1.1569469565217371e-06)*(x-12000)**2 &
+ (5.0531452173913039e-03)*(x-12000) +
      (6.2969999999999997e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
      qnoselow = (-4.0398043478262290e-11)*(x-14000)**3 +
      (1.4936382608695658e-06)*(x-14000)**2 &
+ (1.0354315652173918e-02)*(x-14000) +
      (2.1480000000000000e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
      qnosehigh = (3.0111111111111108e-06)*(x) +
      (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
      qnosehigh = (2.8076028985507420e-10)*(x-9000)**3 +
      (3.4665956521738907e-07)*(x-9000)**2 &
+ (6.4801449275369020e-06)*(x-9000) + (2.7099999999999999
      e-02)
END IF
IF ((x>=9500).AND.(x<10000))THEN

```

```

qnosehigh = (2.8076028985507126e-10)*(x-9500)**3 +
              (7.67800000000000136e-07)*(x-9500)**2 &
+ (5.6370992753623165e-04)*(x-9500) + (1.52100000000000001
              e-01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnosehigh = (4.5979420289853994e-11)*(x-10000)**3 +
              (1.1889404347826094e-06)*(x-10000)**2 &
+ (1.5420801449275365e-03)*(x-10000) +
              (6.61000000000000003e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnosehigh = (6.3222028985507541e-11)*(x-11000)**3 +
              (1.3268786956521739e-06)*(x-11000)**2 &
+ (4.0578992753623174e-03)*(x-11000) +
              (3.43800000000000002e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
qnosehigh = (5.9396920289854532e-11)*(x-12000)**3 +
              (1.5165447826086972e-06)*(x-12000)**2 &
+ (6.9013227536231878e-03)*(x-12000) +
              (8.8859999999999992e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
qnosehigh = (-1.9029384057970108e-11)*(x-14000)**3 +
              (1.8729263043478250e-06)*(x-14000)**2 &
+ (1.3680264927536231e-02)*(x-14000) +
              (2.92300000000000000e+01)

```

```

END IF

  qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
  newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
rndiff = 1.0-(0.230000-nose)/(0.230000-0.100000)
IF ((nose>0.1000000).AND.(nose<=0.23)) THEN
  IF ((x>=0).AND.(x<9000))THEN
    qnoselow = (1.0395555555555555e-05)*(x) +
      (0.0000000000000000e+00)
  END IF
  IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (6.8667788359788124e-10)*(x-9000)**3 +
      (9.9310317460317866e-07)*(x-9000)**2 &
      + (-3.9341058201059818e-05)*(x-9000) +
      (9.3560000000000004e-02)
  END IF
  IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (6.8667788359788082e-10)*(x-9500)**3 +
      (2.0231200000000007e-06)*(x-9500)**2 &
      + (1.4687705291005298e-03)*(x-9500) + (4.0799999999999997
        e-01)
  END IF
  IF ((x>=10000).AND.(x<11000))THEN
    qnoselow = (2.9964232804234473e-11)*(x-10000)**3 +
      (3.0531368253968245e-06)*(x-10000)**2 &
      + (4.0068989417989410e-03)*(x-10000) +
      (1.7340000000000000e+00)
  END IF

```

```

IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (9.9905185185187399e-11)*(x-11000)**3 +
    (3.1430295238095186e-06)*(x-11000)**2 &
  + (1.0203065291005294e-02)*(x-11000) +
    (8.8239999999999998e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (6.3917486772489647e-11)*(x-12000)**3 +
    (3.4427450793650740e-06)*(x-12000)**2 &
  + (1.6788839894179893e-02)*(x-12000) +
    (2.2270000000000000e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnoselow = (6.3917486772485201e-11)*(x-14000)**3 +
    (3.8262499999999966e-06)*(x-14000)**2 &
  + (3.1326830052910065e-02)*(x-14000) +
    (7.0129999999999995e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
  qnosehigh = (1.9344444444444446e-05)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (6.8506772486772668e-10)*(x-9000)**3 +
    (1.9473984126984121e-06)*(x-9000)**2 &
  + (-1.3256613756613777e-04)*(x-9000) +
    (1.7410000000000000e-01)
END IF

```

```

IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (6.8506772486773092e-10)*(x-9500)**3 +
    (2.9749999999999969e-06)*(x-9500)**2 &
+ (2.3286330687830692e-03)*(x-9500) + (6.8030000000000002
  e-01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (-1.0403544973545408e-10)*(x-10000)**3 +
    (4.0026015873015907e-06)*(x-10000)**2 &
+ (5.8174338624338642e-03)*(x-10000) +
    (2.6739999999999999e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (-1.1025925925925492e-11)*(x-11000)**3 +
    (3.6904952380952395e-06)*(x-11000)**2 &
+ (1.3510530687830683e-02)*(x-11000) +
    (1.23900000000000001e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (1.6680423280421707e-11)*(x-12000)**3 +
    (3.6574174603174647e-06)*(x-12000)**2 &
+ (2.0858443386243385e-02)*(x-12000) +
    (2.9579999999999998e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (1.6680423280424247e-11)*(x-14000)**3 +
    (3.7574999999999975e-06)*(x-14000)**2 &

```

```

+ (3.5688278306878304e-02)*(x-14000) +
      (8.6060000000000002e+01)
END IF

qnosel = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<9000))THEN
qnoselow = (3.0111111111111108e-06)*(x) +
      (0.0000000000000000e+00)
END IF

IF ((x>=9000).AND.(x<9500))THEN
qnoselow = (2.8076028985507420e-10)*(x-9000)**3 +
      (3.4665956521738907e-07)*(x-9000)**2 &
+ (6.4801449275369020e-06)*(x-9000) + (2.7099999999999999
      e-02)
END IF

IF ((x>=9500).AND.(x<10000))THEN
qnoselow = (2.8076028985507126e-10)*(x-9500)**3 +
      (7.678000000000000136e-07)*(x-9500)**2 &
+ (5.6370992753623165e-04)*(x-9500) + (1.52100000000000001
      e-01)
END IF

IF ((x>=10000).AND.(x<11000))THEN
qnoselow = (4.5979420289853994e-11)*(x-10000)**3 +
      (1.1889404347826094e-06)*(x-10000)**2 &
+ (1.5420801449275365e-03)*(x-10000) +
      (6.61000000000000003e-01)
END IF

IF ((x>=11000).AND.(x<12000))THEN

```

```

      qnoselow = (6.3222028985507541e-11)*(x-11000)**3 +
      (1.3268786956521739e-06)*(x-11000)**2 &
+ (4.0578992753623174e-03)*(x-11000) +
      (3.4380000000000002e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
      qnoselow = (5.9396920289854532e-11)*(x-12000)**3 +
      (1.5165447826086972e-06)*(x-12000)**2 &
+ (6.9013227536231878e-03)*(x-12000) +
      (8.8859999999999992e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
      qnoselow = (-1.9029384057970108e-11)*(x-14000)**3 +
      (1.8729263043478250e-06)*(x-14000)**2 &
+ (1.3680264927536231e-02)*(x-14000) +
      (2.9230000000000000e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
      qnosehigh = (5.9066666666666663e-06)*(x) +
      (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
      qnosehigh = (3.0399669565217168e-10)*(x-9000)**3 +
      (7.4552495652174201e-07)*(x-9000)**2 &
+ (-3.8681652173913951e-05)*(x-9000) +
      (5.3159999999999999e-02)
END IF
IF ((x>=9500).AND.(x<10000))THEN

```



```

qnosehigh = (3.0399669565217256e-10)*(x-9500)**3 +
(1.20152000000000002e-06)*(x-9500)**2 &
+ (9.3484082608695691e-04)*(x-9500) + (2.5819999999999999
e-01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnosehigh = (4.1266086956516973e-12)*(x-10000)**3 +
(1.6575150434782620e-06)*(x-10000)**2 &
+ (2.3643583478260865e-03)*(x-10000) +
(1.06400000000000001e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnosehigh = (4.8336869565219628e-11)*(x-11000)**3 +
(1.6698948695652149e-06)*(x-11000)**2 &
+ (5.6917682608695654e-03)*(x-11000) +
(5.0899999999999999e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
qnosehigh = (3.4655108695652446e-11)*(x-12000)**3 +
(1.8149054782608679e-06)*(x-12000)**2 &
+ (9.1765686086956542e-03)*(x-12000) +
(1.25000000000000000e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
qnosehigh = (-1.8181021739132138e-11)*(x-14000)**3 +
(2.0228361304347883e-06)*(x-14000)**2 &
+ (1.6852051826086956e-02)*(x-14000) +
(3.83900000000000001e+01)

```

```

END IF

  qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
  newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
rndiff = 1.0-(0.500000-nose)/(0.500000-0.230000)
IF ((nose>0.230000).AND.(nose<=0.50)) THEN
  IF ((x>=0).AND.(x<9000))THEN
    qnoselow = (1.93444444444444446e-05)*(x) +
      (0.00000000000000000e+00)
  END IF
  IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (6.8506772486772668e-10)*(x-9000)**3 +
      (1.9473984126984121e-06)*(x-9000)**2 &
      + (-1.3256613756613777e-04)*(x-9000) +
      (1.74100000000000000e-01)
  END IF
  IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (6.8506772486773092e-10)*(x-9500)**3 +
      (2.9749999999999969e-06)*(x-9500)**2 &
      + (2.3286330687830692e-03)*(x-9500) + (6.80300000000000002
        e-01)
  END IF
  IF ((x>=10000).AND.(x<11000))THEN
    qnoselow = (-1.0403544973545408e-10)*(x-10000)**3 +
      (4.0026015873015907e-06)*(x-10000)**2 &
      + (5.8174338624338642e-03)*(x-10000) +
      (2.6739999999999999e+00)
  END IF

```

```

IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (-1.1025925925925492e-11)*(x-11000)**3 +
    (3.6904952380952395e-06)*(x-11000)**2 &
  + (1.3510530687830683e-02)*(x-11000) +
    (1.23900000000000001e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (1.6680423280421707e-11)*(x-12000)**3 +
    (3.6574174603174647e-06)*(x-12000)**2 &
  + (2.0858443386243385e-02)*(x-12000) +
    (2.9579999999999998e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnoselow = (1.6680423280424247e-11)*(x-14000)**3 +
    (3.7574999999999975e-06)*(x-14000)**2 &
  + (3.5688278306878304e-02)*(x-14000) +
    (8.60600000000000002e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
  qnosehigh = (3.4188888888888885e-05)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (6.4466878306878408e-10)*(x-9000)**3 +
    (3.0763968253968224e-06)*(x-9000)**2 &
  + (-1.4676560846560704e-04)*(x-9000) +
    (3.0769999999999997e-01)
END IF

```

```

IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (6.4466878306877137e-10)*(x-9500)**3 +
    (4.0434000000000086e-06)*(x-9500)**2 &
+ (3.4131328042328033e-03)*(x-9500) + (1.0840000000000001
    e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (-3.3243756613756546e-10)*(x-10000)**3 +
    (5.0104031746031758e-06)*(x-10000)**2 &
+ (7.9400343915343903e-03)*(x-10000) +
    (3.8820000000000001e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (-2.6618518518515803e-11)*(x-11000)**3 +
    (4.0130904761904735e-06)*(x-11000)**2 &
+ (1.6963528042328046e-02)*(x-11000) +
    (1.6500000000000000e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (-6.2830820105814792e-11)*(x-12000)**3 +
    (3.9332349206349058e-06)*(x-12000)**2 &
+ (2.4909853439153445e-02)*(x-12000) +
    (3.7450000000000003e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (-6.2830820105826657e-11)*(x-14000)**3 +
    (3.5562500000000068e-06)*(x-14000)**2 &

```

```

+ (3.9888823280423293e-02)*(x-14000) +
    (1.0250000000000000e+02)
END IF

qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<9000))THEN
qnoselow = (5.9066666666666663e-06)*(x) +
    (0.0000000000000000e+00)
END IF

IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (3.0399669565217168e-10)*(x-9000)**3 +
        (7.4552495652174201e-07)*(x-9000)**2 &
+ (-3.8681652173913951e-05)*(x-9000) +
    (5.3159999999999999e-02)
END IF

IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (3.0399669565217256e-10)*(x-9500)**3 +
        (1.2015200000000002e-06)*(x-9500)**2 &
+ (9.3484082608695691e-04)*(x-9500) + (2.5819999999999999
    e-01)
END IF

IF ((x>=10000).AND.(x<11000))THEN
    qnoselow = (4.1266086956516973e-12)*(x-10000)**3 +
        (1.6575150434782620e-06)*(x-10000)**2 &
+ (2.3643583478260865e-03)*(x-10000) +
    (1.0640000000000001e+00)
END IF

IF ((x>=11000).AND.(x<12000))THEN

```

```

      qnoselow = (4.8336869565219628e-11)*(x-11000)**3 +
      (1.6698948695652149e-06)*(x-11000)**2 &
+ (5.6917682608695654e-03)*(x-11000) +
      (5.0899999999999999e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
      qnoselow = (3.4655108695652446e-11)*(x-12000)**3 +
      (1.8149054782608679e-06)*(x-12000)**2 &
+ (9.1765686086956542e-03)*(x-12000) +
      (1.2500000000000000e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
      qnoselow = (-1.8181021739132138e-11)*(x-14000)**3 +
      (2.0228361304347883e-06)*(x-14000)**2 &
+ (1.6852051826086956e-02)*(x-14000) +
      (3.8390000000000000e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
      qnosehigh = (1.0795555555555555e-05)*(x) +
      (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
      qnosehigh = (2.4984886956521521e-10)*(x-9000)**3 +
      (1.3567466956521770e-06)*(x-9000)**2 &
+ (-1.0575556521739226e-04)*(x-9000) +
      (9.7159999999999996e-02)
END IF
IF ((x>=9500).AND.(x<10000))THEN

```

```

qnosehigh = (2.4984886956522110e-10)*(x-9500)**3 +
(1.7315199999999970e-06)*(x-9500)**2 &
+ (1.4383777826086962e-03)*(x-9500) + (4.1470000000000001
e-01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnosehigh = (-5.5577739130435639e-11)*(x-10000)**3 +
(2.1062933043478272e-06)*(x-10000)**2 &
+ (3.3572844347826091e-03)*(x-10000) +
(1.59800000000000001e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnosehigh = (1.3020869565217379e-12)*(x-11000)**3 +
(1.9395600869565237e-06)*(x-11000)**2 &
+ (7.4031378260869564e-03)*(x-11000) +
(7.00600000000000002e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
qnosehigh = (-7.7423913043301298e-13)*(x-12000)**3 +
(1.9434663478260804e-06)*(x-12000)**2 &
+ (1.1286164260869569e-02)*(x-12000) +
(1.63500000000000001e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
qnosehigh = (-4.7095152173914534e-11)*(x-14000)**3 +
(1.9388209130434836e-06)*(x-14000)**2 &
+ (1.9050738782608694e-02)*(x-14000) +
(4.6689999999999998e+01)

```

```

END IF

  qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
  newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
rndiff = 1.0-(1.000000-nose)/(1.000000-0.500000)
IF ((nose>0.5000000).AND.(nose<=1.00)) THEN
  IF ((x>=0).AND.(x<9000))THEN
    qnoselow = (3.4188888888888885e-05)*(x) +
      (0.0000000000000000e+00)
  END IF
  IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (6.4466878306878408e-10)*(x-9000)**3 +
      (3.0763968253968224e-06)*(x-9000)**2 &
      + (-1.4676560846560704e-04)*(x-9000) +
      (3.0769999999999997e-01)
  END IF
  IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (6.4466878306877137e-10)*(x-9500)**3 +
      (4.04340000000000086e-06)*(x-9500)**2 &
      + (3.4131328042328033e-03)*(x-9500) + (1.08400000000000001
      e+00)
  END IF
  IF ((x>=10000).AND.(x<11000))THEN
    qnoselow = (-3.3243756613756546e-10)*(x-10000)**3 +
      (5.0104031746031758e-06)*(x-10000)**2 &
      + (7.9400343915343903e-03)*(x-10000) +
      (3.88200000000000001e+00)
  END IF

```



```

IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (-2.6618518518515803e-11)*(x-11000)**3 +
    (4.0130904761904735e-06)*(x-11000)**2 &
  + (1.6963528042328046e-02)*(x-11000) +
    (1.6500000000000000e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (-6.2830820105814792e-11)*(x-12000)**3 +
    (3.9332349206349058e-06)*(x-12000)**2 &
  + (2.4909853439153445e-02)*(x-12000) +
    (3.7450000000000003e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnoselow = (-6.2830820105826657e-11)*(x-14000)**3 +
    (3.55625000000000068e-06)*(x-14000)**2 &
  + (3.9888823280423293e-02)*(x-14000) +
    (1.0250000000000000e+02)
END IF
if ((x>=0).AND.(x<9000))THEN
  qnosehigh = (5.7700000000000000e-05)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (6.04844444444445382e-10)*(x-9000)**3 +
    (4.30933333333333228e-06)*(x-9000)**2 &
  + (-5.64777777777775631e-05)*(x-9000) +
    (5.1929999999999998e-01)
END IF

```

```

IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (6.0484444444443180e-10)*(x-9500)**3 +
    (5.2166000000000078e-06)*(x-9500)**2 &
+ (4.7064888888888878e-03)*(x-9500) + (1.6439999999999999
  e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (-6.8758888888889111e-10)*(x-10000)**3 +
    (6.1238666666666725e-06)*(x-10000)**2 &
+ (1.0376722222222219e-02)*(x-10000) +
    (5.3769999999999998e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (5.77211111111110053e-10)*(x-11000)**3 +
    (4.0611000000000084e-06)*(x-11000)**2 &
+ (2.0561688888888891e-02)*(x-11000) +
    (2.1190000000000001e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (-3.2399722222222003e-10)*(x-12000)**3 +
    (5.7927333333333339e-06)*(x-12000)**2 &
+ (3.0415522222222210e-02)*(x-12000) +
    (4.6390000000000001e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (-3.2399722222223146e-10)*(x-14000)**3 +
    (3.8487500000000121e-06)*(x-14000)**2 &

```

```

+ (4.9698488888888906e-02)*(x-14000) +
    (1.2780000000000000e+02)
END IF

qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<9000))THEN
qnoselow = (1.0795555555555555e-05)*(x) +
    (0.0000000000000000e+00)
END IF

IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (2.4984886956521521e-10)*(x-9000)**3 +
        (1.3567466956521770e-06)*(x-9000)**2 &
+ (-1.0575556521739226e-04)*(x-9000) +
    (9.7159999999999996e-02)
END IF

IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (2.4984886956522110e-10)*(x-9500)**3 +
        (1.7315199999999970e-06)*(x-9500)**2 &
+ (1.4383777826086962e-03)*(x-9500) + (4.1470000000000001
    e-01)
END IF

IF ((x>=10000).AND.(x<11000))THEN
    qnoselow = (-5.5577739130435639e-11)*(x-10000)**3 +
        (2.1062933043478272e-06)*(x-10000)**2 &
+ (3.3572844347826091e-03)*(x-10000) +
    (1.5980000000000001e+00)
END IF

IF ((x>=11000).AND.(x<12000))THEN

```

```

      qnoselow = (1.3020869565217379e-12)*(x-11000)**3 +
      (1.9395600869565237e-06)*(x-11000)**2 &
+ (7.4031378260869564e-03)*(x-11000) +
      (7.0060000000000002e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
      qnoselow = (-7.7423913043301298e-13)*(x-12000)**3 +
      (1.9434663478260804e-06)*(x-12000)**2 &
+ (1.1286164260869569e-02)*(x-12000) +
      (1.6350000000000001e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
      qnoselow = (-4.7095152173914534e-11)*(x-14000)**3 +
      (1.9388209130434836e-06)*(x-14000)**2 &
+ (1.9050738782608694e-02)*(x-14000) +
      (4.6689999999999998e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
      qnosehigh = (1.8133333333333335e-05)*(x) +
      (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
      qnosehigh = (1.0391971014492448e-10)*(x-9000)**3 +
      (2.1413204347826131e-06)*(x-9000)**2 &
+ (-1.6944014492753778e-04)*(x-9000) +
      (1.6320000000000001e-01)
END IF
IF ((x>=9500).AND.(x<10000))THEN

```

```

qnosehigh = (1.0391971014493549e-10)*(x-9500)**3 +
(2.2971999999999939e-06)*(x-9500)**2 &
+ (2.0498200724637689e-03)*(x-9500) + (6.2680000000000002
e-01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnosehigh = (-5.8039420289856761e-11)*(x-10000)**3 +
(2.4530795652173920e-06)*(x-10000)**2 &
+ (4.4249598550724642e-03)*(x-10000) +
(2.2389999999999999e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnosehigh = (-2.5496202898550761e-10)*(x-11000)**3 +
(2.2789613043478294e-06)*(x-11000)**2 &
+ (9.1570007246376782e-03)*(x-11000) +
(9.0589999999999993e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
qnosehigh = (2.8420307971014545e-10)*(x-12000)**3 +
(1.5140752173913028e-06)*(x-12000)**2 &
+ (1.2950037246376814e-02)*(x-12000) +
(2.0239999999999998e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
qnosehigh = (-4.8509061594202872e-10)*(x-14000)**3 +
(3.2192936956521708e-06)*(x-14000)**2 &
+ (2.2416775072463771e-02)*(x-14000) +
(5.4469999999999999e+01)

```

```

END IF

  qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
  newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
rndiff = 1.0-(2.300000-nose)/(2.300000-1.000000)
IF ((nose>1.0000000).AND.(nose<=2.30)) THEN
  IF ((x>=0).AND.(x<9000))THEN
    qnoselow = (5.7700000000000000e-05)*(x) +
      (0.0000000000000000e+00)
  END IF
  IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (6.04844444444445382e-10)*(x-9000)**3 +
      (4.3093333333333228e-06)*(x-9000)**2 &
      + (-5.64777777777775631e-05)*(x-9000) +
      (5.1929999999999998e-01)
  END IF
  IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (6.04844444444443180e-10)*(x-9500)**3 +
      (5.2166000000000078e-06)*(x-9500)**2 &
      + (4.7064888888888878e-03)*(x-9500) + (1.6439999999999999
      e+00)
  END IF
  IF ((x>=10000).AND.(x<11000))THEN
    qnoselow = (-6.8758888888889111e-10)*(x-10000)**3 +
      (6.1238666666666725e-06)*(x-10000)**2 &
      + (1.0376722222222219e-02)*(x-10000) +
      (5.3769999999999998e+00)
  END IF

```

```

IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (5.77211111111110053e-10)*(x-11000)**3 +
    (4.0611000000000084e-06)*(x-11000)**2 &
+ (2.0561688888888891e-02)*(x-11000) +
    (2.1190000000000001e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (-3.2399722222222003e-10)*(x-12000)**3 +
    (5.792733333333339e-06)*(x-12000)**2 &
+ (3.0415522222222210e-02)*(x-12000) +
    (4.6390000000000001e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnoselow = (-3.2399722222223146e-10)*(x-14000)**3 +
    (3.8487500000000121e-06)*(x-14000)**2 &
+ (4.9698488888888906e-02)*(x-14000) +
    (1.2780000000000000e+02)
END IF
if ((x>=0).AND.(x<9000))THEN
  qnosehigh = (1.1244444444444445e-04)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (4.6965079365081316e-10)*(x-9000)**3 +
    (6.3215238095237838e-06)*(x-9000)**2 &
+ (2.1382539682540447e-04)*(x-9000) + (1.0120000000000000
    e+00)
END IF

```

```

IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (4.6965079365078265e-10)*(x-9500)**3 +
    (7.02600000000000112e-06)*(x-9500)**2 &
+ (6.8875873015872982e-03)*(x-9500) + (2.7580000000000000
    e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (-1.0433015873015943e-09)*(x-10000)**3 +
    (7.7304761904761962e-06)*(x-10000)**2 &
+ (1.4265825396825397e-02)*(x-10000) +
    (8.0169999999999995e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (8.1255555555555568e-10)*(x-11000)**3 +
    (4.6005714285714336e-06)*(x-11000)**2 &
+ (2.6596873015873008e-02)*(x-11000) +
    (2.8969999999999999e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (-4.8803968253968532e-10)*(x-12000)**3 +
    (7.0382380952381006e-06)*(x-12000)**2 &
+ (3.8235682539682542e-02)*(x-12000) +
    (6.0979999999999997e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (-4.8803968253967136e-10)*(x-14000)**3 +
    (4.10999999999999878e-06)*(x-14000)**2 &

```



```

+ (6.0532158730158721e-02)*(x-14000) +
    (1.6169999999999999e+02)
END IF

qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<9000))THEN
qnoselow = (1.8133333333333335e-05)*(x) +
    (0.0000000000000000e+00)
END IF

IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (1.0391971014492448e-10)*(x-9000)**3 +
        (2.1413204347826131e-06)*(x-9000)**2 &
+ (-1.6944014492753778e-04)*(x-9000) +
    (1.63200000000000001e-01)
END IF

IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (1.0391971014493549e-10)*(x-9500)**3 +
        (2.2971999999999939e-06)*(x-9500)**2 &
+ (2.0498200724637689e-03)*(x-9500) + (6.26800000000000002
    e-01)
END IF

IF ((x>=10000).AND.(x<11000))THEN
    qnoselow = (-5.8039420289856761e-11)*(x-10000)**3 +
        (2.4530795652173920e-06)*(x-10000)**2 &
+ (4.4249598550724642e-03)*(x-10000) +
    (2.2389999999999999e+00)
END IF

IF ((x>=11000).AND.(x<12000))THEN

```

```

      qnoselow =  (-2.5496202898550761e-10)*(x-11000)**3 +
                  (2.2789613043478294e-06)*(x-11000)**2 &
+   (9.1570007246376782e-03)*(x-11000) +
                  (9.0589999999999993e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
      qnoselow =  (2.8420307971014545e-10)*(x-12000)**3 +
                  (1.5140752173913028e-06)*(x-12000)**2 &
+   (1.2950037246376814e-02)*(x-12000) +
                  (2.0239999999999998e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
      qnoselow =  (-4.8509061594202872e-10)*(x-14000)**3 +
                  (3.2192936956521708e-06)*(x-14000)**2 &
+   (2.2416775072463771e-02)*(x-14000) +
                  (5.4469999999999999e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
      qnosehigh = (3.4022222222222226e-05)*(x) +
                  (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
      qnosehigh =  (1.2433913043485579e-11)*(x-9000)**3 +
                  (3.0637491304347728e-06)*(x-9000)**2 &
+   (-8.9383043478258098e-05)*(x-9000) +
                  (3.0620000000000003e-01)
END IF
IF ((x>=9500).AND.(x<10000))THEN

```

```

qnosehigh = (1.2433913043475415e-11)*(x-9500)**3 +
(3.08240000000000058e-06)*(x-9500)**2 &
+ (2.9836915217391290e-03)*(x-9500) + (1.0289999999999999
e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnosehigh = (-3.1946782608695561e-10)*(x-10000)**3 +
(3.1010508695652130e-06)*(x-10000)**2 &
+ (6.0754169565217413e-03)*(x-10000) +
(3.29300000000000001e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnosehigh = (3.0823739130434582e-10)*(x-11000)**3 +
(2.1426473913043547e-06)*(x-11000)**2 &
+ (1.1319115217391301e-02)*(x-11000) +
(1.21500000000000000e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
qnosehigh = (-2.0471032608695631e-10)*(x-12000)**3 +
(3.0673595652173887e-06)*(x-12000)**2 &
+ (1.6529122173913047e-02)*(x-12000) +
(2.59200000000000002e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
qnosehigh = (-6.8807934782608647e-11)*(x-14000)**3 +
(1.8390976086956524e-06)*(x-14000)**2 &
+ (2.6342036521739127e-02)*(x-14000) +
(6.9609999999999999e+01)

```

```

END IF

  qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
  newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
rndiff = 1.0-(5.000000-nose)/(5.000000-2.300000)
IF ((nose>2.300000).AND.(nose<=5.00)) THEN
  IF ((x>=0).AND.(x<9000))THEN
    qnoselow = (1.1244444444444445e-04)*(x) +
      (0.0000000000000000e+00)
  END IF
  IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (4.6965079365081316e-10)*(x-9000)**3 +
      (6.3215238095237838e-06)*(x-9000)**2 &
      + (2.1382539682540447e-04)*(x-9000) + (1.0120000000000000
        e+00)
  END IF
  IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (4.6965079365078265e-10)*(x-9500)**3 +
      (7.02600000000000112e-06)*(x-9500)**2 &
      + (6.8875873015872982e-03)*(x-9500) + (2.7580000000000000
        e+00)
  END IF
  IF ((x>=10000).AND.(x<11000))THEN
    qnoselow = (-1.0433015873015943e-09)*(x-10000)**3 +
      (7.7304761904761962e-06)*(x-10000)**2 &
      + (1.4265825396825397e-02)*(x-10000) +
      (8.0169999999999995e+00)
  END IF

```

```

IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (8.1255555555555568e-10)*(x-11000)**3 +
    (4.6005714285714336e-06)*(x-11000)**2 &
  + (2.6596873015873008e-02)*(x-11000) +
    (2.8969999999999999e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (-4.8803968253968532e-10)*(x-12000)**3 +
    (7.0382380952381006e-06)*(x-12000)**2 &
  + (3.8235682539682542e-02)*(x-12000) +
    (6.0979999999999997e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnoselow = (-4.8803968253967136e-10)*(x-14000)**3 +
    (4.10999999999999878e-06)*(x-14000)**2 &
  + (6.0532158730158721e-02)*(x-14000) +
    (1.6169999999999999e+02)
END IF
if ((x>=0).AND.(x<9000))THEN
  qnosehigh = (2.1511111111111110e-04)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (2.1707936507934286e-10)*(x-9000)**3 +
    (8.8943809523809754e-06)*(x-9000)**2 &
  + (8.0253968253967756e-04)*(x-9000) + (1.9359999999999999
    e+00)
END IF

```

```

IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (2.1707936507935640e-10)*(x-9500)**3 +
    (9.2199999999999998e-06)*(x-9500)**2 &
+ (9.8597301587301598e-03)*(x-9500) + (4.5880000000000001
  e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (-1.5881587301587148e-09)*(x-10000)**3 +
    (9.5456190476190344e-06)*(x-10000)**2 &
+ (1.9242539682539677e-02)*(x-10000) +
    (1.1850000000000000e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (1.1595555555555525e-09)*(x-11000)**3 +
    (4.7811428571428519e-06)*(x-11000)**2 &
+ (3.3569301587301599e-02)*(x-11000) +
    (3.9049999999999997e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (-7.0246825396825559e-10)*(x-12000)**3 +
    (8.2598095238095280e-06)*(x-12000)**2 &
+ (4.6610253968253960e-02)*(x-12000) +
    (7.8560000000000002e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (-7.0246825396825042e-10)*(x-14000)**3 +
    (4.0450000000000002e-06)*(x-14000)**2 &

```

```

+ (7.1219873015873000e-02)*(x-14000) +
(1.9919999999999999e+02)
END IF

qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<9000))THEN
qnoselow = (3.4022222222222226e-05)*(x) +
(0.0000000000000000e+00)
END IF

IF ((x>=9000).AND.(x<9500))THEN
qnoselow = (1.2433913043485579e-11)*(x-9000)**3 +
(3.0637491304347728e-06)*(x-9000)**2 &
+ (-8.9383043478258098e-05)*(x-9000) +
(3.0620000000000003e-01)
END IF

IF ((x>=9500).AND.(x<10000))THEN
qnoselow = (1.2433913043475415e-11)*(x-9500)**3 +
(3.08240000000000058e-06)*(x-9500)**2 &
+ (2.9836915217391290e-03)*(x-9500) + (1.0289999999999999
e+00)
END IF

IF ((x>=10000).AND.(x<11000))THEN
qnoselow = (-3.1946782608695561e-10)*(x-10000)**3 +
(3.1010508695652130e-06)*(x-10000)**2 &
+ (6.0754169565217413e-03)*(x-10000) +
(3.2930000000000001e+00)
END IF

IF ((x>=11000).AND.(x<12000))THEN

```

```

      qnoselow = (3.0823739130434582e-10)*(x-11000)**3 +
      (2.1426473913043547e-06)*(x-11000)**2 &
+ (1.1319115217391301e-02)*(x-11000) +
      (1.2150000000000000e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
      qnoselow = (-2.0471032608695631e-10)*(x-12000)**3 +
      (3.0673595652173887e-06)*(x-12000)**2 &
+ (1.6529122173913047e-02)*(x-12000) +
      (2.5920000000000002e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
      qnoselow = (-6.8807934782608647e-11)*(x-14000)**3 +
      (1.8390976086956524e-06)*(x-14000)**2 &
+ (2.6342036521739127e-02)*(x-14000) +
      (6.9609999999999999e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
      qnosehigh = (6.3822222222222218e-05)*(x) +
      (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
      qnosehigh = (-1.2174956521739795e-10)*(x-9000)**3 +
      (4.1414243478260953e-06)*(x-9000)**2 &
+ (1.1292521739130202e-04)*(x-9000) + (5.7440000000000002
      e-01)
END IF
IF ((x>=9500).AND.(x<10000))THEN

```



```

    qnosehigh = (-1.2174956521739456e-10)*(x-9500)**3 +
                (3.9588000000000001e-06)*(x-9500)**2 &
+ (4.1630373913043488e-03)*(x-9500) + (1.6510000000000000
    e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN
    qnosehigh = (-5.3370086956521461e-10)*(x-10000)**3 +
                (3.7761756521739117e-06)*(x-10000)**2 &
+ (8.0305252173913030e-03)*(x-10000) +
                (4.7069999999999999e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
    qnosehigh = (5.6315304347826145e-10)*(x-11000)**3 +
                (2.1750730434782584e-06)*(x-11000)**2 &
+ (1.3981773913043482e-02)*(x-11000) +
                (1.5980000000000000e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
    qnosehigh = (-3.6011086956521609e-10)*(x-12000)**3 +
                (3.8645321739130386e-06)*(x-12000)**2 &
+ (2.0021379130434783e-02)*(x-12000) +
                (3.27000000000000003e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
    qnosehigh = (-4.3977826086961645e-11)*(x-14000)**3 +
                (1.7038669565217527e-06)*(x-14000)**2 &
+ (3.1158177391304345e-02)*(x-14000) +
                (8.5319999999999993e+01)

```

```

END IF

  qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
  newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
rndiff = 1.0-(10.000000-nose)/(10.000000-5.000000)
IF ((nose>5.0000000).AND.(nose<=10.00)) THEN
  IF ((x>=0).AND.(x<9000))THEN
    qnoselow = (2.151111111111110e-04)*(x) +
      (0.0000000000000000e+00)
  END IF
  IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (2.1707936507934286e-10)*(x-9000)**3 +
      (8.8943809523809754e-06)*(x-9000)**2 &
      + (8.0253968253967756e-04)*(x-9000) + (1.935999999999999
        e+00)
  END IF
  IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (2.1707936507935640e-10)*(x-9500)**3 +
      (9.219999999999998e-06)*(x-9500)**2 &
      + (9.8597301587301598e-03)*(x-9500) + (4.5880000000000001
        e+00)
  END IF
  IF ((x>=10000).AND.(x<11000))THEN
    qnoselow = (-1.5881587301587148e-09)*(x-10000)**3 +
      (9.5456190476190344e-06)*(x-10000)**2 &
      + (1.9242539682539677e-02)*(x-10000) +
      (1.1850000000000000e+01)
  END IF

```

```

IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (1.1595555555555525e-09)*(x-11000)**3 +
    (4.7811428571428519e-06)*(x-11000)**2 &
  + (3.3569301587301599e-02)*(x-11000) +
    (3.9049999999999997e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (-7.0246825396825559e-10)*(x-12000)**3 +
    (8.2598095238095280e-06)*(x-12000)**2 &
  + (4.6610253968253960e-02)*(x-12000) +
    (7.8560000000000002e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnoselow = (-7.0246825396825042e-10)*(x-14000)**3 +
    (4.0450000000000002e-06)*(x-14000)**2 &
  + (7.1219873015873000e-02)*(x-14000) +
    (1.9919999999999999e+02)
END IF
if ((x>=0).AND.(x<9000))THEN
  qnosehigh = (3.8166666666666666e-04)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (-2.1270370370370099e-09)*(x-9000)**3 +
    (1.4720555555555513e-05)*(x-9000)**2 &
  + (8.5148148148149613e-04)*(x-9000) + (3.4350000000000001
    e+00)
END IF

```

```

IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (-2.1270370370370372e-09)*(x-9500)**3 +
    (1.15300000000000012e-05)*(x-9500)**2 &
+ (1.3976759259259251e-02)*(x-9500) + (7.2750000000000004
    e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (8.2907407407407901e-10)*(x-10000)**3 +
    (8.3394444444444340e-06)*(x-10000)**2 &
+ (2.3911481481481485e-02)*(x-10000) +
    (1.6879999999999999e+01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (-2.3642592592592600e-09)*(x-11000)**3 +
    (1.0826666666666671e-05)*(x-11000)**2 &
+ (4.3077592592592591e-02)*(x-11000) +
    (4.99600000000000001e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (8.6018518518521542e-11)*(x-12000)**3 +
    (3.7338888888888843e-06)*(x-12000)**2 &
+ (5.7638148148148154e-02)*(x-12000) +
    (1.0150000000000000e+02)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (8.6018518518516876e-11)*(x-14000)**3 +
    (4.2499999999999957e-06)*(x-14000)**2 &

```

```

+ (7.3605925925925950e-02)*(x-14000) +
    (2.3240000000000001e+02)
END IF

qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<9000))THEN
qnoselow = (6.382222222222218e-05)*(x) +
    (0.0000000000000000e+00)
END IF

IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (-1.2174956521739795e-10)*(x-9000)**3 +
        (4.1414243478260953e-06)*(x-9000)**2 &
+ (1.1292521739130202e-04)*(x-9000) + (5.7440000000000002
    e-01)
END IF

IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (-1.2174956521739456e-10)*(x-9500)**3 +
        (3.9588000000000001e-06)*(x-9500)**2 &
+ (4.1630373913043488e-03)*(x-9500) + (1.6510000000000000
    e+00)
END IF

IF ((x>=10000).AND.(x<11000))THEN
    qnoselow = (-5.3370086956521461e-10)*(x-10000)**3 +
        (3.7761756521739117e-06)*(x-10000)**2 &
+ (8.0305252173913030e-03)*(x-10000) +
        (4.7069999999999999e+00)
END IF

IF ((x>=11000).AND.(x<12000))THEN

```

```

      qnoselow = (5.6315304347826145e-10)*(x-11000)**3 +
      (2.1750730434782584e-06)*(x-11000)**2 &
+ (1.3981773913043482e-02)*(x-11000) +
      (1.5980000000000000e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
      qnoselow = (-3.6011086956521609e-10)*(x-12000)**3 +
      (3.8645321739130386e-06)*(x-12000)**2 &
+ (2.0021379130434783e-02)*(x-12000) +
      (3.27000000000000003e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
      qnoselow = (-4.3977826086961645e-11)*(x-14000)**3 +
      (1.7038669565217527e-06)*(x-14000)**2 &
+ (3.1158177391304345e-02)*(x-14000) +
      (8.5319999999999993e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
      qnosehigh = (1.13444444444444443e-04)*(x) +
      (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
      qnosehigh = (-9.5734202898551496e-10)*(x-9000)**3 +
      (6.3760130434782717e-06)*(x-9000)**2 &
+ (1.0732898550724279e-04)*(x-9000) + (1.0209999999999999
      e+00)
END IF
IF ((x>=9500).AND.(x<10000))THEN

```

```

qnosehigh = (-9.5734202898550979e-10)*(x-9500)**3 +
(4.9399999999999976e-06)*(x-9500)**2 &
+ (5.7653355072463785e-03)*(x-9500) + (2.5489999999999999
e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnosehigh = (4.1168405797101727e-10)*(x-10000)**3 +
(3.5039869565217379e-06)*(x-10000)**2 &
+ (9.9873289855072437e-03)*(x-10000) +
(6.5469999999999997e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnosehigh = (-8.8939420289854722e-10)*(x-11000)**3 +
(4.7390391304347788e-06)*(x-11000)**2 &
+ (1.8230355072463771e-02)*(x-11000) +
(2.0449999999999999e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
qnosehigh = (7.5759057971014226e-11)*(x-12000)**3 +
(2.0708565217391287e-06)*(x-12000)**2 &
+ (2.5040250724637687e-02)*(x-12000) +
(4.25300000000000001e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
qnosehigh = (-3.3340181159420385e-10)*(x-14000)**3 +
(2.5254108695652242e-06)*(x-14000)**2 &
+ (3.4232785507246372e-02)*(x-14000) +
(1.0150000000000000e+02)

```

```

END IF

  qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
  newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if

END IF !density if

!!!!!!!!!!!!!!!!!!!!NEW ALTITUDE RANGE!!!!!!!!!!!!!!!!!!!!

IF ((rho <= 0.0006313700) .AND. (rho >= 0.0003059200)) THEN
  rhodiff = (0.0006313700 - rho) / (0.0006313700 - 0.0003059200)
  rndiff = 1.0 - (0.100000 - nose) / (0.100000 - 0.049999)
  IF ((nose > 0.049999) .AND. (nose <= 0.10)) THEN
    IF ((x >= 0) .AND. (x < 9000)) THEN
      qnoselow = (1.6955555555555555e-06)*(x) +
        (0.0000000000000000e+00)
    END IF
    IF ((x >= 9000) .AND. (x < 9500)) THEN
      qnoselow = (2.0969739130434884e-10)*(x-9000)**3 +
        (1.9549391304347674e-07)*(x-9000)**2 &
        + (1.2448695652174435e-05)*(x-9000) + (1.5259999999999999
          e-02)
    END IF
    IF ((x >= 9500) .AND. (x < 10000)) THEN
      qnoselow = (2.0969739130434809e-10)*(x-9500)**3 +
        (5.10040000000000046e-07)*(x-9500)**2 &
        + (3.6521565217391281e-04)*(x-9500) + (9.6570000000000003
          e-02)
    END IF
  END IF
END IF

```



```

END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (6.4985217391303241e-11)*(x-10000)**3 +
    (8.2458608695652238e-07)*(x-10000)**2 &
+ (1.0325286956521743e-03)*(x-10000) +
    (4.32900000000000001e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (4.5801739130434293e-11)*(x-11000)**3 +
    (1.0195417391304361e-06)*(x-11000)**2 &
+ (2.8766565217391289e-03)*(x-11000) +
    (2.35500000000000000e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (5.6115217391305519e-11)*(x-12000)**3 +
    (1.1569469565217371e-06)*(x-12000)**2 &
+ (5.0531452173913039e-03)*(x-12000) +
    (6.2969999999999997e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnoselow = (-4.0398043478262290e-11)*(x-14000)**3 +
    (1.4936382608695658e-06)*(x-14000)**2 &
+ (1.0354315652173918e-02)*(x-14000) +
    (2.14800000000000000e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnoselow = (-4.0398043478258794e-11)*(x-16000)**3 +
    (1.2512499999999988e-06)*(x-16000)**2 &

```

```

+ (1.5844092173913034e-02)*(x-16000) +
(4.7840000000000003e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
qnosehigh = (3.0111111111111108e-06)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnosehigh = (2.8076028985507420e-10)*(x-9000)**3 +
(3.4665956521738907e-07)*(x-9000)**2 &
+ (6.4801449275369020e-06)*(x-9000) + (2.7099999999999999
e-02)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnosehigh = (2.8076028985507126e-10)*(x-9500)**3 +
(7.67800000000000136e-07)*(x-9500)**2 &
+ (5.6370992753623165e-04)*(x-9500) + (1.5210000000000001
e-01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnosehigh = (4.5979420289853994e-11)*(x-10000)**3 +
(1.1889404347826094e-06)*(x-10000)**2 &
+ (1.5420801449275365e-03)*(x-10000) +
(6.6100000000000003e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnosehigh = (6.3222028985507541e-11)*(x-11000)**3 +
(1.3268786956521739e-06)*(x-11000)**2 &

```

```

+ (4.0578992753623174e-03)*(x-11000) +
    (3.4380000000000002e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
    qnosehigh = (5.9396920289854532e-11)*(x-12000)**3 +
        (1.5165447826086972e-06)*(x-12000)**2 &
+ (6.9013227536231878e-03)*(x-12000) +
    (8.8859999999999992e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
    qnosehigh = (-1.9029384057970108e-11)*(x-14000)**3 +
        (1.8729263043478250e-06)*(x-14000)**2 &
+ (1.3680264927536231e-02)*(x-14000) +
    (2.9230000000000000e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
    qnosehigh = (-1.9029384057971484e-11)*(x-16000)**3 +
        (1.7587499999999982e-06)*(x-16000)**2 &
+ (2.0943617536231890e-02)*(x-16000) +
    (6.3930000000000000e+01)
END IF
    qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
    if ((x>=0).AND.(x<9000))THEN
        qnoselow = (4.5177777777777781e-07)*(x) +
            (0.0000000000000000e+00)
    END IF
    IF ((x>=9000).AND.(x<9500))THEN

```

```

      qnoselow = (5.7920394202898254e-11)*(x-9000)**3 +
                (9.0571408695652644e-08)*(x-9000)**2 &
+ (1.3421971014491164e-06)*(x-9000) + (4.0660000000000002
      e-03)
END IF
IF ((x>=9500).AND.(x<10000))THEN
      qnoselow = (5.7920394202898460e-11)*(x-9500)**3 +
                (1.7745199999999991e-07)*(x-9500)**2 &
+ (1.3535390144927544e-04)*(x-9500) + (3.4619999999999998
      e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
      qnoselow = (3.3321211594203017e-11)*(x-10000)**3 +
                (2.6433259130434769e-07)*(x-10000)**2 &
+ (3.5624619710144921e-04)*(x-10000) +
      (1.53900000000000001e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
      qnoselow = (1.7028759420290216e-11)*(x-11000)**3 +
                (3.6429622608695626e-07)*(x-11000)**2 &
+ (9.8487501449275364e-04)*(x-11000) +
      (8.0779999999999996e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
      qnoselow = (2.1920311594203027e-11)*(x-12000)**3 +
                (4.1538250434782533e-07)*(x-12000)**2 &
+ (1.7645537449275368e-03)*(x-12000) +
      (2.1739999999999999e+00)

```

```

END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnoselow = (-2.8234062318840565e-11)*(x-14000)**3 +
    (5.4690437391304380e-07)*(x-14000)**2 &
+ (3.6891275014492744e-03)*(x-14000) +
    (7.5400000000000000e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnoselow = (-2.8234062318840856e-11)*(x-16000)**3 +
    (3.77500000000000020e-07)*(x-16000)**2 &
+ (5.5379362492753628e-03)*(x-16000) +
    (1.6879999999999999e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
  qnosehigh = (8.2244444444444446e-07)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (1.0485099130434734e-10)*(x-9000)**3 +
    (1.1016751304347899e-07)*(x-9000)**2 &
+ (1.6679495652173693e-05)*(x-9000) + (7.401999999999997
    e-03)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (1.0485099130434755e-10)*(x-9500)**3 +
    (2.6744399999999997e-07)*(x-9500)**2 &
+ (2.0548525217391317e-04)*(x-9500) + (5.6390000000000003
    e-02)

```

```

END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (3.0612017391304490e-11)*(x-10000)**3 +
    (4.2472048695652151e-07)*(x-10000)**2 &
+ (5.5156749565217380e-04)*(x-10000) +
    (2.3910000000000001e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (2.5598939130434478e-11)*(x-11000)**3 +
    (5.1655653913043551e-07)*(x-11000)**2 &
+ (1.4928445217391303e-03)*(x-11000) +
    (1.2460000000000000e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (3.2509717391303809e-11)*(x-12000)**3 +
    (5.9335335652173969e-07)*(x-12000)**2 &
+ (2.6027544173913048e-03)*(x-12000) +
    (3.2810000000000001e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (-2.5776943478260463e-11)*(x-14000)**3 +
    (7.8841166086956655e-07)*(x-14000)**2 &
+ (5.3662844521739092e-03)*(x-14000) +
    (1.1119999999999999e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnosehigh = (-2.5776943478262421e-11)*(x-16000)**3 +
    (6.3374999999999880e-07)*(x-16000)**2 &

```

```

+ (8.2106077739130499e-03)*(x-16000) +
(2.4800000000000001e+01)
END IF

qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
newqqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
rndiff = 1.0-(0.230000-nose)/(0.230000-0.100000)
IF ((nose>0.1000000).AND.(nose<=0.23)) THEN
  IF ((x>=0).AND.(x<9000))THEN
    qnoselow = (3.0111111111111108e-06)*(x) +
(0.0000000000000000e+00)
  END IF
  IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (2.8076028985507420e-10)*(x-9000)**3 +
(3.4665956521738907e-07)*(x-9000)**2 &
+ (6.4801449275369020e-06)*(x-9000) + (2.7099999999999999
e-02)
  END IF
  IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (2.8076028985507126e-10)*(x-9500)**3 +
(7.67800000000000136e-07)*(x-9500)**2 &
+ (5.6370992753623165e-04)*(x-9500) + (1.5210000000000001
e-01)
  END IF
  IF ((x>=10000).AND.(x<11000))THEN
    qnoselow = (4.5979420289853994e-11)*(x-10000)**3 +
(1.1889404347826094e-06)*(x-10000)**2 &

```

```

+ (1.5420801449275365e-03)*(x-10000) +
(6.6100000000000003e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnoselow = (6.3222028985507541e-11)*(x-11000)**3 +
(1.3268786956521739e-06)*(x-11000)**2 &
+ (4.0578992753623174e-03)*(x-11000) +
(3.4380000000000002e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
qnoselow = (5.9396920289854532e-11)*(x-12000)**3 +
(1.5165447826086972e-06)*(x-12000)**2 &
+ (6.9013227536231878e-03)*(x-12000) +
(8.8859999999999992e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
qnoselow = (-1.9029384057970108e-11)*(x-14000)**3 +
(1.8729263043478250e-06)*(x-14000)**2 &
+ (1.3680264927536231e-02)*(x-14000) +
(2.9230000000000000e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
qnoselow = (-1.9029384057971484e-11)*(x-16000)**3 +
(1.7587499999999982e-06)*(x-16000)**2 &
+ (2.0943617536231890e-02)*(x-16000) +
(6.3930000000000000e+01)
END IF
if ((x>=0).AND.(x<9000))THEN

```



```

qnosehigh = (5.9066666666666663e-06)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (3.0399669565217168e-10)*(x-9000)**3 +
(7.4552495652174201e-07)*(x-9000)**2 &
+ (-3.8681652173913951e-05)*(x-9000) +
(5.3159999999999999e-02)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (3.0399669565217256e-10)*(x-9500)**3 +
(1.2015200000000002e-06)*(x-9500)**2 &
+ (9.3484082608695691e-04)*(x-9500) + (2.5819999999999999
e-01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (4.1266086956516973e-12)*(x-10000)**3 +
(1.6575150434782620e-06)*(x-10000)**2 &
+ (2.3643583478260865e-03)*(x-10000) +
(1.0640000000000001e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (4.8336869565219628e-11)*(x-11000)**3 +
(1.6698948695652149e-06)*(x-11000)**2 &
+ (5.6917682608695654e-03)*(x-11000) +
(5.0899999999999999e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN

```

```

      qnosehigh = (3.4655108695652446e-11)*(x-12000)**3 +
      (1.8149054782608679e-06)*(x-12000)**2 &
+ (9.1765686086956542e-03)*(x-12000) +
      (1.2500000000000000e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
      qnosehigh = (-1.8181021739132138e-11)*(x-14000)**3 +
      (2.0228361304347883e-06)*(x-14000)**2 &
+ (1.6852051826086956e-02)*(x-14000) +
      (3.8390000000000001e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
      qnosehigh = (-1.8181021739132138e-11)*(x-16000)**3 +
      (1.9137500000000022e-06)*(x-16000)**2 &
+ (2.4725224086956522e-02)*(x-16000) +
      (8.0040000000000006e+01)
END IF
      qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<9000))THEN
      qnoselow = (8.2244444444444446e-07)*(x) +
      (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
      qnoselow = (1.0485099130434734e-10)*(x-9000)**3 +
      (1.1016751304347899e-07)*(x-9000)**2 &
+ (1.6679495652173693e-05)*(x-9000) + (7.401999999999997
      e-03)
END IF

```

```

IF ((x>=9500).AND.(x<10000))THEN
  qnoselow = (1.0485099130434755e-10)*(x-9500)**3 +
    (2.6744399999999997e-07)*(x-9500)**2 &
+ (2.0548525217391317e-04)*(x-9500) + (5.6390000000000003
  e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (3.0612017391304490e-11)*(x-10000)**3 +
    (4.2472048695652151e-07)*(x-10000)**2 &
+ (5.5156749565217380e-04)*(x-10000) +
    (2.3910000000000001e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (2.5598939130434478e-11)*(x-11000)**3 +
    (5.1655653913043551e-07)*(x-11000)**2 &
+ (1.4928445217391303e-03)*(x-11000) +
    (1.2460000000000000e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (3.2509717391303809e-11)*(x-12000)**3 +
    (5.9335335652173969e-07)*(x-12000)**2 &
+ (2.6027544173913048e-03)*(x-12000) +
    (3.2810000000000001e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnoselow = (-2.5776943478260463e-11)*(x-14000)**3 +
    (7.8841166086956655e-07)*(x-14000)**2 &

```

```

+ (5.3662844521739092e-03)*(x-14000) +
  (1.1119999999999999e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnoselow = (-2.5776943478262421e-11)*(x-16000)**3 +
    (6.3374999999999880e-07)*(x-16000)**2 &
+ (8.2106077739130499e-03)*(x-16000) +
  (2.48000000000000001e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
  qnosehigh = (1.6866666666666667e-06)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (1.3770701449275411e-10)*(x-9000)**3 +
    (2.3571947826086895e-07)*(x-9000)**2 &
+ (1.3393507246377001e-05)*(x-9000) + (1.51800000000000001
  e-02)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (1.3770701449275359e-10)*(x-9500)**3 +
    (4.42280000000000028e-07)*(x-9500)**2 &
+ (3.5239324637681153e-04)*(x-9500) + (9.8019999999999996
  e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (2.5205971014492993e-11)*(x-10000)**3 +
    (6.4884052173913002e-07)*(x-10000)**2 &

```

```

+ (8.9795350724637701e-04)*(x-10000) +
  (4.0200000000000002e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (2.6289101449273926e-11)*(x-11000)**3 +
    (7.2445843478261048e-07)*(x-11000)**2 &
+ (2.2712524637681160e-03)*(x-11000) +
  (1.9740000000000000e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (3.2827971014492479e-11)*(x-12000)**3 +
    (8.0332573913043575e-07)*(x-12000)**2 &
+ (3.7990366376811587e-03)*(x-12000) +
  (4.9960000000000004e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (-1.2965594202898902e-11)*(x-14000)**3 +
    (1.0002935652173921e-06)*(x-14000)**2 &
+ (7.4062752463768117e-03)*(x-14000) +
  (1.6070000000000000e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnosehigh = (-1.2965594202898479e-11)*(x-16000)**3 +
    (9.2249999999999890e-07)*(x-16000)**2 &
+ (1.1251862376811594e-02)*(x-16000) +
  (3.4780000000000001e+01)
END IF
qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)

```

```

    newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
rndiff = 1.0-(0.500000-nose)/(0.500000-0.230000)
IF ((nose>0.230000).AND.(nose<=0.50)) THEN
    IF ((x>=0).AND.(x<9000))THEN
        qnoselow = (5.9066666666666663e-06)*(x) +
            (0.0000000000000000e+00)
    END IF
    IF ((x>=9000).AND.(x<9500))THEN
        qnoselow = (3.0399669565217168e-10)*(x-9000)**3 +
            (7.4552495652174201e-07)*(x-9000)**2 &
            + (-3.8681652173913951e-05)*(x-9000) +
            (5.3159999999999999e-02)
    END IF
    IF ((x>=9500).AND.(x<10000))THEN
        qnoselow = (3.0399669565217256e-10)*(x-9500)**3 +
            (1.2015200000000002e-06)*(x-9500)**2 &
            + (9.3484082608695691e-04)*(x-9500) + (2.5819999999999999
                e-01)
    END IF
    IF ((x>=10000).AND.(x<11000))THEN
        qnoselow = (4.1266086956516973e-12)*(x-10000)**3 +
            (1.6575150434782620e-06)*(x-10000)**2 &
            + (2.3643583478260865e-03)*(x-10000) +
            (1.0640000000000001e+00)
    END IF
    IF ((x>=11000).AND.(x<12000))THEN

```

```

      qnoselow = (4.8336869565219628e-11)*(x-11000)**3 +
      (1.6698948695652149e-06)*(x-11000)**2 &
+ (5.6917682608695654e-03)*(x-11000) +
      (5.0899999999999999e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
      qnoselow = (3.4655108695652446e-11)*(x-12000)**3 +
      (1.8149054782608679e-06)*(x-12000)**2 &
+ (9.1765686086956542e-03)*(x-12000) +
      (1.2500000000000000e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
      qnoselow = (-1.8181021739132138e-11)*(x-14000)**3 +
      (2.0228361304347883e-06)*(x-14000)**2 &
+ (1.6852051826086956e-02)*(x-14000) +
      (3.8390000000000000e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
      qnoselow = (-1.8181021739132138e-11)*(x-16000)**3 +
      (1.9137500000000002e-06)*(x-16000)**2 &
+ (2.4725224086956522e-02)*(x-16000) +
      (8.0040000000000000e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
      qnosehigh = (1.0795555555555555e-05)*(x) +
      (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN

```

```

qnosehigh = (2.4984886956521521e-10)*(x-9000)**3 +
(1.3567466956521770e-06)*(x-9000)**2 &
+ (-1.0575556521739226e-04)*(x-9000) +
(9.7159999999999996e-02)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnosehigh = (2.4984886956522110e-10)*(x-9500)**3 +
(1.7315199999999970e-06)*(x-9500)**2 &
+ (1.4383777826086962e-03)*(x-9500) + (4.1470000000000001
e-01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnosehigh = (-5.5577739130435639e-11)*(x-10000)**3 +
(2.1062933043478272e-06)*(x-10000)**2 &
+ (3.3572844347826091e-03)*(x-10000) +
(1.5980000000000001e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnosehigh = (1.3020869565217379e-12)*(x-11000)**3 +
(1.9395600869565237e-06)*(x-11000)**2 &
+ (7.4031378260869564e-03)*(x-11000) +
(7.0060000000000002e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
qnosehigh = (-7.7423913043301298e-13)*(x-12000)**3 +
(1.9434663478260804e-06)*(x-12000)**2 &
+ (1.1286164260869569e-02)*(x-12000) +
(1.6350000000000001e+01)

```



```

END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (-4.7095152173914534e-11)*(x-14000)**3 +
    (1.9388209130434836e-06)*(x-14000)**2 &
+ (1.9050738782608694e-02)*(x-14000) +
    (4.6689999999999998e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnosehigh = (-4.7095152173915484e-11)*(x-16000)**3 +
    (1.65625000000000051e-06)*(x-16000)**2 &
+ (2.6240880608695654e-02)*(x-16000) +
    (9.21700000000000002e+01)
END IF
  qnosel = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<9000))THEN
  qnoselow = (1.6866666666666667e-06)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnoselow = (1.3770701449275411e-10)*(x-9000)**3 +
    (2.3571947826086895e-07)*(x-9000)**2 &
+ (1.3393507246377001e-05)*(x-9000) + (1.51800000000000001
    e-02)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnoselow = (1.3770701449275359e-10)*(x-9500)**3 +
    (4.422800000000000028e-07)*(x-9500)**2 &

```

```

+ (3.5239324637681153e-04)*(x-9500) + (9.801999999999996
e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (2.5205971014492993e-11)*(x-10000)**3 +
    (6.4884052173913002e-07)*(x-10000)**2 &
+ (8.9795350724637701e-04)*(x-10000) +
    (4.0200000000000002e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (2.6289101449273926e-11)*(x-11000)**3 +
    (7.2445843478261048e-07)*(x-11000)**2 &
+ (2.2712524637681160e-03)*(x-11000) +
    (1.9740000000000000e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (3.2827971014492479e-11)*(x-12000)**3 +
    (8.0332573913043575e-07)*(x-12000)**2 &
+ (3.7990366376811587e-03)*(x-12000) +
    (4.9960000000000004e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnoselow = (-1.2965594202898902e-11)*(x-14000)**3 +
    (1.0002935652173921e-06)*(x-14000)**2 &
+ (7.4062752463768117e-03)*(x-14000) +
    (1.6070000000000000e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN

```

```

qnoselow = (-1.2965594202898479e-11)*(x-16000)**3 +
(9.2249999999999890e-07)*(x-16000)**2 &
+ (1.1251862376811594e-02)*(x-16000) +
(3.47800000000000001e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
qnosehigh = (3.2511111111111111e-06)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnosehigh = (1.5389553623188397e-10)*(x-9000)**3 +
(4.3587669565217406e-07)*(x-9000)**2 &
+ (8.6677681159419420e-06)*(x-9000) + (2.92600000000000001
e-02)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnosehigh = (1.5389553623188462e-10)*(x-9500)**3 +
(6.6671999999999982e-07)*(x-9500)**2 &
+ (5.5996611594202905e-04)*(x-9500) + (1.6180000000000000
e-01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnosehigh = (-7.3710724637680253e-12)*(x-10000)**3 +
(8.9756330434782568e-07)*(x-10000)**2 &
+ (1.3421077681159423e-03)*(x-10000) +
(6.27700000000000004e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN

```

```

      qnosehigh = (2.1428753623187794e-11)*(x-11000)**3 +
                  (8.7545008695652330e-07)*(x-11000)**2 &
+ (3.1151211594202897e-03)*(x-11000) +
                  (2.8599999999999999e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
      qnosehigh = (1.9804927536231981e-11)*(x-12000)**3 +
                  (9.3973634782608593e-07)*(x-12000)**2 &
+ (4.9303075942028997e-03)*(x-12000) +
                  (6.8719999999999999e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
      qnosehigh = (-1.9760985507245403e-11)*(x-14000)**3 +
                  (1.0585659130434768e-06)*(x-14000)**2 &
+ (8.9269121159420272e-03)*(x-14000) +
                  (2.0649999999999999e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
      qnosehigh = (-1.9760985507246779e-11)*(x-16000)**3 +
                  (9.4000000000000000e-07)*(x-16000)**2 &
+ (1.2924043942028989e-02)*(x-16000) +
                  (4.2579999999999998e+01)
END IF
      qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
      newqnr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
      rndiff = 1.0-(1.000000-nose)/(1.000000-0.500000)
IF ((nose>0.5000000).AND.(nose<=1.00)) THEN

```

```

IF ((x>=0).AND.(x<9000))THEN
qnoselow = (1.0795555555555555e-05)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnoselow = (2.4984886956521521e-10)*(x-9000)**3 +
(1.3567466956521770e-06)*(x-9000)**2 &
+ (-1.0575556521739226e-04)*(x-9000) +
(9.7159999999999996e-02)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnoselow = (2.4984886956522110e-10)*(x-9500)**3 +
(1.7315199999999970e-06)*(x-9500)**2 &
+ (1.4383777826086962e-03)*(x-9500) + (4.1470000000000001
e-01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnoselow = (-5.5577739130435639e-11)*(x-10000)**3 +
(2.1062933043478272e-06)*(x-10000)**2 &
+ (3.3572844347826091e-03)*(x-10000) +
(1.5980000000000001e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnoselow = (1.3020869565217379e-12)*(x-11000)**3 +
(1.9395600869565237e-06)*(x-11000)**2 &
+ (7.4031378260869564e-03)*(x-11000) +
(7.0060000000000002e+00)
END IF

```

```

IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (-7.7423913043301298e-13)*(x-12000)**3 +
    (1.9434663478260804e-06)*(x-12000)**2 &
  + (1.1286164260869569e-02)*(x-12000) +
    (1.63500000000000001e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnoselow = (-4.7095152173914534e-11)*(x-14000)**3 +
    (1.9388209130434836e-06)*(x-14000)**2 &
  + (1.9050738782608694e-02)*(x-14000) +
    (4.6689999999999998e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnoselow = (-4.7095152173915484e-11)*(x-16000)**3 +
    (1.65625000000000051e-06)*(x-16000)**2 &
  + (2.6240880608695654e-02)*(x-16000) +
    (9.21700000000000002e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
  qnosehigh = (1.8133333333333335e-05)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (1.0391971014492448e-10)*(x-9000)**3 +
    (2.1413204347826131e-06)*(x-9000)**2 &
  + (-1.6944014492753778e-04)*(x-9000) +
    (1.63200000000000001e-01)
END IF

```

```

IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (1.0391971014493549e-10)*(x-9500)**3 +
    (2.2971999999999939e-06)*(x-9500)**2 &
+ (2.0498200724637689e-03)*(x-9500) + (6.2680000000000002
  e-01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (-5.8039420289856761e-11)*(x-10000)**3 +
    (2.4530795652173920e-06)*(x-10000)**2 &
+ (4.4249598550724642e-03)*(x-10000) +
    (2.2389999999999999e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (-2.5496202898550761e-10)*(x-11000)**3 +
    (2.2789613043478294e-06)*(x-11000)**2 &
+ (9.1570007246376782e-03)*(x-11000) +
    (9.0589999999999993e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (2.8420307971014545e-10)*(x-12000)**3 +
    (1.5140752173913028e-06)*(x-12000)**2 &
+ (1.2950037246376814e-02)*(x-12000) +
    (2.0239999999999998e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (-4.8509061594202872e-10)*(x-14000)**3 +
    (3.2192936956521708e-06)*(x-14000)**2 &

```

```

+ (2.2416775072463771e-02)*(x-14000) +
(5.446999999999999e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnosehigh = (-4.8509061594202707e-10)*(x-16000)**3 +
(3.0874999999999831e-07)*(x-16000)**2 &
+ (2.9472862463768110e-02)*(x-16000) +
(1.0830000000000000e+02)
END IF
  qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
  if ((x>=0).AND.(x<9000))THEN
    qnoselow = (3.251111111111111e-06)*(x) +
(0.0000000000000000e+00)
  END IF
  IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (1.5389553623188397e-10)*(x-9000)**3 +
(4.3587669565217406e-07)*(x-9000)**2 &
+ (8.6677681159419420e-06)*(x-9000) + (2.9260000000000001
e-02)
  END IF
  IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (1.5389553623188462e-10)*(x-9500)**3 +
(6.6671999999999982e-07)*(x-9500)**2 &
+ (5.5996611594202905e-04)*(x-9500) + (1.6180000000000000
e-01)
  END IF
  IF ((x>=10000).AND.(x<11000))THEN

```



```

qnoselow = (-7.3710724637680253e-12)*(x-10000)**3 +
(8.9756330434782568e-07)*(x-10000)**2 &
+ (1.3421077681159423e-03)*(x-10000) +
(6.2770000000000004e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnoselow = (2.1428753623187794e-11)*(x-11000)**3 +
(8.7545008695652330e-07)*(x-11000)**2 &
+ (3.1151211594202897e-03)*(x-11000) +
(2.8599999999999999e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
qnoselow = (1.9804927536231981e-11)*(x-12000)**3 +
(9.3973634782608593e-07)*(x-12000)**2 &
+ (4.9303075942028997e-03)*(x-12000) +
(6.8719999999999999e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
qnoselow = (-1.9760985507245403e-11)*(x-14000)**3 +
(1.0585659130434768e-06)*(x-14000)**2 &
+ (8.9269121159420272e-03)*(x-14000) +
(2.0649999999999999e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
qnoselow = (-1.9760985507246779e-11)*(x-16000)**3 +
(9.4000000000000000e-07)*(x-16000)**2 &
+ (1.2924043942028989e-02)*(x-16000) +
(4.2579999999999998e+01)

```

```

END IF
if ((x>=0).AND.(x<9000))THEN
qnosehigh = (5.728888888888889e-06)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnosehigh = (1.1814249275362591e-10)*(x-9000)**3 +
(7.5330626086956127e-07)*(x-9000)**2 &
+ (-1.5308753623187065e-05)*(x-9000) +
(5.1560000000000002e-02)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnosehigh = (1.1814249275362359e-10)*(x-9500)**3 +
(9.30520000000000079e-07)*(x-9500)**2 &
+ (8.2660437681159364e-04)*(x-9500) + (2.4700000000000000
e-01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnosehigh = (-3.9164985507245381e-11)*(x-10000)**3 +
(1.1077337391304330e-06)*(x-10000)**2 &
+ (1.8457312463768122e-03)*(x-10000) +
(9.0769999999999995e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnosehigh = (9.0574492753606305e-12)*(x-11000)**3 +
(9.9023878260869750e-07)*(x-11000)**2 &
+ (3.9437037681159422e-03)*(x-11000) +
(3.8220000000000001e+00)

```

```

END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (-2.1689855072467357e-12)*(x-12000)**3 +
    (1.0174111304347832e-06)*(x-12000)**2 &
+ (5.9513536811594192e-03)*(x-12000) +
    (8.7650000000000006e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (-2.3441202898550733e-11)*(x-14000)**3 +
    (1.0043972173913060e-06)*(x-14000)**2 &
+ (9.9949703768115912e-03)*(x-14000) +
    (2.4719999999999999e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnosehigh = (-2.3441202898553379e-11)*(x-16000)**3 +
    (8.6375000000000264e-07)*(x-16000)**2 &
+ (1.3731264811594207e-02)*(x-16000) +
    (4.8539999999999999e+01)
END IF
  qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
  newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
  rndiff = 1.0-(2.300000-nose)/(2.300000-1.000000)
IF ((nose>1.0000000).AND.(nose<=2.30)) THEN
  IF ((x>=0).AND.(x<9000))THEN
    qnoselow = (1.8133333333333335e-05)*(x) +
      (0.0000000000000000e+00)
  END IF

```

```

IF ((x>=9000).AND.(x<9500))THEN
  qnoselow = (1.0391971014492448e-10)*(x-9000)**3 +
    (2.1413204347826131e-06)*(x-9000)**2 &
  + (-1.6944014492753778e-04)*(x-9000) +
    (1.6320000000000001e-01)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnoselow = (1.0391971014493549e-10)*(x-9500)**3 +
    (2.2971999999999939e-06)*(x-9500)**2 &
  + (2.0498200724637689e-03)*(x-9500) + (6.2680000000000002
    e-01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (-5.8039420289856761e-11)*(x-10000)**3 +
    (2.4530795652173920e-06)*(x-10000)**2 &
  + (4.4249598550724642e-03)*(x-10000) +
    (2.2389999999999999e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (-2.5496202898550761e-10)*(x-11000)**3 +
    (2.2789613043478294e-06)*(x-11000)**2 &
  + (9.1570007246376782e-03)*(x-11000) +
    (9.0589999999999993e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (2.8420307971014545e-10)*(x-12000)**3 +
    (1.5140752173913028e-06)*(x-12000)**2 &

```

```

+ (1.2950037246376814e-02)*(x-12000) +
(2.0239999999999998e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnoselow = (-4.8509061594202872e-10)*(x-14000)**3 +
(3.2192936956521708e-06)*(x-14000)**2 &
+ (2.2416775072463771e-02)*(x-14000) +
(5.4469999999999999e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnoselow = (-4.8509061594202707e-10)*(x-16000)**3 +
(3.0874999999999831e-07)*(x-16000)**2 &
+ (2.9472862463768110e-02)*(x-16000) +
(1.0830000000000000e+02)
END IF
if ((x>=0).AND.(x<9000))THEN
  qnosehigh = (3.4022222222222226e-05)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (1.2433913043485579e-11)*(x-9000)**3 +
(3.0637491304347728e-06)*(x-9000)**2 &
+ (-8.9383043478258098e-05)*(x-9000) +
(3.0620000000000003e-01)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (1.2433913043475415e-11)*(x-9500)**3 +
(3.0824000000000058e-06)*(x-9500)**2 &

```

```

+ (2.9836915217391290e-03)*(x-9500) + (1.0289999999999999
e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (-3.1946782608695561e-10)*(x-10000)**3 +
    (3.1010508695652130e-06)*(x-10000)**2 &
+ (6.0754169565217413e-03)*(x-10000) +
    (3.29300000000000001e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (3.0823739130434582e-10)*(x-11000)**3 +
    (2.1426473913043547e-06)*(x-11000)**2 &
+ (1.1319115217391301e-02)*(x-11000) +
    (1.21500000000000000e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (-2.0471032608695631e-10)*(x-12000)**3 +
    (3.0673595652173887e-06)*(x-12000)**2 &
+ (1.6529122173913047e-02)*(x-12000) +
    (2.59200000000000002e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (-6.8807934782608647e-11)*(x-14000)**3 +
    (1.8390976086956524e-06)*(x-14000)**2 &
+ (2.6342036521739127e-02)*(x-14000) +
    (6.9609999999999999e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN

```

```

qnosehigh = (-6.8807934782611283e-11)*(x-16000)**3 +
(1.42625000000000073e-06)*(x-16000)**2 &
+ (3.2872731739130433e-02)*(x-16000) +
(1.2909999999999999e+02)
END IF

qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<9000))THEN
qnoselow = (5.7288888888888889e-06)*(x) +
(0.0000000000000000e+00)
END IF

IF ((x>=9000).AND.(x<9500))THEN
qnoselow = (1.1814249275362591e-10)*(x-9000)**3 +
(7.5330626086956127e-07)*(x-9000)**2 &
+ (-1.5308753623187065e-05)*(x-9000) +
(5.1560000000000002e-02)
END IF

IF ((x>=9500).AND.(x<10000))THEN
qnoselow = (1.1814249275362359e-10)*(x-9500)**3 +
(9.30520000000000079e-07)*(x-9500)**2 &
+ (8.2660437681159364e-04)*(x-9500) + (2.4700000000000000
e-01)
END IF

IF ((x>=10000).AND.(x<11000))THEN
qnoselow = (-3.9164985507245381e-11)*(x-10000)**3 +
(1.1077337391304330e-06)*(x-10000)**2 &
+ (1.8457312463768122e-03)*(x-10000) +
(9.0769999999999995e-01)
END IF

```

```

IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (9.0574492753606305e-12)*(x-11000)**3 +
    (9.9023878260869750e-07)*(x-11000)**2 &
  + (3.9437037681159422e-03)*(x-11000) +
    (3.8220000000000001e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (-2.1689855072467357e-12)*(x-12000)**3 +
    (1.0174111304347832e-06)*(x-12000)**2 &
  + (5.9513536811594192e-03)*(x-12000) +
    (8.7650000000000006e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnoselow = (-2.3441202898550733e-11)*(x-14000)**3 +
    (1.0043972173913060e-06)*(x-14000)**2 &
  + (9.9949703768115912e-03)*(x-14000) +
    (2.4719999999999999e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnoselow = (-2.3441202898553379e-11)*(x-16000)**3 +
    (8.6375000000000264e-07)*(x-16000)**2 &
  + (1.3731264811594207e-02)*(x-16000) +
    (4.8539999999999999e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
  qnosehigh = (1.0982222222222222e-05)*(x) +
    (0.0000000000000000e+00)
END IF

```



```

IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (2.1558666666667148e-11)*(x-9000)**3 +
    (1.2769419999999990e-06)*(x-9000)**2 &
+ (-3.73406666666666246e-05)*(x-9000) +
    (9.883999999999997e-02)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (2.15586666666665032e-11)*(x-9500)**3 +
    (1.30928000000000017e-06)*(x-9500)**2 &
+ (1.2557703333333332e-03)*(x-9500) + (4.0210000000000001
    e-01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (-8.68373333333331995e-11)*(x-10000)**3 +
    (1.3416179999999980e-06)*(x-10000)**2 &
+ (2.5812193333333336e-03)*(x-10000) +
    (1.36000000000000001e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (1.8950666666666521e-11)*(x-11000)**3 +
    (1.08110600000000009e-06)*(x-11000)**2 &
+ (5.0039433333333334e-03)*(x-11000) +
    (5.195999999999997e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (-7.59808333333334867e-11)*(x-12000)**3 +
    (1.13795800000000022e-06)*(x-12000)**2 &

```

```

+ (7.2230073333333349e-03)*(x-12000) +
  (1.13000000000000001e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (1.19446166666666876e-10)*(x-14000)**3 +
    (6.8207299999999925e-07)*(x-14000)**2 &
+ (1.0863069333333326e-02)*(x-14000) +
  (2.96900000000000001e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnosehigh = (1.19446166666666284e-10)*(x-16000)**3 +
    (1.39875000000000004e-06)*(x-16000)**2 &
+ (1.5024715333333348e-02)*(x-16000) +
  (5.51000000000000001e+01)
END IF
  qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
  newqr = qnose2*rhodiff+qnosel*(1.0-rhodiff)
END IF !nose radius if
rndiff = 1.0-(5.000000-nose)/(5.000000-2.300000)
IF ((nose>2.300000).AND.(nose<=5.00)) THEN
  IF ((x>=0).AND.(x<9000))THEN
    qnoselow = (3.4022222222222226e-05)*(x) +
      (0.0000000000000000e+00)
  END IF
  IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (1.2433913043485579e-11)*(x-9000)**3 +
      (3.0637491304347728e-06)*(x-9000)**2 &

```

```

+ (−8.9383043478258098e−05)*(x−9000) +
(3.0620000000000003e−01)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnoselow = (1.2433913043475415e−11)*(x−9500)**3 +
(3.08240000000000058e−06)*(x−9500)**2 &
+ (2.9836915217391290e−03)*(x−9500) + (1.0289999999999999
e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnoselow = (−3.1946782608695561e−10)*(x−10000)**3 +
(3.1010508695652130e−06)*(x−10000)**2 &
+ (6.0754169565217413e−03)*(x−10000) +
(3.29300000000000001e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnoselow = (3.0823739130434582e−10)*(x−11000)**3 +
(2.1426473913043547e−06)*(x−11000)**2 &
+ (1.1319115217391301e−02)*(x−11000) +
(1.21500000000000000e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
qnoselow = (−2.0471032608695631e−10)*(x−12000)**3 +
(3.0673595652173887e−06)*(x−12000)**2 &
+ (1.6529122173913047e−02)*(x−12000) +
(2.59200000000000002e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN

```

```

      qnoselow =  (-6.8807934782608647e-11)*(x-14000)**3 +
      (1.8390976086956524e-06)*(x-14000)**2 &
+ (2.6342036521739127e-02)*(x-14000) +
      (6.960999999999999e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
      qnoselow =  (-6.8807934782611283e-11)*(x-16000)**3 +
      (1.42625000000000073e-06)*(x-16000)**2 &
+ (3.2872731739130433e-02)*(x-16000) +
      (1.2909999999999999e+02)
END IF
if ((x>=0).AND.(x<9000))THEN
      qnosehigh = (6.3822222222222218e-05)*(x) +
      (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
      qnosehigh =  (-1.2174956521739795e-10)*(x-9000)**3 +
      (4.1414243478260953e-06)*(x-9000)**2 &
+ (1.1292521739130202e-04)*(x-9000) + (5.7440000000000002
      e-01)
END IF
IF ((x>=9500).AND.(x<10000))THEN
      qnosehigh =  (-1.2174956521739456e-10)*(x-9500)**3 +
      (3.9588000000000001e-06)*(x-9500)**2 &
+ (4.1630373913043488e-03)*(x-9500) + (1.6510000000000000
      e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN

```

```

qnosehigh = (-5.3370086956521461e-10)*(x-10000)**3 +
(3.7761756521739117e-06)*(x-10000)**2 &
+ (8.0305252173913030e-03)*(x-10000) +
(4.7069999999999999e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnosehigh = (5.6315304347826145e-10)*(x-11000)**3 +
(2.1750730434782584e-06)*(x-11000)**2 &
+ (1.3981773913043482e-02)*(x-11000) +
(1.5980000000000000e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
qnosehigh = (-3.6011086956521609e-10)*(x-12000)**3 +
(3.8645321739130386e-06)*(x-12000)**2 &
+ (2.0021379130434783e-02)*(x-12000) +
(3.2700000000000000e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
qnosehigh = (-4.3977826086961645e-11)*(x-14000)**3 +
(1.7038669565217527e-06)*(x-14000)**2 &
+ (3.1158177391304345e-02)*(x-14000) +
(8.5319999999999993e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
qnosehigh = (-4.3977826086954659e-11)*(x-16000)**3 +
(1.4400000000000000e-06)*(x-16000)**2 &
+ (3.7445911304347816e-02)*(x-16000) +
(1.5409999999999999e+02)

```

```

END IF

  qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
  if ((x>=0).AND.(x<9000))THEN
    qnoselow = (1.0982222222222222e-05)*(x) +
      (0.0000000000000000e+00)
  END IF

  IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (2.155866666666667148e-11)*(x-9000)**3 +
      (1.27694199999999990e-06)*(x-9000)**2 &
    + (-3.734066666666666246e-05)*(x-9000) +
      (9.8839999999999997e-02)
  END IF

  IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (2.155866666666665032e-11)*(x-9500)**3 +
      (1.30928000000000017e-06)*(x-9500)**2 &
    + (1.2557703333333332e-03)*(x-9500) + (4.0210000000000001
      e-01)
  END IF

  IF ((x>=10000).AND.(x<11000))THEN
    qnoselow = (-8.68373333333331995e-11)*(x-10000)**3 +
      (1.34161799999999980e-06)*(x-10000)**2 &
    + (2.5812193333333336e-03)*(x-10000) +
      (1.36000000000000001e+00)
  END IF

  IF ((x>=11000).AND.(x<12000))THEN
    qnoselow = (1.89506666666666521e-11)*(x-11000)**3 +
      (1.08110600000000009e-06)*(x-11000)**2 &

```

```

+ (5.0039433333333334e-03)*(x-11000) +
(5.1959999999999997e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
qnoselow = (-7.59808333333334867e-11)*(x-12000)**3 +
(1.13795800000000022e-06)*(x-12000)**2 &
+ (7.2230073333333349e-03)*(x-12000) +
(1.13000000000000001e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
qnoselow = (1.19446166666666876e-10)*(x-14000)**3 +
(6.8207299999999925e-07)*(x-14000)**2 &
+ (1.0863069333333326e-02)*(x-14000) +
(2.96900000000000001e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
qnoselow = (1.19446166666666284e-10)*(x-16000)**3 +
(1.39875000000000004e-06)*(x-16000)**2 &
+ (1.5024715333333348e-02)*(x-16000) +
(5.51000000000000001e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
qnosehigh = (2.0244444444444444e-05)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnosehigh = (-8.2005797101450814e-11)*(x-9000)**3 +
(1.8378086956521752e-06)*(x-9000)**2 &

```

```

+ (1.9971014492750070e-06)*(x-9000) + (1.8220000000000000
e-01)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (-8.2005797101447001e-11)*(x-9500)**3 +
    (1.7147999999999990e-06)*(x-9500)**2 &
+ (1.7783014492753621e-03)*(x-9500) + (6.3239999999999996
e-01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (-1.7938840579710258e-10)*(x-10000)**3 +
    (1.5917913043478262e-06)*(x-10000)**2 &
+ (3.4315971014492758e-03)*(x-10000) +
    (1.9399999999999999e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (1.9535942028985493e-10)*(x-11000)**3 +
    (1.0536260869565248e-06)*(x-11000)**2 &
+ (6.0770144927536204e-03)*(x-11000) +
    (6.7839999999999998e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (-1.2618840579710134e-10)*(x-12000)**3 +
    (1.6397043478260860e-06)*(x-12000)**2 &
+ (8.7703449275362347e-03)*(x-12000) +
    (1.4109999999999999e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN

```



```

      qnosehigh = (-3.2512318840580415e-11)*(x-14000)**3 +
      (8.8257391304347698e-07)*(x-14000)**2 &
+ (1.3814901449275363e-02)*(x-14000) +
      (3.7200000000000003e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
      qnosehigh = (-3.2512318840576499e-11)*(x-16000)**3 +
      (6.8750000000000231e-07)*(x-16000)**2 &
+ (1.6955049275362306e-02)*(x-16000) +
      (6.8099999999999994e+01)
END IF
      qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
      newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
      rndiff = 1.0-(10.000000-nose)/(10.000000-5.000000)
IF ((nose>5.0000000).AND.(nose<=10.00)) THEN
      IF ((x>=0).AND.(x<9000))THEN
      qnoselow = (6.3822222222222218e-05)*(x) +
      (0.0000000000000000e+00)
END IF
      IF ((x>=9000).AND.(x<9500))THEN
      qnoselow = (-1.2174956521739795e-10)*(x-9000)**3 +
      (4.1414243478260953e-06)*(x-9000)**2 &
+ (1.1292521739130202e-04)*(x-9000) + (5.7440000000000002
      e-01)
END IF
      IF ((x>=9500).AND.(x<10000))THEN

```

```

qnoselow = (-1.2174956521739456e-10)*(x-9500)**3 +
(3.95880000000000001e-06)*(x-9500)**2 &
+ (4.1630373913043488e-03)*(x-9500) + (1.6510000000000000
e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnoselow = (-5.3370086956521461e-10)*(x-10000)**3 +
(3.7761756521739117e-06)*(x-10000)**2 &
+ (8.0305252173913030e-03)*(x-10000) +
(4.7069999999999999e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnoselow = (5.6315304347826145e-10)*(x-11000)**3 +
(2.1750730434782584e-06)*(x-11000)**2 &
+ (1.3981773913043482e-02)*(x-11000) +
(1.59800000000000000e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
qnoselow = (-3.6011086956521609e-10)*(x-12000)**3 +
(3.8645321739130386e-06)*(x-12000)**2 &
+ (2.0021379130434783e-02)*(x-12000) +
(3.270000000000000003e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
qnoselow = (-4.3977826086961645e-11)*(x-14000)**3 +
(1.7038669565217527e-06)*(x-14000)**2 &
+ (3.1158177391304345e-02)*(x-14000) +
(8.5319999999999993e+01)

```

```

END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnoselow = (-4.3977826086954659e-11)*(x-16000)**3 +
    (1.44000000000000038e-06)*(x-16000)**2 &
+ (3.7445911304347816e-02)*(x-16000) +
    (1.5409999999999999e+02)
END IF
if ((x>=0).AND.(x<9000))THEN
  qnosehigh = (1.1344444444444443e-04)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (-9.5734202898551496e-10)*(x-9000)**3 +
    (6.3760130434782717e-06)*(x-9000)**2 &
+ (1.0732898550724279e-04)*(x-9000) + (1.0209999999999999
    e+00)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (-9.5734202898550979e-10)*(x-9500)**3 +
    (4.9399999999999976e-06)*(x-9500)**2 &
+ (5.7653355072463785e-03)*(x-9500) + (2.5489999999999999
    e+00)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (4.1168405797101727e-10)*(x-10000)**3 +
    (3.5039869565217379e-06)*(x-10000)**2 &
+ (9.9873289855072437e-03)*(x-10000) +
    (6.5469999999999997e+00)

```

```

END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (-8.8939420289854722e-10)*(x-11000)**3 +
    (4.7390391304347788e-06)*(x-11000)**2 &
+ (1.8230355072463771e-02)*(x-11000) +
    (2.0449999999999999e+01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (7.5759057971014226e-11)*(x-12000)**3 +
    (2.0708565217391287e-06)*(x-12000)**2 &
+ (2.5040250724637687e-02)*(x-12000) +
    (4.25300000000000001e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (-3.3340181159420385e-10)*(x-14000)**3 +
    (2.5254108695652242e-06)*(x-14000)**2 &
+ (3.4232785507246372e-02)*(x-14000) +
    (1.01500000000000000e+02)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnosehigh = (-3.3340181159420902e-10)*(x-16000)**3 +
    (5.25000000000000461e-07)*(x-16000)**2 &
+ (4.0333607246376822e-02)*(x-16000) +
    (1.77400000000000001e+02)
END IF
  qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<9000))THEN

```

```

qnoselow = (2.0244444444444444e-05)*(x) +
            (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (-8.2005797101450814e-11)*(x-9000)**3 +
                (1.8378086956521752e-06)*(x-9000)**2 &
                + (1.9971014492750070e-06)*(x-9000) + (1.8220000000000000
                e-01)
END IF
IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (-8.2005797101447001e-11)*(x-9500)**3 +
                (1.7147999999999990e-06)*(x-9500)**2 &
                + (1.7783014492753621e-03)*(x-9500) + (6.3239999999999996
                e-01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
    qnoselow = (-1.7938840579710258e-10)*(x-10000)**3 +
                (1.5917913043478262e-06)*(x-10000)**2 &
                + (3.4315971014492758e-03)*(x-10000) +
                (1.9399999999999999e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
    qnoselow = (1.9535942028985493e-10)*(x-11000)**3 +
                (1.0536260869565248e-06)*(x-11000)**2 &
                + (6.0770144927536204e-03)*(x-11000) +
                (6.7839999999999998e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN

```

```

      qnoselow =  (-1.2618840579710134e-10)*(x-12000)**3 +
      (1.6397043478260860e-06)*(x-12000)**2 &
+ (8.7703449275362347e-03)*(x-12000) +
      (1.4109999999999999e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
      qnoselow =  (-3.2512318840580415e-11)*(x-14000)**3 +
      (8.8257391304347698e-07)*(x-14000)**2 &
+ (1.3814901449275363e-02)*(x-14000) +
      (3.72000000000000003e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
      qnoselow =  (-3.2512318840576499e-11)*(x-16000)**3 +
      (6.87500000000000231e-07)*(x-16000)**2 &
+ (1.6955049275362306e-02)*(x-16000) +
      (6.8099999999999994e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
      qnosehigh = (3.58111111111111109e-05)*(x) +
      (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
      qnosehigh =  (-1.9278579710145095e-10)*(x-9000)**3 +
      (2.4185786956521755e-06)*(x-9000)**2 &
+ (9.5907101449275170e-05)*(x-9000) + (3.2229999999999998
      e-01)
END IF
IF ((x>=9500).AND.(x<10000))THEN

```

```

qnosehigh = (-1.9278579710144841e-10)*(x-9500)**3 +
(2.1293999999999995e-06)*(x-9500)**2 &
+ (2.3698964492753624e-03)*(x-9500) + (9.507999999999998
e-01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnosehigh = (-2.9592840579709823e-10)*(x-10000)**3 +
(1.8402213043478218e-06)*(x-10000)**2 &
+ (4.3547071014492754e-03)*(x-10000) +
(2.6440000000000001e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnosehigh = (3.6719942028985369e-10)*(x-11000)**3 +
(9.5243608695652446e-07)*(x-11000)**2 &
+ (7.1473644927536242e-03)*(x-11000) +
(8.5429999999999993e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
qnosehigh = (-2.1297590579710074e-10)*(x-12000)**3 +
(2.0540343478260836e-06)*(x-12000)**2 &
+ (1.0153834927536234e-02)*(x-12000) +
(1.7010000000000002e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
qnosehigh = (-3.2904818840578967e-11)*(x-14000)**3 +
(7.7617891304347834e-07)*(x-14000)**2 &
+ (1.5814261449275360e-02)*(x-14000) +
(4.3829999999999998e+01)

```

```

END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnosehigh = (-3.2904818840580693e-11)*(x-16000)**3 +
    (5.78750000000000124e-07)*(x-16000)**2 &
+ (1.8524119275362325e-02)*(x-16000) +
    (7.8299999999999997e+01)
END IF
  qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
  newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if

END IF !density if
!!!!!!!!!!!!!!!!!!!!NEW ALTITUDE RANGE!!!!!!!!!!!!!!!!!!!!

IF ((rho<=0.0003059200).AND.(rho>=0.0001471300)) THEN
  rhodiff = (0.0003059200-rho)/(0.0003059200-0.0001471300)
  rndiff = 1.0-(0.100000-nose)/(0.100000-0.049999)
  IF ((nose>0.049999).AND.(nose<=0.10)) THEN
    IF ((x>=0).AND.(x<9000))THEN
      qnoselow = (4.5177777777777781e-07)*(x) +
        (0.0000000000000000e+00)
    END IF
    IF ((x>=9000).AND.(x<9500))THEN
      qnoselow = (5.7920394202898254e-11)*(x-9000)**3 +
        (9.0571408695652644e-08)*(x-9000)**2 &
+ (1.3421971014491164e-06)*(x-9000) + (4.06600000000000002
        e-03)
    END IF
  END IF
END IF

```



```

END IF
IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (5.7920394202898460e-11)*(x-9500)**3 +
        (1.7745199999999991e-07)*(x-9500)**2 &
+ (1.3535390144927544e-04)*(x-9500) + (3.4619999999999998
    e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
    qnoselow = (3.3321211594203017e-11)*(x-10000)**3 +
        (2.6433259130434769e-07)*(x-10000)**2 &
+ (3.5624619710144921e-04)*(x-10000) +
        (1.53900000000000001e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
    qnoselow = (1.7028759420290216e-11)*(x-11000)**3 +
        (3.6429622608695626e-07)*(x-11000)**2 &
+ (9.8487501449275364e-04)*(x-11000) +
        (8.0779999999999996e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
    qnoselow = (2.1920311594203027e-11)*(x-12000)**3 +
        (4.1538250434782533e-07)*(x-12000)**2 &
+ (1.7645537449275368e-03)*(x-12000) +
        (2.1739999999999999e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
    qnoselow = (-2.8234062318840565e-11)*(x-14000)**3 +
        (5.4690437391304380e-07)*(x-14000)**2 &

```

```

+ (3.6891275014492744e-03)*(x-14000) +
  (7.5400000000000000e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnoselow = (-2.8234062318840856e-11)*(x-16000)**3 +
    (3.77500000000000020e-07)*(x-16000)**2 &
+ (5.5379362492753628e-03)*(x-16000) +
  (1.6879999999999999e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
  qnosehigh = (8.2244444444444446e-07)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (1.0485099130434734e-10)*(x-9000)**3 +
    (1.1016751304347899e-07)*(x-9000)**2 &
+ (1.6679495652173693e-05)*(x-9000) + (7.401999999999997
  e-03)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (1.0485099130434755e-10)*(x-9500)**3 +
    (2.6744399999999997e-07)*(x-9500)**2 &
+ (2.0548525217391317e-04)*(x-9500) + (5.6390000000000003
  e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (3.0612017391304490e-11)*(x-10000)**3 +
    (4.2472048695652151e-07)*(x-10000)**2 &

```

```

+ (5.5156749565217380e-04)*(x-10000) +
(2.3910000000000001e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (2.5598939130434478e-11)*(x-11000)**3 +
(5.1655653913043551e-07)*(x-11000)**2 &
+ (1.4928445217391303e-03)*(x-11000) +
(1.2460000000000000e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (3.2509717391303809e-11)*(x-12000)**3 +
(5.9335335652173969e-07)*(x-12000)**2 &
+ (2.6027544173913048e-03)*(x-12000) +
(3.2810000000000001e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (-2.5776943478260463e-11)*(x-14000)**3 +
(7.8841166086956655e-07)*(x-14000)**2 &
+ (5.3662844521739092e-03)*(x-14000) +
(1.1119999999999999e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnosehigh = (-2.5776943478262421e-11)*(x-16000)**3 +
(6.3374999999999880e-07)*(x-16000)**2 &
+ (8.2106077739130499e-03)*(x-16000) +
(2.4800000000000001e+01)
END IF
qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)

```

```

if ((x>=0).AND.(x<9000))THEN
qnoselow = (1.1844444444444446e-07)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnoselow = (1.3391959420289862e-11)*(x-9000)**3 +
(3.8584060869565191e-08)*(x-9000)**2 &
+ (-9.3202028985506223e-07)*(x-9000) +
(1.06600000000000001e-03)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnoselow = (1.3391959420289889e-11)*(x-9500)**3 +
(5.8671999999999998e-08)*(x-9500)**2 &
+ (4.7696010144927526e-05)*(x-9500) + (1.1920000000000000
e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnoselow = (1.1818081159420293e-11)*(x-10000)**3 +
(7.8759939130434765e-08)*(x-10000)**2 &
+ (1.1641197971014494e-04)*(x-10000) +
(5.2109999999999997e-02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnoselow = (6.4997159420288411e-12)*(x-11000)**3 +
(1.1421418260869587e-07)*(x-11000)**2 &
+ (3.0938610144927534e-04)*(x-11000) +
(2.5910000000000000e-01)
END IF

```

```

IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (8.2899311594203144e-12)*(x-12000)**3 +
    (1.3371333043478252e-07)*(x-12000)**2 &
  + (5.5731361449275359e-04)*(x-12000) +
    (6.8920000000000003e-01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnoselow = (-1.5137986231883886e-11)*(x-14000)**3 +
    (1.8345291739130403e-07)*(x-14000)**2 &
  + (1.1916461101449274e-03)*(x-14000) +
    (2.4049999999999998e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnoselow = (-1.5137986231884381e-11)*(x-16000)**3 +
    (9.26250000000000368e-08)*(x-16000)**2 &
  + (1.7438019449275369e-03)*(x-16000) +
    (5.4009999999999998e+00)
END IF
if ((x>=0).AND.(x<9000))THEN
  qnosehigh = (2.1711111111111111e-07)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (2.5348968115942017e-11)*(x-9000)**3 +
    (5.2444547826086976e-08)*(x-9000)**2 &
  + (3.1324840579710078e-06)*(x-9000) + (1.9540000000000000
    e-03)
END IF

```

```

IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (2.5348968115941962e-11)*(x-9500)**3 +
    (9.04680000000000026e-08)*(x-9500)**2 &
+ (7.4588757971014497e-05)*(x-9500) + (1.98000000000000002
  e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (1.6160063768116075e-11)*(x-10000)**3 +
    (1.2849145217391294e-07)*(x-10000)**2 &
+ (1.8406848405797099e-04)*(x-10000) +
    (8.2879999999999995e-02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (1.0896776811594468e-11)*(x-11000)**3 +
    (1.7697164347826043e-07)*(x-11000)**2 &
+ (4.8953157971014509e-04)*(x-11000) +
    (4.11600000000000002e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (1.1627713768115993e-11)*(x-12000)**3 +
    (2.0966197391304339e-07)*(x-12000)**2 &
+ (8.7616519710144935e-04)*(x-12000) +
    (1.08900000000000000e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (-1.5800542753623243e-11)*(x-14000)**3 +
    (2.7942825652173935e-07)*(x-14000)**2 &

```

```

+ (1.8543456579710148e-03)*(x-14000) +
(3.7730000000000001e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnosehigh = (-1.5800542753622926e-11)*(x-16000)**3 +
(1.8462499999999905e-07)*(x-16000)**2 &
+ (2.7824521710144933e-03)*(x-16000) +
(8.4730000000000008e+00)
END IF
  qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
  newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
  rndiff = 1.0-(0.230000-nose)/(0.230000-0.100000)
IF ((nose>0.1000000).AND.(nose<=0.23)) THEN
  IF ((x>=0).AND.(x<9000))THEN
    qnoselow = (8.2244444444444446e-07)*(x) +
(0.0000000000000000e+00)
  END IF
  IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (1.0485099130434734e-10)*(x-9000)**3 +
(1.1016751304347899e-07)*(x-9000)**2 &
+ (1.6679495652173693e-05)*(x-9000) + (7.401999999999997
e-03)
  END IF
  IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (1.0485099130434755e-10)*(x-9500)**3 +
(2.6744399999999997e-07)*(x-9500)**2 &

```

```

+ (2.0548525217391317e-04)*(x-9500) + (5.6390000000000003
e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (3.0612017391304490e-11)*(x-10000)**3 +
    (4.2472048695652151e-07)*(x-10000)**2 &
+ (5.5156749565217380e-04)*(x-10000) +
    (2.39100000000000001e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (2.5598939130434478e-11)*(x-11000)**3 +
    (5.1655653913043551e-07)*(x-11000)**2 &
+ (1.4928445217391303e-03)*(x-11000) +
    (1.24600000000000000e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (3.2509717391303809e-11)*(x-12000)**3 +
    (5.9335335652173969e-07)*(x-12000)**2 &
+ (2.6027544173913048e-03)*(x-12000) +
    (3.28100000000000001e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnoselow = (-2.5776943478260463e-11)*(x-14000)**3 +
    (7.8841166086956655e-07)*(x-14000)**2 &
+ (5.3662844521739092e-03)*(x-14000) +
    (1.1119999999999999e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN

```



```

qnoselow = (-2.5776943478262421e-11)*(x-16000)**3 +
(6.3374999999999880e-07)*(x-16000)**2 &
+ (8.2106077739130499e-03)*(x-16000) +
(2.48000000000000001e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
qnosehigh = (1.6866666666666667e-06)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnosehigh = (1.3770701449275411e-10)*(x-9000)**3 +
(2.3571947826086895e-07)*(x-9000)**2 &
+ (1.3393507246377001e-05)*(x-9000) + (1.51800000000000001
e-02)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnosehigh = (1.3770701449275359e-10)*(x-9500)**3 +
(4.42280000000000028e-07)*(x-9500)**2 &
+ (3.5239324637681153e-04)*(x-9500) + (9.801999999999996
e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnosehigh = (2.5205971014492993e-11)*(x-10000)**3 +
(6.4884052173913002e-07)*(x-10000)**2 &
+ (8.9795350724637701e-04)*(x-10000) +
(4.02000000000000002e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN

```

```

    qnosehigh = (2.6289101449273926e-11)*(x-11000)**3 +
        (7.2445843478261048e-07)*(x-11000)**2 &
+ (2.2712524637681160e-03)*(x-11000) +
        (1.9740000000000000e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
    qnosehigh = (3.2827971014492479e-11)*(x-12000)**3 +
        (8.0332573913043575e-07)*(x-12000)**2 &
+ (3.7990366376811587e-03)*(x-12000) +
        (4.9960000000000004e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
    qnosehigh = (-1.2965594202898902e-11)*(x-14000)**3 +
        (1.0002935652173921e-06)*(x-14000)**2 &
+ (7.4062752463768117e-03)*(x-14000) +
        (1.6070000000000000e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
    qnosehigh = (-1.2965594202898479e-11)*(x-16000)**3 +
        (9.2249999999999890e-07)*(x-16000)**2 &
+ (1.1251862376811594e-02)*(x-16000) +
        (3.4780000000000001e+01)
END IF
    qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<9000))THEN
qnoselow = (2.1711111111111111e-07)*(x) +
        (0.0000000000000000e+00)
END IF

```

```

IF ((x>=9000).AND.(x<9500))THEN
  qnoselow = (2.5348968115942017e-11)*(x-9000)**3 +
    (5.2444547826086976e-08)*(x-9000)**2 &
+ (3.1324840579710078e-06)*(x-9000) + (1.9540000000000000
    e-03)
END IF

IF ((x>=9500).AND.(x<10000))THEN
  qnoselow = (2.5348968115941962e-11)*(x-9500)**3 +
    (9.04680000000000026e-08)*(x-9500)**2 &
+ (7.4588757971014497e-05)*(x-9500) + (1.98000000000000002
    e-02)
END IF

IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (1.6160063768116075e-11)*(x-10000)**3 +
    (1.2849145217391294e-07)*(x-10000)**2 &
+ (1.8406848405797099e-04)*(x-10000) +
    (8.2879999999999995e-02)
END IF

IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (1.0896776811594468e-11)*(x-11000)**3 +
    (1.7697164347826043e-07)*(x-11000)**2 &
+ (4.8953157971014509e-04)*(x-11000) +
    (4.11600000000000002e-01)
END IF

IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (1.1627713768115993e-11)*(x-12000)**3 +
    (2.0966197391304339e-07)*(x-12000)**2 &

```

```

+ (8.7616519710144935e-04)*(x-12000) +
    (1.0890000000000000e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
    qnoselow = (-1.5800542753623243e-11)*(x-14000)**3 +
        (2.7942825652173935e-07)*(x-14000)**2 &
+ (1.8543456579710148e-03)*(x-14000) +
    (3.7730000000000001e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
    qnoselow = (-1.5800542753622926e-11)*(x-16000)**3 +
        (1.8462499999999905e-07)*(x-16000)**2 &
+ (2.7824521710144933e-03)*(x-16000) +
    (8.4730000000000008e+00)
END IF
if ((x>=0).AND.(x<9000))THEN
    qnosehigh = (4.5366666666666664e-07)*(x) +
        (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
    qnosehigh = (4.1923443478260443e-11)*(x-9000)**3 +
        (9.1000834782609251e-08)*(x-9000)**2 &
+ (7.1927217391302702e-06)*(x-9000) + (4.0829999999999998
    e-03)
END IF
IF ((x>=9500).AND.(x<10000))THEN
    qnosehigh = (4.1923443478260785e-11)*(x-9500)**3 +
        (1.5388599999999987e-07)*(x-9500)**2 &

```

```

+ (1.2963613913043485e-04)*(x-9500) + (3.5670000000000000
e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (2.0864113043478208e-11)*(x-10000)**3 +
    (2.1677116521739147e-07)*(x-10000)**2 &
+ (3.1496472173913031e-04)*(x-10000) +
    (1.4419999999999999e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (1.2737104347825845e-11)*(x-11000)**3 +
    (2.7936350434782639e-07)*(x-11000)**2 &
+ (8.1109939130434789e-04)*(x-11000) +
    (6.9679999999999997e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (1.6703163043478249e-11)*(x-12000)**3 +
    (3.1757481739130435e-07)*(x-12000)**2 &
+ (1.4080377130434782e-03)*(x-12000) +
    (1.8000000000000000e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (-1.2340632608695035e-11)*(x-14000)**3 +
    (4.1779379565217269e-07)*(x-14000)**2 &
+ (2.8787749391304346e-03)*(x-14000) +
    (6.0199999999999996e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN

```

```

      qnosehigh = (-1.2340632608696332e-11)*(x-16000)**3 +
      (3.43750000000000031e-07)*(x-16000)**2 &
+ (4.4018625304347849e-03)*(x-16000) +
      (1.3350000000000000e+01)
END IF

      qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
      newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if

      rndiff = 1.0-(0.500000-nose)/(0.500000-0.230000)
IF ((nose>0.2300000).AND.(nose<=0.50)) THEN
  IF ((x>=0).AND.(x<9000))THEN
    qnoselow = (1.6866666666666667e-06)*(x) +
      (0.0000000000000000e+00)
  END IF
  IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (1.3770701449275411e-10)*(x-9000)**3 +
      (2.3571947826086895e-07)*(x-9000)**2 &
+ (1.3393507246377001e-05)*(x-9000) + (1.51800000000000001
      e-02)
  END IF
  IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (1.3770701449275359e-10)*(x-9500)**3 +
      (4.42280000000000028e-07)*(x-9500)**2 &
+ (3.5239324637681153e-04)*(x-9500) + (9.8019999999999996
      e-02)
  END IF
  IF ((x>=10000).AND.(x<11000))THEN

```

```

      qnoselow = (2.5205971014492993e-11)*(x-10000)**3 +
      (6.4884052173913002e-07)*(x-10000)**2 &
+ (8.9795350724637701e-04)*(x-10000) +
      (4.0200000000000002e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
      qnoselow = (2.6289101449273926e-11)*(x-11000)**3 +
      (7.2445843478261048e-07)*(x-11000)**2 &
+ (2.2712524637681160e-03)*(x-11000) +
      (1.9740000000000000e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
      qnoselow = (3.2827971014492479e-11)*(x-12000)**3 +
      (8.0332573913043575e-07)*(x-12000)**2 &
+ (3.7990366376811587e-03)*(x-12000) +
      (4.9960000000000004e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
      qnoselow = (-1.2965594202898902e-11)*(x-14000)**3 +
      (1.0002935652173921e-06)*(x-14000)**2 &
+ (7.4062752463768117e-03)*(x-14000) +
      (1.6070000000000000e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
      qnoselow = (-1.2965594202898479e-11)*(x-16000)**3 +
      (9.2249999999999890e-07)*(x-16000)**2 &
+ (1.1251862376811594e-02)*(x-16000) +
      (3.4780000000000001e+01)

```

```

END IF
if ((x>=0).AND.(x<9000))THEN
qnosehigh = (3.2511111111111111e-06)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnosehigh = (1.5389553623188397e-10)*(x-9000)**3 +
(4.3587669565217406e-07)*(x-9000)**2 &
+ (8.6677681159419420e-06)*(x-9000) + (2.9260000000000001
e-02)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnosehigh = (1.5389553623188462e-10)*(x-9500)**3 +
(6.66719999999999982e-07)*(x-9500)**2 &
+ (5.5996611594202905e-04)*(x-9500) + (1.6180000000000000
e-01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnosehigh = (-7.3710724637680253e-12)*(x-10000)**3 +
(8.9756330434782568e-07)*(x-10000)**2 &
+ (1.3421077681159423e-03)*(x-10000) +
(6.27700000000000004e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnosehigh = (2.1428753623187794e-11)*(x-11000)**3 +
(8.7545008695652330e-07)*(x-11000)**2 &
+ (3.1151211594202897e-03)*(x-11000) +
(2.8599999999999999e+00)

```



```

END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (1.9804927536231981e-11)*(x-12000)**3 +
    (9.3973634782608593e-07)*(x-12000)**2 &
+ (4.9303075942028997e-03)*(x-12000) +
    (6.871999999999999e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (-1.9760985507245403e-11)*(x-14000)**3 +
    (1.0585659130434768e-06)*(x-14000)**2 &
+ (8.9269121159420272e-03)*(x-14000) +
    (2.064999999999999e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnosehigh = (-1.9760985507246779e-11)*(x-16000)**3 +
    (9.400000000000000e-07)*(x-16000)**2 &
+ (1.2924043942028989e-02)*(x-16000) +
    (4.257999999999999e+01)
END IF
  qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
  if ((x>=0).AND.(x<9000))THEN
    qnoselow = (4.536666666666666e-07)*(x) +
      (0.000000000000000e+00)
  END IF
  IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (4.1923443478260443e-11)*(x-9000)**3 +
      (9.1000834782609251e-08)*(x-9000)**2 &

```

```

+ (7.1927217391302702e-06)*(x-9000) + (4.082999999999998
e-03)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnoselow = (4.1923443478260785e-11)*(x-9500)**3 +
    (1.5388599999999987e-07)*(x-9500)**2 &
+ (1.2963613913043485e-04)*(x-9500) + (3.5670000000000000
e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (2.0864113043478208e-11)*(x-10000)**3 +
    (2.1677116521739147e-07)*(x-10000)**2 &
+ (3.1496472173913031e-04)*(x-10000) +
    (1.4419999999999999e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (1.2737104347825845e-11)*(x-11000)**3 +
    (2.7936350434782639e-07)*(x-11000)**2 &
+ (8.1109939130434789e-04)*(x-11000) +
    (6.967999999999997e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (1.6703163043478249e-11)*(x-12000)**3 +
    (3.1757481739130435e-07)*(x-12000)**2 &
+ (1.4080377130434782e-03)*(x-12000) +
    (1.8000000000000000e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN

```

```

      qnoselow = (-1.2340632608695035e-11)*(x-14000)**3 +
      (4.1779379565217269e-07)*(x-14000)**2 &
+ (2.8787749391304346e-03)*(x-14000) +
      (6.0199999999999996e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
      qnoselow = (-1.2340632608696332e-11)*(x-16000)**3 +
      (3.43750000000000031e-07)*(x-16000)**2 &
+ (4.4018625304347849e-03)*(x-16000) +
      (1.3350000000000000e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
      qnosehigh = (9.0077777777777771e-07)*(x) +
      (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
      qnosehigh = (5.2910121739131146e-11)*(x-9000)**3 +
      (1.6344881739130341e-07)*(x-9000)**2 &
+ (1.0234060869565519e-05)*(x-9000) + (8.1069999999999996
      e-03)
END IF
IF ((x>=9500).AND.(x<10000))THEN
      qnosehigh = (5.2910121739130086e-11)*(x-9500)**3 +
      (2.42814000000000042e-07)*(x-9500)**2 &
+ (2.1336546956521728e-04)*(x-9500) + (6.0699999999999997
      e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN

```

```

    qnosehigh = (2.1258756521738738e-11)*(x-10000)**3 +
        (3.2217918260869607e-07)*(x-10000)**2 &
+ (4.9586206086956526e-04)*(x-10000) +
        (2.3469999999999999e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
    qnosehigh = (8.0478521739127639e-12)*(x-11000)**3 +
        (3.8595545217391367e-07)*(x-11000)**2 &
+ (1.2039966956521736e-03)*(x-11000) +
        (1.0740000000000001e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
    qnosehigh = (1.7687706521738982e-11)*(x-12000)**3 +
        (4.1009900869565238e-07)*(x-12000)**2 &
+ (2.0000511565217392e-03)*(x-12000) +
        (2.6720000000000002e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
    qnosehigh = (-1.0537541304348163e-11)*(x-14000)**3 +
        (5.1622524782608818e-07)*(x-14000)**2 &
+ (3.8526996695652164e-03)*(x-14000) +
        (8.4540000000000006e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
    qnosehigh = (-1.0537541304347184e-11)*(x-16000)**3 +
        (4.5299999999999856e-07)*(x-16000)**2 &
+ (5.7911501652173911e-03)*(x-16000) +
        (1.8140000000000001e+01)

```

```

END IF

  qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
  newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
rndiff = 1.0-(1.000000-nose)/(1.000000-0.500000)
IF ((nose>0.5000000).AND.(nose<=1.00)) THEN
  IF ((x>=0).AND.(x<9000))THEN
    qnoselow = (3.251111111111111e-06)*(x) +
      (0.0000000000000000e+00)
  END IF
  IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (1.5389553623188397e-10)*(x-9000)**3 +
      (4.3587669565217406e-07)*(x-9000)**2 &
      + (8.6677681159419420e-06)*(x-9000) + (2.9260000000000001
        e-02)
  END IF
  IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (1.5389553623188462e-10)*(x-9500)**3 +
      (6.6671999999999982e-07)*(x-9500)**2 &
      + (5.5996611594202905e-04)*(x-9500) + (1.6180000000000000
        e-01)
  END IF
  IF ((x>=10000).AND.(x<11000))THEN
    qnoselow = (-7.3710724637680253e-12)*(x-10000)**3 +
      (8.9756330434782568e-07)*(x-10000)**2 &
      + (1.3421077681159423e-03)*(x-10000) +
      (6.2770000000000004e-01)
  END IF

```

```

IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (2.1428753623187794e-11)*(x-11000)**3 +
    (8.7545008695652330e-07)*(x-11000)**2 &
  + (3.1151211594202897e-03)*(x-11000) +
    (2.8599999999999999e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (1.9804927536231981e-11)*(x-12000)**3 +
    (9.3973634782608593e-07)*(x-12000)**2 &
  + (4.9303075942028997e-03)*(x-12000) +
    (6.8719999999999999e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnoselow = (-1.9760985507245403e-11)*(x-14000)**3 +
    (1.0585659130434768e-06)*(x-14000)**2 &
  + (8.9269121159420272e-03)*(x-14000) +
    (2.0649999999999999e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnoselow = (-1.9760985507246779e-11)*(x-16000)**3 +
    (9.4000000000000000e-07)*(x-16000)**2 &
  + (1.2924043942028989e-02)*(x-16000) +
    (4.2579999999999998e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
  qnosehigh = (5.7288888888888889e-06)*(x) +
    (0.0000000000000000e+00)
END IF

```

```

IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (1.1814249275362591e-10)*(x-9000)**3 +
    (7.5330626086956127e-07)*(x-9000)**2 &
  + (-1.5308753623187065e-05)*(x-9000) +
    (5.1560000000000002e-02)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (1.1814249275362359e-10)*(x-9500)**3 +
    (9.30520000000000079e-07)*(x-9500)**2 &
  + (8.2660437681159364e-04)*(x-9500) + (2.4700000000000000
    e-01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (-3.9164985507245381e-11)*(x-10000)**3 +
    (1.1077337391304330e-06)*(x-10000)**2 &
  + (1.8457312463768122e-03)*(x-10000) +
    (9.0769999999999995e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (9.0574492753606305e-12)*(x-11000)**3 +
    (9.9023878260869750e-07)*(x-11000)**2 &
  + (3.9437037681159422e-03)*(x-11000) +
    (3.8220000000000001e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (-2.1689855072467357e-12)*(x-12000)**3 +
    (1.0174111304347832e-06)*(x-12000)**2 &

```

```

+ (5.9513536811594192e-03)*(x-12000) +
      (8.7650000000000006e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (-2.3441202898550733e-11)*(x-14000)**3 +
      (1.0043972173913060e-06)*(x-14000)**2 &
+ (9.9949703768115912e-03)*(x-14000) +
      (2.4719999999999999e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnosehigh = (-2.3441202898553379e-11)*(x-16000)**3 +
      (8.6375000000000264e-07)*(x-16000)**2 &
+ (1.3731264811594207e-02)*(x-16000) +
      (4.8539999999999999e+01)
END IF
  qnosel = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<9000))THEN
  qnoselow = (9.0077777777777771e-07)*(x) +
      (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnoselow = (5.2910121739131146e-11)*(x-9000)**3 +
      (1.6344881739130341e-07)*(x-9000)**2 &
+ (1.0234060869565519e-05)*(x-9000) + (8.1069999999999996
      e-03)
END IF
IF ((x>=9500).AND.(x<10000))THEN

```



```

qnoselow = (5.2910121739130086e-11)*(x-9500)**3 +
(2.42814000000000042e-07)*(x-9500)**2 &
+ (2.1336546956521728e-04)*(x-9500) + (6.0699999999999997
e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnoselow = (2.1258756521738738e-11)*(x-10000)**3 +
(3.2217918260869607e-07)*(x-10000)**2 &
+ (4.9586206086956526e-04)*(x-10000) +
(2.3469999999999999e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnoselow = (8.0478521739127639e-12)*(x-11000)**3 +
(3.8595545217391367e-07)*(x-11000)**2 &
+ (1.2039966956521736e-03)*(x-11000) +
(1.07400000000000001e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
qnoselow = (1.7687706521738982e-11)*(x-12000)**3 +
(4.1009900869565238e-07)*(x-12000)**2 &
+ (2.0000511565217392e-03)*(x-12000) +
(2.67200000000000002e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
qnoselow = (-1.0537541304348163e-11)*(x-14000)**3 +
(5.1622524782608818e-07)*(x-14000)**2 &
+ (3.8526996695652164e-03)*(x-14000) +
(8.45400000000000006e+00)

```

```

END IF
IF ((x>=16000).AND.(x<18000))THEN
    qnoselow = (-1.0537541304347184e-11)*(x-16000)**3 +
        (4.5299999999999856e-07)*(x-16000)**2 &
+ (5.7911501652173911e-03)*(x-16000) +
        (1.8140000000000001e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
    qnosehigh = (1.6488888888888890e-06)*(x) +
        (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
    qnosehigh = (6.8925391304346964e-11)*(x-9000)**3 +
        (2.3965191304347959e-07)*(x-9000)**2 &
+ (2.6282695652173449e-05)*(x-9000) + (1.4840000000000001
        e-02)
END IF
IF ((x>=9500).AND.(x<10000))THEN
    qnosehigh = (6.8925391304348127e-11)*(x-9500)**3 +
        (3.4303999999999956e-07)*(x-9500)**2 &
+ (3.1762865217391326e-04)*(x-9500) + (9.650999999999999
        e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
    qnosehigh = (2.5092173913042931e-12)*(x-10000)**3 +
        (4.4642808695652170e-07)*(x-10000)**2 &
+ (7.1236269565217391e-04)*(x-10000) +
        (3.4970000000000001e-01)

```

```

END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (7.2977391304346795e-12)*(x-11000)**3 +
    (4.5395573913043505e-07)*(x-11000)**2 &
+ (1.6127465217391302e-03)*(x-11000) +
    (1.5109999999999999e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (9.5627173913041786e-12)*(x-12000)**3 +
    (4.7584895652173957e-07)*(x-12000)**2 &
+ (2.5425512173913043e-03)*(x-12000) +
    (3.5850000000000000e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (-1.1787543478260328e-11)*(x-14000)**3 +
    (5.3322526086956389e-07)*(x-14000)**2 &
+ (4.5606996521739126e-03)*(x-14000) +
    (1.0650000000000000e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnosehigh = (-1.1787543478261837e-11)*(x-16000)**3 +
    (4.6250000000000245e-07)*(x-16000)**2 &
+ (6.5521501739130442e-03)*(x-16000) +
    (2.1809999999999999e+01)
END IF
  qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
  newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if

```

```

rndiff = 1.0-(2.300000-nose)/(2.300000-1.000000)
IF ((nose>1.0000000).AND.(nose<=2.30)) THEN
  IF ((x>=0).AND.(x<9000))THEN
    qnoselow = (5.728888888888889e-06)*(x) +
      (0.0000000000000000e+00)
  END IF
  IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (1.1814249275362591e-10)*(x-9000)**3 +
      (7.5330626086956127e-07)*(x-9000)**2 &
    + (-1.5308753623187065e-05)*(x-9000) +
      (5.1560000000000002e-02)
  END IF
  IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (1.1814249275362359e-10)*(x-9500)**3 +
      (9.30520000000000079e-07)*(x-9500)**2 &
    + (8.2660437681159364e-04)*(x-9500) + (2.4700000000000000
      e-01)
  END IF
  IF ((x>=10000).AND.(x<11000))THEN
    qnoselow = (-3.9164985507245381e-11)*(x-10000)**3 +
      (1.1077337391304330e-06)*(x-10000)**2 &
    + (1.8457312463768122e-03)*(x-10000) +
      (9.0769999999999995e-01)
  END IF
  IF ((x>=11000).AND.(x<12000))THEN
    qnoselow = (9.0574492753606305e-12)*(x-11000)**3 +
      (9.9023878260869750e-07)*(x-11000)**2 &

```

```

+ (3.9437037681159422e-03)*(x-11000) +
    (3.8220000000000001e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
    qnoselow = (-2.1689855072467357e-12)*(x-12000)**3 +
        (1.0174111304347832e-06)*(x-12000)**2 &
+ (5.9513536811594192e-03)*(x-12000) +
    (8.7650000000000006e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
    qnoselow = (-2.3441202898550733e-11)*(x-14000)**3 +
        (1.0043972173913060e-06)*(x-14000)**2 &
+ (9.9949703768115912e-03)*(x-14000) +
    (2.4719999999999999e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
    qnoselow = (-2.3441202898553379e-11)*(x-16000)**3 +
        (8.6375000000000264e-07)*(x-16000)**2 &
+ (1.3731264811594207e-02)*(x-16000) +
    (4.8539999999999999e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
    qnosehigh = (1.0982222222222222e-05)*(x) +
        (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
    qnosehigh = (2.15586666666667148e-11)*(x-9000)**3 +
        (1.2769419999999990e-06)*(x-9000)**2 &

```

```

+ (-3.73406666666666246e-05)*(x-9000) +
(9.8839999999999997e-02)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (2.15586666666665032e-11)*(x-9500)**3 +
(1.30928000000000017e-06)*(x-9500)**2 &
+ (1.2557703333333332e-03)*(x-9500) + (4.0210000000000001
e-01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (-8.6837333333331995e-11)*(x-10000)**3 +
(1.34161799999999980e-06)*(x-10000)**2 &
+ (2.5812193333333336e-03)*(x-10000) +
(1.36000000000000001e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (1.8950666666666521e-11)*(x-11000)**3 +
(1.08110600000000009e-06)*(x-11000)**2 &
+ (5.0039433333333334e-03)*(x-11000) +
(5.1959999999999997e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (-7.59808333333334867e-11)*(x-12000)**3 +
(1.13795800000000022e-06)*(x-12000)**2 &
+ (7.22300733333333349e-03)*(x-12000) +
(1.13000000000000001e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN

```

```

      qnosehigh = (1.19446166666666876e-10)*(x-14000)**3 +
        (6.8207299999999925e-07)*(x-14000)**2 &
+ (1.0863069333333326e-02)*(x-14000) +
        (2.96900000000000001e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
      qnosehigh = (1.19446166666666284e-10)*(x-16000)**3 +
        (1.39875000000000004e-06)*(x-16000)**2 &
+ (1.5024715333333348e-02)*(x-16000) +
        (5.51000000000000001e+01)
END IF
      qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<9000))THEN
      qnoselow = (1.6488888888888890e-06)*(x) +
        (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
      qnoselow = (6.8925391304346964e-11)*(x-9000)**3 +
        (2.3965191304347959e-07)*(x-9000)**2 &
+ (2.6282695652173449e-05)*(x-9000) + (1.48400000000000001
        e-02)
END IF
IF ((x>=9500).AND.(x<10000))THEN
      qnoselow = (6.8925391304348127e-11)*(x-9500)**3 +
        (3.4303999999999956e-07)*(x-9500)**2 &
+ (3.1762865217391326e-04)*(x-9500) + (9.650999999999999
        e-02)
END IF

```

```

IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (2.5092173913042931e-12)*(x-10000)**3 +
    (4.4642808695652170e-07)*(x-10000)**2 &
+ (7.1236269565217391e-04)*(x-10000) +
    (3.4970000000000001e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (7.2977391304346795e-12)*(x-11000)**3 +
    (4.5395573913043505e-07)*(x-11000)**2 &
+ (1.6127465217391302e-03)*(x-11000) +
    (1.5109999999999999e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (9.5627173913041786e-12)*(x-12000)**3 +
    (4.7584895652173957e-07)*(x-12000)**2 &
+ (2.5425512173913043e-03)*(x-12000) +
    (3.5850000000000000e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnoselow = (-1.1787543478260328e-11)*(x-14000)**3 +
    (5.3322526086956389e-07)*(x-14000)**2 &
+ (4.5606996521739126e-03)*(x-14000) +
    (1.0650000000000000e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnoselow = (-1.1787543478261837e-11)*(x-16000)**3 +
    (4.6250000000000245e-07)*(x-16000)**2 &

```



```

+ (6.5521501739130442e-03)*(x-16000) +
(2.1809999999999999e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
qnosehigh = (3.3444444444444441e-06)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnosehigh = (5.5820869565218372e-11)*(x-9000)**3 +
(4.1726869565217255e-07)*(x-9000)**2 &
+ (3.9610434782609169e-05)*(x-9000) + (3.0099999999999998
e-02)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnosehigh = (5.5820869565216569e-11)*(x-9500)**3 +
(5.01000000000000058e-07)*(x-9500)**2 &
+ (4.9874478260869548e-04)*(x-9500) + (1.6120000000000001
e-01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnosehigh = (-2.2141739130434676e-11)*(x-10000)**3 +
(5.8473130434782607e-07)*(x-10000)**2 &
+ (1.0416104347826085e-03)*(x-10000) +
(5.4279999999999995e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnosehigh = (4.6086956522748904e-14)*(x-11000)**3 +
(5.1830608695652067e-07)*(x-11000)**2 &

```

```

+ (2.1446478260869566e-03)*(x-11000) +
    (2.1469999999999998e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
    qnosehigh = (-2.0717391304353179e-12)*(x-12000)**3 +
        (5.1844434782608807e-07)*(x-12000)**2 &
+ (3.1813982608695662e-03)*(x-12000) +
    (4.8099999999999996e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
    qnosehigh = (-1.5585652173913201e-11)*(x-14000)**3 +
        (5.0601391304347891e-07)*(x-14000)**2 &
+ (5.2303147826086947e-03)*(x-14000) +
    (1.3230000000000000e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
    qnosehigh = (-1.5585652173913201e-11)*(x-16000)**3 +
        (4.12500000000000050e-07)*(x-16000)**2 &
+ (7.0673426086956519e-03)*(x-16000) +
    (2.5590000000000000e+01)
END IF
    qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
    newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
rndiff = 1.0-(5.000000-nose)/(5.000000-2.300000)
IF ((nose>2.3000000).AND.(nose<=5.00)) THEN
    IF ((x>=0).AND.(x<9000))THEN

```

```

qnoselow = (1.098222222222222e-05)*(x) +
            (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (2.15586666666667148e-11)*(x-9000)**3 +
                (1.2769419999999990e-06)*(x-9000)**2 &
                + (-3.73406666666666246e-05)*(x-9000) +
                (9.883999999999997e-02)
END IF
IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (2.15586666666665032e-11)*(x-9500)**3 +
                (1.3092800000000017e-06)*(x-9500)**2 &
                + (1.2557703333333332e-03)*(x-9500) + (4.0210000000000001
                e-01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
    qnoselow = (-8.68373333333331995e-11)*(x-10000)**3 +
                (1.3416179999999980e-06)*(x-10000)**2 &
                + (2.5812193333333336e-03)*(x-10000) +
                (1.3600000000000001e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
    qnoselow = (1.89506666666666521e-11)*(x-11000)**3 +
                (1.0811060000000009e-06)*(x-11000)**2 &
                + (5.0039433333333334e-03)*(x-11000) +
                (5.195999999999997e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN

```

```

      qnoselow =  (-7.59808333333334867e-11)*(x-12000)**3 +
      (1.13795800000000022e-06)*(x-12000)**2 &
+ (7.22300733333333349e-03)*(x-12000) +
      (1.13000000000000001e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
      qnoselow =  (1.19446166666666876e-10)*(x-14000)**3 +
      (6.8207299999999925e-07)*(x-14000)**2 &
+ (1.08630693333333326e-02)*(x-14000) +
      (2.96900000000000001e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
      qnoselow =  (1.19446166666666284e-10)*(x-16000)**3 +
      (1.39875000000000004e-06)*(x-16000)**2 &
+ (1.50247153333333348e-02)*(x-16000) +
      (5.51000000000000001e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
      qnosehigh = (2.02444444444444444e-05)*(x) +
      (0.00000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
      qnosehigh =  (-8.2005797101450814e-11)*(x-9000)**3 +
      (1.8378086956521752e-06)*(x-9000)**2 &
+ (1.9971014492750070e-06)*(x-9000) + (1.8220000000000000
      e-01)
END IF
IF ((x>=9500).AND.(x<10000))THEN

```

```

    qnosehigh = (-8.2005797101447001e-11)*(x-9500)**3 +
                (1.7147999999999990e-06)*(x-9500)**2 &
+ (1.7783014492753621e-03)*(x-9500) + (6.3239999999999996
    e-01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
    qnosehigh = (-1.7938840579710258e-10)*(x-10000)**3 +
                (1.5917913043478262e-06)*(x-10000)**2 &
+ (3.4315971014492758e-03)*(x-10000) +
                (1.9399999999999999e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
    qnosehigh = (1.9535942028985493e-10)*(x-11000)**3 +
                (1.0536260869565248e-06)*(x-11000)**2 &
+ (6.0770144927536204e-03)*(x-11000) +
                (6.7839999999999998e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
    qnosehigh = (-1.2618840579710134e-10)*(x-12000)**3 +
                (1.6397043478260860e-06)*(x-12000)**2 &
+ (8.7703449275362347e-03)*(x-12000) +
                (1.4109999999999999e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
    qnosehigh = (-3.2512318840580415e-11)*(x-14000)**3 +
                (8.8257391304347698e-07)*(x-14000)**2 &
+ (1.3814901449275363e-02)*(x-14000) +
                (3.7200000000000003e+01)

```

```

END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnosehigh = (-3.2512318840576499e-11)*(x-16000)**3 +
    (6.875000000000000231e-07)*(x-16000)**2 &
+ (1.6955049275362306e-02)*(x-16000) +
    (6.8099999999999994e+01)
END IF
  qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<9000))THEN
  qnoselow = (3.3444444444444441e-06)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnoselow = (5.5820869565218372e-11)*(x-9000)**3 +
    (4.1726869565217255e-07)*(x-9000)**2 &
+ (3.9610434782609169e-05)*(x-9000) + (3.0099999999999998
    e-02)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnoselow = (5.5820869565216569e-11)*(x-9500)**3 +
    (5.01000000000000058e-07)*(x-9500)**2 &
+ (4.9874478260869548e-04)*(x-9500) + (1.6120000000000001
    e-01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (-2.2141739130434676e-11)*(x-10000)**3 +
    (5.8473130434782607e-07)*(x-10000)**2 &

```

```

+ (1.0416104347826085e-03)*(x-10000) +
(5.4279999999999995e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnoselow = (4.6086956522748904e-14)*(x-11000)**3 +
(5.1830608695652067e-07)*(x-11000)**2 &
+ (2.1446478260869566e-03)*(x-11000) +
(2.1469999999999998e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
qnoselow = (-2.0717391304353179e-12)*(x-12000)**3 +
(5.1844434782608807e-07)*(x-12000)**2 &
+ (3.1813982608695662e-03)*(x-12000) +
(4.8099999999999996e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
qnoselow = (-1.5585652173913201e-11)*(x-14000)**3 +
(5.0601391304347891e-07)*(x-14000)**2 &
+ (5.2303147826086947e-03)*(x-14000) +
(1.3230000000000000e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
qnoselow = (-1.5585652173913201e-11)*(x-16000)**3 +
(4.12500000000000050e-07)*(x-16000)**2 &
+ (7.0673426086956519e-03)*(x-16000) +
(2.5590000000000000e+01)
END IF
if ((x>=0).AND.(x<9000))THEN

```

```

qnosehigh = (6.2344444444444448e-06)*(x) +
            (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
    qnosehigh = (1.6877565217388714e-11)*(x-9000)**3 +
                (6.4670365217391630e-07)*(x-9000)**2 &
                + (6.0808782608694712e-05)*(x-9000) + (5.6110000000000000
                e-02)
END IF
IF ((x>=9500).AND.(x<10000))THEN
    qnosehigh = (1.6877565217391680e-11)*(x-9500)**3 +
                (6.7201999999999895e-07)*(x-9500)**2 &
                + (7.2017060869565260e-04)*(x-9500) + (2.5030000000000002
                e-01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
    qnosehigh = (-3.7685130434783418e-11)*(x-10000)**3 +
                (6.9733634782608848e-07)*(x-10000)**2 &
                + (1.4048487826086953e-03)*(x-10000) +
                (7.804999999999997e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
    qnosehigh = (-5.9747043478261916e-11)*(x-11000)**3 +
                (5.8428095652173970e-07)*(x-11000)**2 &
                + (2.6864660869565220e-03)*(x-11000) +
                (2.8450000000000002e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN

```



```

      qnosehigh = (7.1533369565217201e-11)*(x-12000)**3 +
                  (4.0503982608695787e-07)*(x-12000)**2 &
+ (3.6757868695652157e-03)*(x-12000) +
                  (6.0560000000000000e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
      qnosehigh = (-9.3206673913042961e-11)*(x-14000)**3 +
                  (8.3424004347825924e-07)*(x-14000)**2 &
+ (6.1543466086956536e-03)*(x-14000) +
                  (1.5600000000000000e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
      qnosehigh = (-9.3206673913044047e-11)*(x-16000)**3 +
                  (2.75000000000000097e-07)*(x-16000)**2 &
+ (8.3728266956521746e-03)*(x-16000) +
                  (3.0500000000000000e+01)
END IF
      qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
      newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
      rndiff = 1.0-(10.000000-nose)/(10.000000-5.000000)
IF ((nose>5.0000000).AND.(nose<=10.00)) THEN
      IF ((x>=0).AND.(x<9000))THEN
      qnoselow = (2.0244444444444444e-05)*(x) +
                  (0.0000000000000000e+00)
      END IF
      IF ((x>=9000).AND.(x<9500))THEN

```

```

qnoselow = (-8.2005797101450814e-11)*(x-9000)**3 +
            (1.8378086956521752e-06)*(x-9000)**2 &
+ (1.9971014492750070e-06)*(x-9000) + (1.8220000000000000
            e-01)
END IF
IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (-8.2005797101447001e-11)*(x-9500)**3 +
                (1.7147999999999990e-06)*(x-9500)**2 &
+ (1.7783014492753621e-03)*(x-9500) + (6.3239999999999996
            e-01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
    qnoselow = (-1.7938840579710258e-10)*(x-10000)**3 +
                (1.5917913043478262e-06)*(x-10000)**2 &
+ (3.4315971014492758e-03)*(x-10000) +
            (1.9399999999999999e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
    qnoselow = (1.9535942028985493e-10)*(x-11000)**3 +
                (1.0536260869565248e-06)*(x-11000)**2 &
+ (6.0770144927536204e-03)*(x-11000) +
            (6.7839999999999998e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
    qnoselow = (-1.2618840579710134e-10)*(x-12000)**3 +
                (1.6397043478260860e-06)*(x-12000)**2 &
+ (8.7703449275362347e-03)*(x-12000) +
            (1.4109999999999999e+01)

```

```

END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnoselow = (-3.2512318840580415e-11)*(x-14000)**3 +
    (8.8257391304347698e-07)*(x-14000)**2 &
+ (1.3814901449275363e-02)*(x-14000) +
    (3.72000000000000003e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnoselow = (-3.2512318840576499e-11)*(x-16000)**3 +
    (6.87500000000000231e-07)*(x-16000)**2 &
+ (1.6955049275362306e-02)*(x-16000) +
    (6.8099999999999994e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
  qnosehigh = (3.5811111111111109e-05)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (-1.9278579710145095e-10)*(x-9000)**3 +
    (2.4185786956521755e-06)*(x-9000)**2 &
+ (9.5907101449275170e-05)*(x-9000) + (3.2229999999999998
    e-01)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (-1.9278579710144841e-10)*(x-9500)**3 +
    (2.1293999999999995e-06)*(x-9500)**2 &
+ (2.3698964492753624e-03)*(x-9500) + (9.5079999999999998
    e-01)

```

```

END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (-2.9592840579709823e-10)*(x-10000)**3 +
    (1.8402213043478218e-06)*(x-10000)**2 &
+ (4.3547071014492754e-03)*(x-10000) +
    (2.6440000000000001e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (3.6719942028985369e-10)*(x-11000)**3 +
    (9.5243608695652446e-07)*(x-11000)**2 &
+ (7.1473644927536242e-03)*(x-11000) +
    (8.5429999999999993e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (-2.1297590579710074e-10)*(x-12000)**3 +
    (2.0540343478260836e-06)*(x-12000)**2 &
+ (1.0153834927536234e-02)*(x-12000) +
    (1.7010000000000002e+01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (-3.2904818840578967e-11)*(x-14000)**3 +
    (7.7617891304347834e-07)*(x-14000)**2 &
+ (1.5814261449275360e-02)*(x-14000) +
    (4.3829999999999998e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnosehigh = (-3.2904818840580693e-11)*(x-16000)**3 +
    (5.7875000000000124e-07)*(x-16000)**2 &

```

```

+ (1.8524119275362325e-02)*(x-16000) +
(7.8299999999999997e+01)
END IF

qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<9000))THEN
qnoselow = (6.2344444444444448e-06)*(x) +
(0.0000000000000000e+00)
END IF

IF ((x>=9000).AND.(x<9500))THEN
qnoselow = (1.6877565217388714e-11)*(x-9000)**3 +
(6.4670365217391630e-07)*(x-9000)**2 &
+ (6.0808782608694712e-05)*(x-9000) + (5.6110000000000000
e-02)
END IF

IF ((x>=9500).AND.(x<10000))THEN
qnoselow = (1.6877565217391680e-11)*(x-9500)**3 +
(6.72019999999999895e-07)*(x-9500)**2 &
+ (7.2017060869565260e-04)*(x-9500) + (2.5030000000000002
e-01)
END IF

IF ((x>=10000).AND.(x<11000))THEN
qnoselow = (-3.7685130434783418e-11)*(x-10000)**3 +
(6.9733634782608848e-07)*(x-10000)**2 &
+ (1.4048487826086953e-03)*(x-10000) +
(7.8049999999999997e-01)
END IF

IF ((x>=11000).AND.(x<12000))THEN

```

```

      qnoselow =  (-5.9747043478261916e-11)*(x-11000)**3 +
                  (5.8428095652173970e-07)*(x-11000)**2 &
+   (2.6864660869565220e-03)*(x-11000) +
                  (2.8450000000000002e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
      qnoselow =  (7.1533369565217201e-11)*(x-12000)**3 +
                  (4.0503982608695787e-07)*(x-12000)**2 &
+   (3.6757868695652157e-03)*(x-12000) +
                  (6.0560000000000000e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
      qnoselow =  (-9.3206673913042961e-11)*(x-14000)**3 +
                  (8.3424004347825924e-07)*(x-14000)**2 &
+   (6.1543466086956536e-03)*(x-14000) +
                  (1.5600000000000000e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
      qnoselow =  (-9.3206673913044047e-11)*(x-16000)**3 +
                  (2.75000000000000097e-07)*(x-16000)**2 &
+   (8.3728266956521746e-03)*(x-16000) +
                  (3.0500000000000000e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
      qnosehigh = (1.0900000000000001e-05)*(x) +
                  (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN

```

```

qnosehigh = (-3.1883188405796613e-11)*(x-9000)**3 +
(9.0362478260869516e-07)*(x-9000)**2 &
+ (9.9158405797101557e-05)*(x-9000) + (9.8100000000000007
e-02)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnosehigh = (-3.1883188405796186e-11)*(x-9500)**3 +
(8.5579999999999950e-07)*(x-9500)**2 &
+ (9.7887079710144926e-04)*(x-9500) + (3.695999999999998
e-01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnosehigh = (-9.7733623188405880e-11)*(x-10000)**3 +
(8.0797521739130405e-07)*(x-10000)**2 &
+ (1.8107584057971016e-03)*(x-10000) +
(1.0690000000000000e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnosehigh = (1.0171768115942038e-10)*(x-11000)**3 +
(5.1477434782608735e-07)*(x-11000)**2 &
+ (3.1335079710144921e-03)*(x-11000) +
(3.5899999999999999e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
qnosehigh = (-6.0766123188406390e-11)*(x-12000)**3 +
(8.1992739130434902e-07)*(x-12000)**2 &
+ (4.4682097101449280e-03)*(x-12000) +
(7.3399999999999999e+00)

```

```

END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (-2.8596775362318268e-11)*(x-14000)**3 +
    (4.5533065217391325e-07)*(x-14000)**2 &
+ (7.0187257971014473e-03)*(x-14000) +
    (1.9070000000000000e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnosehigh = (-2.8596775362320453e-11)*(x-16000)**3 +
    (2.83750000000000014e-07)*(x-16000)**2 &
+ (8.4968871014492811e-03)*(x-16000) +
    (3.47000000000000003e+01)
END IF
  qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
  newqnr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if

END IF !density if
!!!!!!!!!!!!!!!!!!!!!!!!NEW ALTITUDE RANGE!!!!!!!!!!!!!!!!!!!!!!!!

IF ((rho<=0.0001471300).AND.(rho>=0.0000665930)) THEN
  rhodiff = (0.0001471300-rho)/(0.0001471300-0.0000665930)
  rndiff = 1.0-(0.100000-nose)/(0.100000-0.049999)
IF ((nose>0.049999).AND.(nose<=0.10)) THEN
  IF ((x>=0).AND.(x<9000))THEN
    qnoselow = (1.1844444444444446e-07)*(x) +
      (0.0000000000000000e+00)

```



```

END IF
IF ((x >= 9000) .AND. (x < 9500)) THEN
    qnoselow = (1.3391959420289862e-11)*(x-9000)**3 +
        (3.8584060869565191e-08)*(x-9000)**2 &
+ (-9.3202028985506223e-07)*(x-9000) +
        (1.06600000000000001e-03)
END IF
IF ((x >= 9500) .AND. (x < 10000)) THEN
    qnoselow = (1.3391959420289889e-11)*(x-9500)**3 +
        (5.8671999999999998e-08)*(x-9500)**2 &
+ (4.7696010144927526e-05)*(x-9500) + (1.1920000000000000
        e-02)
END IF
IF ((x >= 10000) .AND. (x < 11000)) THEN
    qnoselow = (1.1818081159420293e-11)*(x-10000)**3 +
        (7.8759939130434765e-08)*(x-10000)**2 &
+ (1.1641197971014494e-04)*(x-10000) +
        (5.2109999999999997e-02)
END IF
IF ((x >= 11000) .AND. (x < 12000)) THEN
    qnoselow = (6.4997159420288411e-12)*(x-11000)**3 +
        (1.1421418260869587e-07)*(x-11000)**2 &
+ (3.0938610144927534e-04)*(x-11000) +
        (2.5910000000000000e-01)
END IF
IF ((x >= 12000) .AND. (x < 14000)) THEN
    qnoselow = (8.2899311594203144e-12)*(x-12000)**3 +
        (1.3371333043478252e-07)*(x-12000)**2 &

```

```

+ (5.5731361449275359e-04)*(x-12000) +
(6.8920000000000003e-01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnoselow = (-1.5137986231883886e-11)*(x-14000)**3 +
(1.8345291739130403e-07)*(x-14000)**2 &
+ (1.1916461101449274e-03)*(x-14000) +
(2.4049999999999998e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnoselow = (-1.5137986231884381e-11)*(x-16000)**3 +
(9.2625000000000368e-08)*(x-16000)**2 &
+ (1.7438019449275369e-03)*(x-16000) +
(5.4009999999999998e+00)
END IF
if ((x>=0).AND.(x<9000))THEN
  qnosehigh = (2.1711111111111111e-07)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (2.5348968115942017e-11)*(x-9000)**3 +
(5.2444547826086976e-08)*(x-9000)**2 &
+ (3.1324840579710078e-06)*(x-9000) + (1.9540000000000000
e-03)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (2.5348968115941962e-11)*(x-9500)**3 +
(9.0468000000000026e-08)*(x-9500)**2 &

```

```

+ (7.4588757971014497e-05)*(x-9500) + (1.9800000000000002
e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (1.6160063768116075e-11)*(x-10000)**3 +
    (1.2849145217391294e-07)*(x-10000)**2 &
+ (1.8406848405797099e-04)*(x-10000) +
    (8.2879999999999995e-02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (1.0896776811594468e-11)*(x-11000)**3 +
    (1.7697164347826043e-07)*(x-11000)**2 &
+ (4.8953157971014509e-04)*(x-11000) +
    (4.11600000000000002e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (1.1627713768115993e-11)*(x-12000)**3 +
    (2.0966197391304339e-07)*(x-12000)**2 &
+ (8.7616519710144935e-04)*(x-12000) +
    (1.08900000000000000e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (-1.5800542753623243e-11)*(x-14000)**3 +
    (2.7942825652173935e-07)*(x-14000)**2 &
+ (1.8543456579710148e-03)*(x-14000) +
    (3.77300000000000001e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN

```

```

qnosehigh = (-1.5800542753622926e-11)*(x-16000)**3 +
(1.8462499999999905e-07)*(x-16000)**2 &
+ (2.7824521710144933e-03)*(x-16000) +
(8.4730000000000008e+00)
END IF

qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<9000))THEN
qnoselow = (2.9100000000000002e-08)*(x) +
(0.0000000000000000e+00)
END IF

IF ((x>=9000).AND.(x<9500))THEN
qnoselow = (1.7183663768115799e-12)*(x-9000)**3 +
(1.5786250434782634e-08)*(x-9000)**2 &
+ (-2.0065168115942117e-06)*(x-9000) +
(2.6190000000000002e-04)
END IF

IF ((x>=9500).AND.(x<10000))THEN
qnoselow = (1.7183663768115799e-12)*(x-9500)**3 +
(1.8363800000000001e-08)*(x-9500)**2 &
+ (1.5068508405797105e-05)*(x-9500) + (3.419999999999999
e-03)
END IF

IF ((x>=10000).AND.(x<11000))THEN
qnoselow = (4.3075672463767870e-12)*(x-10000)**3 +
(2.0941349565217433e-08)*(x-10000)**2 &
+ (3.4721083188405791e-05)*(x-10000) +
(1.5760000000000000e-02)
END IF

```

```

IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (3.7946463768116607e-13)*(x-11000)**3 +
    (3.3864051304347836e-08)*(x-11000)**2 &
  + (8.9526484057971017e-05)*(x-11000) +
    (7.5730000000000006e-02)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (2.3380322463768221e-12)*(x-12000)**3 +
    (3.5002445217391268e-08)*(x-12000)**2 &
  + (1.5839298057971019e-04)*(x-12000) +
    (1.9950000000000001e-01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnoselow = (-4.1301064492753434e-12)*(x-14000)**3 +
    (4.9030638695652153e-08)*(x-14000)**2 &
  + (3.2645914840579712e-04)*(x-14000) +
    (6.7500000000000004e-01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnoselow = (-4.1301064492754904e-12)*(x-16000)**3 +
    (2.4250000000000128e-08)*(x-16000)**2 &
  + (4.7302042579710162e-04)*(x-16000) +
    (1.4910000000000001e+00)
END IF
if ((x>=0).AND.(x<9000))THEN
  qnosehigh = (5.3211111111111107e-08)*(x) +
    (0.0000000000000000e+00)
END IF

```

```

IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (4.4343953623188417e-12)*(x-9000)**3 +
    (2.1550206956521730e-08)*(x-9000)**2 &
+ (-1.3435023188405755e-06)*(x-9000) +
    (4.7889999999999999e-04)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (4.4343953623188611e-12)*(x-9500)**3 +
    (2.8201799999999993e-08)*(x-9500)**2 &
+ (2.3532501159420286e-05)*(x-9500) + (5.7489999999999998
    e-03)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (5.3665092753623151e-12)*(x-10000)**3 +
    (3.4853393043478258e-08)*(x-10000)**2 &
+ (5.5060097681159424e-05)*(x-10000) +
    (2.5120000000000000e-02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (2.2806675362318380e-12)*(x-11000)**3 +
    (5.0952920869565282e-08)*(x-11000)**2 &
+ (1.4086641159420288e-04)*(x-11000) +
    (1.2039999999999999e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (4.2614742753623443e-12)*(x-12000)**3 +
    (5.7794923478260829e-08)*(x-12000)**2 &

```

```

+ (2.4961425594202896e-04)*(x-12000) +
(3.1450000000000000e-01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (-7.2897948550724195e-12)*(x-14000)**3 +
(8.3363769130434670e-08)*(x-14000)**2 &
+ (5.3193164115942040e-04)*(x-14000) +
(1.0790000000000000e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnosehigh = (-7.2897948550725811e-12)*(x-16000)**3 +
(3.96250000000000052e-08)*(x-16000)**2 &
+ (7.7790917942029006e-04)*(x-16000) +
(2.4180000000000001e+00)
END IF
  qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
  newqr = qnose2*rhodiff+qnosel*(1.0-rhodiff)
END IF !nose radius if
  rndiff = 1.0-(0.230000-nose)/(0.230000-0.100000)
IF ((nose>0.1000000).AND.(nose<=0.23)) THEN
  IF ((x>=0).AND.(x<9000))THEN
    qnoselow = (2.1711111111111111e-07)*(x) +
(0.0000000000000000e+00)
  END IF
  IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (2.5348968115942017e-11)*(x-9000)**3 +
(5.2444547826086976e-08)*(x-9000)**2 &

```

```

+ (3.1324840579710078e-06)*(x-9000) + (1.9540000000000000
e-03)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnoselow = (2.5348968115941962e-11)*(x-9500)**3 +
    (9.04680000000000026e-08)*(x-9500)**2 &
+ (7.4588757971014497e-05)*(x-9500) + (1.98000000000000002
e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (1.6160063768116075e-11)*(x-10000)**3 +
    (1.2849145217391294e-07)*(x-10000)**2 &
+ (1.8406848405797099e-04)*(x-10000) +
    (8.2879999999999995e-02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (1.0896776811594468e-11)*(x-11000)**3 +
    (1.7697164347826043e-07)*(x-11000)**2 &
+ (4.8953157971014509e-04)*(x-11000) +
    (4.11600000000000002e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (1.1627713768115993e-11)*(x-12000)**3 +
    (2.0966197391304339e-07)*(x-12000)**2 &
+ (8.7616519710144935e-04)*(x-12000) +
    (1.08900000000000000e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN

```



```

      qnoselow =  (-1.5800542753623243e-11)*(x-14000)**3 +
                  (2.7942825652173935e-07)*(x-14000)**2 &
+   (1.8543456579710148e-03)*(x-14000) +
                  (3.7730000000000001e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
      qnoselow =  (-1.5800542753622926e-11)*(x-16000)**3 +
                  (1.8462499999999905e-07)*(x-16000)**2 &
+   (2.7824521710144933e-03)*(x-16000) +
                  (8.4730000000000008e+00)
END IF
if ((x>=0).AND.(x<9000))THEN
      qnosehigh =  (4.5366666666666664e-07)*(x) +
                  (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
      qnosehigh =  (4.1923443478260443e-11)*(x-9000)**3 +
                  (9.1000834782609251e-08)*(x-9000)**2 &
+   (7.1927217391302702e-06)*(x-9000) + (4.0829999999999998
      e-03)
END IF
IF ((x>=9500).AND.(x<10000))THEN
      qnosehigh =  (4.1923443478260785e-11)*(x-9500)**3 +
                  (1.5388599999999987e-07)*(x-9500)**2 &
+   (1.2963613913043485e-04)*(x-9500) + (3.5670000000000000
      e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN

```

```

qnosehigh = (2.0864113043478208e-11)*(x-10000)**3 +
(2.1677116521739147e-07)*(x-10000)**2 &
+ (3.1496472173913031e-04)*(x-10000) +
(1.4419999999999999e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnosehigh = (1.2737104347825845e-11)*(x-11000)**3 +
(2.7936350434782639e-07)*(x-11000)**2 &
+ (8.1109939130434789e-04)*(x-11000) +
(6.9679999999999997e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
qnosehigh = (1.6703163043478249e-11)*(x-12000)**3 +
(3.1757481739130435e-07)*(x-12000)**2 &
+ (1.4080377130434782e-03)*(x-12000) +
(1.8000000000000000e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
qnosehigh = (-1.2340632608695035e-11)*(x-14000)**3 +
(4.1779379565217269e-07)*(x-14000)**2 &
+ (2.8787749391304346e-03)*(x-14000) +
(6.0199999999999996e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
qnosehigh = (-1.2340632608696332e-11)*(x-16000)**3 +
(3.43750000000000031e-07)*(x-16000)**2 &
+ (4.4018625304347849e-03)*(x-16000) +
(1.3350000000000000e+01)

```

```

END IF

  qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
  if ((x>=0).AND.(x<9000))THEN
    qnoselow = (5.3211111111111107e-08)*(x) +
      (0.0000000000000000e+00)
  END IF

  IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (4.4343953623188417e-12)*(x-9000)**3 +
      (2.1550206956521730e-08)*(x-9000)**2 &
    + (-1.3435023188405755e-06)*(x-9000) +
      (4.7889999999999999e-04)
  END IF

  IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (4.4343953623188611e-12)*(x-9500)**3 +
      (2.8201799999999993e-08)*(x-9500)**2 &
    + (2.3532501159420286e-05)*(x-9500) + (5.7489999999999998
      e-03)
  END IF

  IF ((x>=10000).AND.(x<11000))THEN
    qnoselow = (5.3665092753623151e-12)*(x-10000)**3 +
      (3.4853393043478258e-08)*(x-10000)**2 &
    + (5.5060097681159424e-05)*(x-10000) +
      (2.5120000000000000e-02)
  END IF

  IF ((x>=11000).AND.(x<12000))THEN
    qnoselow = (2.2806675362318380e-12)*(x-11000)**3 +
      (5.0952920869565282e-08)*(x-11000)**2 &

```

```

+ (1.4086641159420288e-04)*(x-11000) +
(1.2039999999999999e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
qnoselow = (4.2614742753623443e-12)*(x-12000)**3 +
(5.7794923478260829e-08)*(x-12000)**2 &
+ (2.4961425594202896e-04)*(x-12000) +
(3.1450000000000000e-01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
qnoselow = (-7.2897948550724195e-12)*(x-14000)**3 +
(8.3363769130434670e-08)*(x-14000)**2 &
+ (5.3193164115942040e-04)*(x-14000) +
(1.0790000000000000e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
qnoselow = (-7.2897948550725811e-12)*(x-16000)**3 +
(3.96250000000000052e-08)*(x-16000)**2 &
+ (7.7790917942029006e-04)*(x-16000) +
(2.4180000000000001e+00)
END IF
if ((x>=0).AND.(x<9000))THEN
qnosehigh = (1.1155555555555554e-07)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnosehigh = (8.5983304347825429e-12)*(x-9000)**3 +
(3.6010504347826203e-08)*(x-9000)**2 &

```

```

+ (-6.4283478260873661e-07)*(x-9000) +
  (1.0039999999999999e-03)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (8.5983304347825300e-12)*(x-9500)**3 +
    (4.89080000000000010e-08)*(x-9500)**2 &
+ (4.1816417391304375e-05)*(x-9500) + (1.0760000000000000
  e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (6.9513391304347948e-12)*(x-10000)**3 +
    (6.1805495652173930e-08)*(x-10000)**2 &
+ (9.7173165217391281e-05)*(x-10000) +
    (4.49700000000000003e-02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (6.2023130434781771e-12)*(x-11000)**3 +
    (8.2659513043478314e-08)*(x-11000)**2 &
+ (2.4163817391304352e-04)*(x-11000) +
    (2.10900000000000000e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (6.1757391304348561e-12)*(x-12000)**3 +
    (1.0126645217391296e-07)*(x-12000)**2 &
+ (4.2556413913043468e-04)*(x-12000) +
    (5.4139999999999999e-01)
END IF
IF ((x>=14000).AND.(x<16000))THEN

```

```

      qnosehigh = (-7.9701478260870283e-12)*(x-14000)**3 +
      (1.3832088695652161e-07)*(x-14000)**2 &
+ (9.0473881739130480e-04)*(x-14000) +
      (1.8470000000000000e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
      qnosehigh = (-7.9701478260865420e-12)*(x-16000)**3 +
      (9.0499999999999546e-08)*(x-16000)**2 &
+ (1.3623805913043469e-03)*(x-16000) +
      (4.1459999999999999e+00)
END IF
      qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
      newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
      rndiff = 1.0-(0.500000-nose)/(0.500000-0.230000)
IF ((nose>0.2300000).AND.(nose<=0.50)) THEN
      IF ((x>=0).AND.(x<9000))THEN
      qnoselow = (4.5366666666666664e-07)*(x) +
      (0.0000000000000000e+00)
END IF
      IF ((x>=9000).AND.(x<9500))THEN
      qnoselow = (4.1923443478260443e-11)*(x-9000)**3 +
      (9.1000834782609251e-08)*(x-9000)**2 &
+ (7.1927217391302702e-06)*(x-9000) + (4.0829999999999998
      e-03)
END IF
      IF ((x>=9500).AND.(x<10000))THEN

```

```

qnoselow = (4.1923443478260785e-11)*(x-9500)**3 +
            (1.5388599999999987e-07)*(x-9500)**2 &
+ (1.2963613913043485e-04)*(x-9500) + (3.5670000000000000
            e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnoselow = (2.0864113043478208e-11)*(x-10000)**3 +
            (2.1677116521739147e-07)*(x-10000)**2 &
+ (3.1496472173913031e-04)*(x-10000) +
            (1.4419999999999999e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnoselow = (1.2737104347825845e-11)*(x-11000)**3 +
            (2.7936350434782639e-07)*(x-11000)**2 &
+ (8.1109939130434789e-04)*(x-11000) +
            (6.9679999999999997e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
qnoselow = (1.6703163043478249e-11)*(x-12000)**3 +
            (3.1757481739130435e-07)*(x-12000)**2 &
+ (1.4080377130434782e-03)*(x-12000) +
            (1.8000000000000000e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
qnoselow = (-1.2340632608695035e-11)*(x-14000)**3 +
            (4.1779379565217269e-07)*(x-14000)**2 &
+ (2.8787749391304346e-03)*(x-14000) +
            (6.0199999999999996e+00)

```

```

END IF
IF ((x>=16000).AND.(x<18000))THEN
    qnoselow = (-1.2340632608696332e-11)*(x-16000)**3 +
        (3.4375000000000031e-07)*(x-16000)**2 &
+ (4.4018625304347849e-03)*(x-16000) +
        (1.3350000000000000e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
    qnosehigh = (9.0077777777777771e-07)*(x) +
        (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
    qnosehigh = (5.2910121739131146e-11)*(x-9000)**3 +
        (1.6344881739130341e-07)*(x-9000)**2 &
+ (1.0234060869565519e-05)*(x-9000) + (8.1069999999999996
        e-03)
END IF
IF ((x>=9500).AND.(x<10000))THEN
    qnosehigh = (5.2910121739130086e-11)*(x-9500)**3 +
        (2.4281400000000042e-07)*(x-9500)**2 &
+ (2.1336546956521728e-04)*(x-9500) + (6.0699999999999997
        e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
    qnosehigh = (2.1258756521738738e-11)*(x-10000)**3 +
        (3.2217918260869607e-07)*(x-10000)**2 &
+ (4.9586206086956526e-04)*(x-10000) +
        (2.3469999999999999e-01)

```



```

END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (8.0478521739127639e-12)*(x-11000)**3 +
    (3.8595545217391367e-07)*(x-11000)**2 &
+ (1.2039966956521736e-03)*(x-11000) +
    (1.0740000000000001e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (1.7687706521738982e-11)*(x-12000)**3 +
    (4.1009900869565238e-07)*(x-12000)**2 &
+ (2.0000511565217392e-03)*(x-12000) +
    (2.6720000000000002e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (-1.0537541304348163e-11)*(x-14000)**3 +
    (5.1622524782608818e-07)*(x-14000)**2 &
+ (3.8526996695652164e-03)*(x-14000) +
    (8.4540000000000006e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnosehigh = (-1.0537541304347184e-11)*(x-16000)**3 +
    (4.5299999999999856e-07)*(x-16000)**2 &
+ (5.7911501652173911e-03)*(x-16000) +
    (1.8140000000000001e+01)
END IF
  qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<9000))THEN

```

```

qnoselow = (1.1155555555555554e-07)*(x) +
            (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (8.5983304347825429e-12)*(x-9000)**3 +
                (3.6010504347826203e-08)*(x-9000)**2 &
                + (-6.4283478260873661e-07)*(x-9000) +
                (1.0039999999999999e-03)
END IF
IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (8.5983304347825300e-12)*(x-9500)**3 +
                (4.8908000000000010e-08)*(x-9500)**2 &
                + (4.1816417391304375e-05)*(x-9500) + (1.0760000000000000
                e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
    qnoselow = (6.9513391304347948e-12)*(x-10000)**3 +
                (6.1805495652173930e-08)*(x-10000)**2 &
                + (9.7173165217391281e-05)*(x-10000) +
                (4.4970000000000003e-02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
    qnoselow = (6.2023130434781771e-12)*(x-11000)**3 +
                (8.2659513043478314e-08)*(x-11000)**2 &
                + (2.4163817391304352e-04)*(x-11000) +
                (2.1090000000000000e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN

```

```

      qnoselow = (6.1757391304348561e-12)*(x-12000)**3 +
      (1.0126645217391296e-07)*(x-12000)**2 &
+ (4.2556413913043468e-04)*(x-12000) +
      (5.413999999999999e-01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
      qnoselow = (-7.9701478260870283e-12)*(x-14000)**3 +
      (1.3832088695652161e-07)*(x-14000)**2 &
+ (9.0473881739130480e-04)*(x-14000) +
      (1.8470000000000000e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
      qnoselow = (-7.9701478260865420e-12)*(x-16000)**3 +
      (9.04999999999999546e-08)*(x-16000)**2 &
+ (1.3623805913043469e-03)*(x-16000) +
      (4.145999999999999e+00)
END IF
if ((x>=0).AND.(x<9000))THEN
      qnosehigh = (2.2399999999999999e-07)*(x) +
      (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
      qnosehigh = (1.0452944927536205e-11)*(x-9000)**3 +
      (6.6412582608695736e-08)*(x-9000)**2 &
+ (-2.0315275362319251e-06)*(x-9000) +
      (2.0160000000000000e-03)
END IF
IF ((x>=9500).AND.(x<10000))THEN

```

```

qnosehigh = (1.0452944927536205e-11)*(x-9500)**3 +
(8.2091999999999978e-08)*(x-9500)**2 &
+ (7.2220763768115969e-05)*(x-9500) + (1.8910000000000000
e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnosehigh = (9.5261101449275267e-12)*(x-10000)**3 +
(9.7771417391304340e-08)*(x-10000)**2 &
+ (1.6215247246376810e-04)*(x-10000) +
(7.6850000000000002e-02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnosehigh = (8.4766144927535892e-12)*(x-11000)**3 +
(1.2634974782608699e-07)*(x-11000)**2 &
+ (3.8627363768115936e-04)*(x-11000) +
(3.4630000000000000e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
qnosehigh = (7.9594601449274952e-12)*(x-12000)**3 +
(1.5177959130434802e-07)*(x-12000)**2 &
+ (6.6440297681159409e-04)*(x-12000) +
(8.6739999999999995e-01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
qnosehigh = (-7.1518920289855623e-12)*(x-14000)**3 +
(1.9953635217391302e-07)*(x-14000)**2 &
+ (1.3670348637681161e-03)*(x-14000) +
(2.8670000000000000e+00)

```

```

END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnosehigh = (-7.1518920289853506e-12)*(x-16000)**3 +
    (1.5662500000000009e-07)*(x-16000)**2 &
+ (2.0793575681159414e-03)*(x-16000) +
    (6.3419999999999996e+00)
END IF
  qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
  newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
  rndiff = 1.0-(1.000000-nose)/(1.000000-0.500000)
IF ((nose>0.5000000).AND.(nose<=1.00)) THEN
  IF ((x>=0).AND.(x<9000))THEN
    qnoselow = (9.0077777777777771e-07)*(x) +
      (0.0000000000000000e+00)
  END IF
  IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (5.2910121739131146e-11)*(x-9000)**3 +
      (1.6344881739130341e-07)*(x-9000)**2 &
+ (1.0234060869565519e-05)*(x-9000) + (8.1069999999999996
      e-03)
  END IF
  IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (5.2910121739130086e-11)*(x-9500)**3 +
      (2.42814000000000042e-07)*(x-9500)**2 &
+ (2.1336546956521728e-04)*(x-9500) + (6.0699999999999997
      e-02)
  END IF

```

```

IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (2.1258756521738738e-11)*(x-10000)**3 +
    (3.2217918260869607e-07)*(x-10000)**2 &
+ (4.9586206086956526e-04)*(x-10000) +
    (2.3469999999999999e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (8.0478521739127639e-12)*(x-11000)**3 +
    (3.8595545217391367e-07)*(x-11000)**2 &
+ (1.2039966956521736e-03)*(x-11000) +
    (1.07400000000000001e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (1.7687706521738982e-11)*(x-12000)**3 +
    (4.1009900869565238e-07)*(x-12000)**2 &
+ (2.0000511565217392e-03)*(x-12000) +
    (2.67200000000000002e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnoselow = (-1.0537541304348163e-11)*(x-14000)**3 +
    (5.1622524782608818e-07)*(x-14000)**2 &
+ (3.8526996695652164e-03)*(x-14000) +
    (8.45400000000000006e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnoselow = (-1.0537541304347184e-11)*(x-16000)**3 +
    (4.5299999999999856e-07)*(x-16000)**2 &

```

```

+ (5.7911501652173911e-03)*(x-16000) +
(1.8140000000000001e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
qnosehigh = (1.6488888888888890e-06)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnosehigh = (6.8925391304346964e-11)*(x-9000)**3 +
(2.3965191304347959e-07)*(x-9000)**2 &
+ (2.6282695652173449e-05)*(x-9000) + (1.4840000000000001
e-02)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnosehigh = (6.8925391304348127e-11)*(x-9500)**3 +
(3.4303999999999956e-07)*(x-9500)**2 &
+ (3.1762865217391326e-04)*(x-9500) + (9.650999999999999
e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnosehigh = (2.5092173913042931e-12)*(x-10000)**3 +
(4.4642808695652170e-07)*(x-10000)**2 &
+ (7.1236269565217391e-04)*(x-10000) +
(3.4970000000000001e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnosehigh = (7.2977391304346795e-12)*(x-11000)**3 +
(4.5395573913043505e-07)*(x-11000)**2 &

```

```

+ (1.6127465217391302e-03)*(x-11000) +
  (1.5109999999999999e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (9.5627173913041786e-12)*(x-12000)**3 +
    (4.7584895652173957e-07)*(x-12000)**2 &
+ (2.5425512173913043e-03)*(x-12000) +
  (3.5850000000000000e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (-1.1787543478260328e-11)*(x-14000)**3 +
    (5.3322526086956389e-07)*(x-14000)**2 &
+ (4.5606996521739126e-03)*(x-14000) +
  (1.0650000000000000e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnosehigh = (-1.1787543478261837e-11)*(x-16000)**3 +
    (4.62500000000000245e-07)*(x-16000)**2 &
+ (6.5521501739130442e-03)*(x-16000) +
  (2.1809999999999999e+01)
END IF
  qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
  if ((x>=0).AND.(x<9000))THEN
    qnoselow = (2.2399999999999999e-07)*(x) +
      (0.0000000000000000e+00)
  END IF
  IF ((x>=9000).AND.(x<9500))THEN

```



```

qnoselow = (1.0452944927536205e-11)*(x-9000)**3 +
            (6.6412582608695736e-08)*(x-9000)**2 &
+ (-2.0315275362319251e-06)*(x-9000) +
            (2.0160000000000000e-03)
END IF
IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (1.0452944927536205e-11)*(x-9500)**3 +
                (8.2091999999999978e-08)*(x-9500)**2 &
+ (7.2220763768115969e-05)*(x-9500) + (1.8910000000000000
    e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
    qnoselow = (9.5261101449275267e-12)*(x-10000)**3 +
                (9.7771417391304340e-08)*(x-10000)**2 &
+ (1.6215247246376810e-04)*(x-10000) +
                (7.6850000000000002e-02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
    qnoselow = (8.4766144927535892e-12)*(x-11000)**3 +
                (1.2634974782608699e-07)*(x-11000)**2 &
+ (3.8627363768115936e-04)*(x-11000) +
                (3.4630000000000000e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
    qnoselow = (7.9594601449274952e-12)*(x-12000)**3 +
                (1.5177959130434802e-07)*(x-12000)**2 &
+ (6.6440297681159409e-04)*(x-12000) +
                (8.6739999999999995e-01)

```

```

END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnoselow = (-7.1518920289855623e-12)*(x-14000)**3 +
    (1.9953635217391302e-07)*(x-14000)**2 &
+ (1.3670348637681161e-03)*(x-14000) +
    (2.8670000000000000e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnoselow = (-7.1518920289853506e-12)*(x-16000)**3 +
    (1.5662500000000009e-07)*(x-16000)**2 &
+ (2.0793575681159414e-03)*(x-16000) +
    (6.3419999999999996e+00)
END IF
if ((x>=0).AND.(x<9000))THEN
  qnosehigh = (4.1766666666666669e-07)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (8.7932579710142953e-12)*(x-9000)**3 +
    (1.0868811304347852e-07)*(x-9000)**2 &
+ (-5.4037101449283536e-07)*(x-9000) +
    (3.75900000000000002e-03)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (8.7932579710145603e-12)*(x-9500)**3 +
    (1.2187799999999990e-07)*(x-9500)**2 &
+ (1.1474268550724641e-04)*(x-9500) + (3.1759999999999997
    e-02)

```

```

END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (1.4116484057970909e-11)*(x-10000)**3 +
    (1.3506788695652187e-07)*(x-10000)**2 &
+ (2.4321562898550723e-04)*(x-10000) +
    (1.2070000000000000e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (5.7818057971015191e-12)*(x-11000)**3 +
    (1.7741733913043481e-07)*(x-11000)**2 &
+ (5.5570085507246369e-04)*(x-11000) +
    (5.1310000000000000e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (7.5233840579708922e-12)*(x-12000)**3 +
    (1.9476275652173936e-07)*(x-12000)**2 &
+ (9.2788095072463786e-04)*(x-12000) +
    (1.2520000000000000e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (-6.0046768115940110e-12)*(x-14000)**3 +
    (2.3990306086956488e-07)*(x-14000)**2 &
+ (1.7972125855072461e-03)*(x-14000) +
    (3.9470000000000001e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnosehigh = (-6.0046768115943947e-12)*(x-16000)**3 +
    (2.0387500000000007e-07)*(x-16000)**2 &

```

```

+ (2.6847687072463775e-03)*(x-16000) +
(8.4529999999999994e+00)
END IF

qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
rndiff = 1.0-(2.300000-nose)/(2.300000-1.000000)
IF ((nose>1.0000000).AND.(nose<=2.30)) THEN
  IF ((x>=0).AND.(x<9000))THEN
    qnoselow = (1.6488888888888890e-06)*(x) +
(0.0000000000000000e+00)
  END IF
  IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (6.8925391304346964e-11)*(x-9000)**3 +
(2.3965191304347959e-07)*(x-9000)**2 &
+ (2.6282695652173449e-05)*(x-9000) + (1.4840000000000001
e-02)
  END IF
  IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (6.8925391304348127e-11)*(x-9500)**3 +
(3.4303999999999956e-07)*(x-9500)**2 &
+ (3.1762865217391326e-04)*(x-9500) + (9.650999999999999
e-02)
  END IF
  IF ((x>=10000).AND.(x<11000))THEN
    qnoselow = (2.5092173913042931e-12)*(x-10000)**3 +
(4.4642808695652170e-07)*(x-10000)**2 &

```

```

+ (7.1236269565217391e-04)*(x-10000) +
(3.4970000000000001e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnoselow = (7.2977391304346795e-12)*(x-11000)**3 +
(4.5395573913043505e-07)*(x-11000)**2 &
+ (1.6127465217391302e-03)*(x-11000) +
(1.5109999999999999e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
qnoselow = (9.5627173913041786e-12)*(x-12000)**3 +
(4.7584895652173957e-07)*(x-12000)**2 &
+ (2.5425512173913043e-03)*(x-12000) +
(3.5850000000000000e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
qnoselow = (-1.1787543478260328e-11)*(x-14000)**3 +
(5.3322526086956389e-07)*(x-14000)**2 &
+ (4.5606996521739126e-03)*(x-14000) +
(1.0650000000000000e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
qnoselow = (-1.1787543478261837e-11)*(x-16000)**3 +
(4.62500000000000245e-07)*(x-16000)**2 &
+ (6.5521501739130442e-03)*(x-16000) +
(2.1809999999999999e+01)
END IF
if ((x>=0).AND.(x<9000))THEN

```

```

qnosehigh = (3.3444444444444441e-06)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (5.5820869565218372e-11)*(x-9000)**3 +
(4.1726869565217255e-07)*(x-9000)**2 &
+ (3.9610434782609169e-05)*(x-9000) + (3.0099999999999998
e-02)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (5.5820869565216569e-11)*(x-9500)**3 +
(5.01000000000000058e-07)*(x-9500)**2 &
+ (4.9874478260869548e-04)*(x-9500) + (1.6120000000000001
e-01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (-2.2141739130434676e-11)*(x-10000)**3 +
(5.8473130434782607e-07)*(x-10000)**2 &
+ (1.0416104347826085e-03)*(x-10000) +
(5.4279999999999995e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (4.6086956522748904e-14)*(x-11000)**3 +
(5.1830608695652067e-07)*(x-11000)**2 &
+ (2.1446478260869566e-03)*(x-11000) +
(2.1469999999999998e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN

```

```

    qnosehigh = (-2.0717391304353179e-12)*(x-12000)**3 +
        (5.1844434782608807e-07)*(x-12000)**2 &
+ (3.1813982608695662e-03)*(x-12000) +
        (4.8099999999999996e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
    qnosehigh = (-1.5585652173913201e-11)*(x-14000)**3 +
        (5.0601391304347891e-07)*(x-14000)**2 &
+ (5.2303147826086947e-03)*(x-14000) +
        (1.3230000000000000e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
    qnosehigh = (-1.5585652173913201e-11)*(x-16000)**3 +
        (4.12500000000000050e-07)*(x-16000)**2 &
+ (7.0673426086956519e-03)*(x-16000) +
        (2.5590000000000000e+01)
END IF
    qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
    if ((x>=0).AND.(x<9000))THEN
    qnoselow = (4.1766666666666669e-07)*(x) +
        (0.0000000000000000e+00)
    END IF
    IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (8.7932579710142953e-12)*(x-9000)**3 +
        (1.0868811304347852e-07)*(x-9000)**2 &
+ (-5.4037101449283536e-07)*(x-9000) +
        (3.75900000000000002e-03)
    END IF

```

```

IF ((x>=9500).AND.(x<10000))THEN
  qnoselow = (8.7932579710145603e-12)*(x-9500)**3 +
    (1.2187799999999990e-07)*(x-9500)**2 &
+ (1.1474268550724641e-04)*(x-9500) + (3.1759999999999997
  e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (1.4116484057970909e-11)*(x-10000)**3 +
    (1.3506788695652187e-07)*(x-10000)**2 &
+ (2.4321562898550723e-04)*(x-10000) +
    (1.2070000000000000e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (5.7818057971015191e-12)*(x-11000)**3 +
    (1.7741733913043481e-07)*(x-11000)**2 &
+ (5.5570085507246369e-04)*(x-11000) +
    (5.1310000000000000e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (7.5233840579708922e-12)*(x-12000)**3 +
    (1.9476275652173936e-07)*(x-12000)**2 &
+ (9.2788095072463786e-04)*(x-12000) +
    (1.2520000000000000e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnoselow = (-6.0046768115940110e-12)*(x-14000)**3 +
    (2.3990306086956488e-07)*(x-14000)**2 &

```



```

+ (1.7972125855072461e-03)*(x-14000) +
(3.9470000000000001e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnoselow = (-6.0046768115943947e-12)*(x-16000)**3 +
(2.0387500000000007e-07)*(x-16000)**2 &
+ (2.6847687072463775e-03)*(x-16000) +
(8.4529999999999994e+00)
END IF
if ((x>=0).AND.(x<9000))THEN
  qnosehigh = (8.7655555555555560e-07)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (3.5136997101449081e-11)*(x-9000)**3 +
(1.2631250434782641e-07)*(x-9000)**2 &
+ (2.7161498550724527e-05)*(x-9000) + (7.8890000000000002
e-03)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (3.5136997101448971e-11)*(x-9500)**3 +
(1.79018000000000010e-07)*(x-9500)**2 &
+ (1.7982675072463774e-04)*(x-9500) + (5.7439999999999998
e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (-1.6020994202898443e-11)*(x-10000)**3 +
(2.3172349565217390e-07)*(x-10000)**2 &

```

```

+ (3.8519749855072457e-04)*(x-10000) +
  (1.9650000000000001e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (2.2357979710144903e-11)*(x-11000)**3 +
    (1.8366051304347815e-07)*(x-11000)**2 &
+ (8.0058150724637703e-04)*(x-11000) +
  (7.9740000000000000e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (-8.6134420289856410e-13)*(x-12000)**3 +
    (2.5073445217391299e-07)*(x-12000)**2 &
+ (1.2349764724637680e-03)*(x-12000) +
  (1.8040000000000000e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (-5.9277311594203114e-12)*(x-14000)**3 +
    (2.4556638695652219e-07)*(x-14000)**2 &
+ (2.2275781507246372e-03)*(x-14000) +
  (5.2699999999999996e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnosehigh = (-5.9277311594206288e-12)*(x-16000)**3 +
    (2.10000000000000011e-07)*(x-16000)**2 &
+ (3.1387109246376822e-03)*(x-16000) +
  (1.0660000000000000e+01)
END IF
qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)

```

```

      newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
      rndiff = 1.0-(5.000000-nose)/(5.000000-2.300000)
IF ((nose>2.300000).AND.(nose<=5.00)) THEN
  IF ((x>=0).AND.(x<9000))THEN
    qnoselow = (3.3444444444444441e-06)*(x) +
      (0.0000000000000000e+00)
  END IF
  IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (5.5820869565218372e-11)*(x-9000)**3 +
      (4.1726869565217255e-07)*(x-9000)**2 &
    + (3.9610434782609169e-05)*(x-9000) + (3.0099999999999998
      e-02)
  END IF
  IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (5.5820869565216569e-11)*(x-9500)**3 +
      (5.01000000000000058e-07)*(x-9500)**2 &
    + (4.9874478260869548e-04)*(x-9500) + (1.6120000000000001
      e-01)
  END IF
  IF ((x>=10000).AND.(x<11000))THEN
    qnoselow = (-2.2141739130434676e-11)*(x-10000)**3 +
      (5.8473130434782607e-07)*(x-10000)**2 &
    + (1.0416104347826085e-03)*(x-10000) +
      (5.4279999999999995e-01)
  END IF
  IF ((x>=11000).AND.(x<12000))THEN

```

```

      qnoselow = (4.6086956522748904e-14)*(x-11000)**3 +
      (5.1830608695652067e-07)*(x-11000)**2 &
+ (2.1446478260869566e-03)*(x-11000) +
      (2.1469999999999998e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
      qnoselow = (-2.0717391304353179e-12)*(x-12000)**3 +
      (5.1844434782608807e-07)*(x-12000)**2 &
+ (3.1813982608695662e-03)*(x-12000) +
      (4.8099999999999996e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
      qnoselow = (-1.5585652173913201e-11)*(x-14000)**3 +
      (5.0601391304347891e-07)*(x-14000)**2 &
+ (5.2303147826086947e-03)*(x-14000) +
      (1.3230000000000000e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
      qnoselow = (-1.5585652173913201e-11)*(x-16000)**3 +
      (4.12500000000000050e-07)*(x-16000)**2 &
+ (7.0673426086956519e-03)*(x-16000) +
      (2.5590000000000000e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
      qnosehigh = (6.2344444444444448e-06)*(x) +
      (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN

```

```

qnosehigh = (1.6877565217388714e-11)*(x-9000)**3 +
(6.4670365217391630e-07)*(x-9000)**2 &
+ (6.0808782608694712e-05)*(x-9000) + (5.6110000000000000
e-02)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnosehigh = (1.6877565217391680e-11)*(x-9500)**3 +
(6.7201999999999895e-07)*(x-9500)**2 &
+ (7.2017060869565260e-04)*(x-9500) + (2.50300000000000002
e-01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnosehigh = (-3.7685130434783418e-11)*(x-10000)**3 +
(6.9733634782608848e-07)*(x-10000)**2 &
+ (1.4048487826086953e-03)*(x-10000) +
(7.804999999999997e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnosehigh = (-5.9747043478261916e-11)*(x-11000)**3 +
(5.8428095652173970e-07)*(x-11000)**2 &
+ (2.6864660869565220e-03)*(x-11000) +
(2.84500000000000002e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
qnosehigh = (7.1533369565217201e-11)*(x-12000)**3 +
(4.0503982608695787e-07)*(x-12000)**2 &
+ (3.6757868695652157e-03)*(x-12000) +
(6.0560000000000000e+00)

```

```

END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (-9.3206673913042961e-11)*(x-14000)**3 +
    (8.3424004347825924e-07)*(x-14000)**2 &
+ (6.1543466086956536e-03)*(x-14000) +
    (1.5600000000000000e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnosehigh = (-9.3206673913044047e-11)*(x-16000)**3 +
    (2.75000000000000097e-07)*(x-16000)**2 &
+ (8.3728266956521746e-03)*(x-16000) +
    (3.0500000000000000e+01)
END IF
  qnosel = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<9000))THEN
  qnoselow = (8.7655555555555560e-07)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnoselow = (3.5136997101449081e-11)*(x-9000)**3 +
    (1.2631250434782641e-07)*(x-9000)**2 &
+ (2.7161498550724527e-05)*(x-9000) + (7.8890000000000002
    e-03)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnoselow = (3.5136997101448971e-11)*(x-9500)**3 +
    (1.7901800000000010e-07)*(x-9500)**2 &

```

```

+ (1.7982675072463774e-04)*(x-9500) + (5.743999999999998
e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (-1.6020994202898443e-11)*(x-10000)**3 +
    (2.3172349565217390e-07)*(x-10000)**2 &
+ (3.8519749855072457e-04)*(x-10000) +
    (1.96500000000000001e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (2.2357979710144903e-11)*(x-11000)**3 +
    (1.8366051304347815e-07)*(x-11000)**2 &
+ (8.0058150724637703e-04)*(x-11000) +
    (7.9740000000000000e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (-8.6134420289856410e-13)*(x-12000)**3 +
    (2.5073445217391299e-07)*(x-12000)**2 &
+ (1.2349764724637680e-03)*(x-12000) +
    (1.8040000000000000e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnoselow = (-5.9277311594203114e-12)*(x-14000)**3 +
    (2.4556638695652219e-07)*(x-14000)**2 &
+ (2.2275781507246372e-03)*(x-14000) +
    (5.2699999999999996e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN

```

```

qnoselow = (-5.9277311594206288e-12)*(x-16000)**3 +
(2.10000000000000011e-07)*(x-16000)**2 &
+ (3.1387109246376822e-03)*(x-16000) +
(1.0660000000000000e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
qnosehigh = (1.7222222222222222e-06)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnosehigh = (1.7501826086956507e-11)*(x-9000)**3 +
(2.1318726086956514e-07)*(x-9000)**2 &
+ (4.7710913043478296e-05)*(x-9000) + (1.5500000000000000
e-02)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnosehigh = (1.7501826086956507e-11)*(x-9500)**3 +
(2.39440000000000014e-07)*(x-9500)**2 &
+ (2.7402454347826081e-04)*(x-9500) + (9.483999999999994
e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnosehigh = (-4.1836521739131100e-12)*(x-10000)**3 +
(2.6569273913043479e-07)*(x-10000)**2 &
+ (5.2659091304347835e-04)*(x-10000) +
(2.9389999999999999e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN

```



```

      qnosehigh = (1.4327826086959417e-12)*(x-11000)**3 +
      (2.5314178260869541e-07)*(x-11000)**2 &
+ (1.0454254347826086e-03)*(x-11000) +
      (1.0820000000000001e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
      qnosehigh = (-7.3469021739130380e-12)*(x-12000)**3 +
      (2.5744013043478239e-07)*(x-12000)**2 &
+ (1.5560073478260873e-03)*(x-12000) +
      (2.3820000000000001e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
      qnosehigh = (9.5443804347826317e-12)*(x-14000)**3 +
      (2.1335871739130466e-07)*(x-14000)**2 &
+ (2.4976050434782605e-03)*(x-14000) +
      (6.464999999999999e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
      qnosehigh = (9.5443804347823005e-12)*(x-16000)**3 +
      (2.7062500000000006e-07)*(x-16000)**2 &
+ (3.4655724782608707e-03)*(x-16000) +
      (1.2390000000000001e+01)
END IF
      qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
      newqnr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
      rndiff = 1.0-(10.000000-nose)/(10.000000-5.000000)
      IF ((nose>5.0000000).AND.(nose<=10.00)) THEN

```

```

IF ((x>=0).AND.(x<9000))THEN
qnoselow = (6.2344444444444448e-06)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnoselow = (1.6877565217388714e-11)*(x-9000)**3 +
(6.4670365217391630e-07)*(x-9000)**2 &
+ (6.0808782608694712e-05)*(x-9000) + (5.6110000000000000
e-02)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnoselow = (1.6877565217391680e-11)*(x-9500)**3 +
(6.72019999999999895e-07)*(x-9500)**2 &
+ (7.2017060869565260e-04)*(x-9500) + (2.50300000000000002
e-01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnoselow = (-3.7685130434783418e-11)*(x-10000)**3 +
(6.9733634782608848e-07)*(x-10000)**2 &
+ (1.4048487826086953e-03)*(x-10000) +
(7.8049999999999997e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnoselow = (-5.9747043478261916e-11)*(x-11000)**3 +
(5.8428095652173970e-07)*(x-11000)**2 &
+ (2.6864660869565220e-03)*(x-11000) +
(2.84500000000000002e+00)
END IF

```

```

IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (7.1533369565217201e-11)*(x-12000)**3 +
    (4.0503982608695787e-07)*(x-12000)**2 &
  + (3.6757868695652157e-03)*(x-12000) +
    (6.0560000000000000e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnoselow = (-9.3206673913042961e-11)*(x-14000)**3 +
    (8.3424004347825924e-07)*(x-14000)**2 &
  + (6.1543466086956536e-03)*(x-14000) +
    (1.5600000000000000e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnoselow = (-9.3206673913044047e-11)*(x-16000)**3 +
    (2.75000000000000097e-07)*(x-16000)**2 &
  + (8.3728266956521746e-03)*(x-16000) +
    (3.0500000000000000e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
  qnosehigh = (1.0900000000000001e-05)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (-3.1883188405796613e-11)*(x-9000)**3 +
    (9.0362478260869516e-07)*(x-9000)**2 &
  + (9.9158405797101557e-05)*(x-9000) + (9.8100000000000007
    e-02)
END IF

```

```

IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (-3.1883188405796186e-11)*(x-9500)**3 +
    (8.5579999999999950e-07)*(x-9500)**2 &
+ (9.7887079710144926e-04)*(x-9500) + (3.6959999999999998
  e-01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (-9.7733623188405880e-11)*(x-10000)**3 +
    (8.0797521739130405e-07)*(x-10000)**2 &
+ (1.8107584057971016e-03)*(x-10000) +
    (1.0690000000000000e+00)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (1.0171768115942038e-10)*(x-11000)**3 +
    (5.1477434782608735e-07)*(x-11000)**2 &
+ (3.1335079710144921e-03)*(x-11000) +
    (3.5899999999999999e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (-6.0766123188406390e-11)*(x-12000)**3 +
    (8.1992739130434902e-07)*(x-12000)**2 &
+ (4.4682097101449280e-03)*(x-12000) +
    (7.3399999999999999e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (-2.8596775362318268e-11)*(x-14000)**3 +
    (4.5533065217391325e-07)*(x-14000)**2 &

```

```

+ (7.0187257971014473e-03)*(x-14000) +
  (1.9070000000000000e+01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnosehigh = (-2.8596775362320453e-11)*(x-16000)**3 +
    (2.83750000000000014e-07)*(x-16000)**2 &
+ (8.4968871014492811e-03)*(x-16000) +
  (3.47000000000000003e+01)
END IF
  qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<9000))THEN
  qnoselow = (1.7222222222222222e-06)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnoselow = (1.7501826086956507e-11)*(x-9000)**3 +
    (2.1318726086956514e-07)*(x-9000)**2 &
+ (4.7710913043478296e-05)*(x-9000) + (1.5500000000000000
  e-02)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnoselow = (1.7501826086956507e-11)*(x-9500)**3 +
    (2.39440000000000014e-07)*(x-9500)**2 &
+ (2.7402454347826081e-04)*(x-9500) + (9.483999999999994
  e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN

```

```

qnoselow = (-4.1836521739131100e-12)*(x-10000)**3 +
(2.6569273913043479e-07)*(x-10000)**2 &
+ (5.2659091304347835e-04)*(x-10000) +
(2.9389999999999999e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnoselow = (1.4327826086959417e-12)*(x-11000)**3 +
(2.5314178260869541e-07)*(x-11000)**2 &
+ (1.0454254347826086e-03)*(x-11000) +
(1.08200000000000001e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
qnoselow = (-7.3469021739130380e-12)*(x-12000)**3 +
(2.5744013043478239e-07)*(x-12000)**2 &
+ (1.5560073478260873e-03)*(x-12000) +
(2.38200000000000001e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
qnoselow = (9.5443804347826317e-12)*(x-14000)**3 +
(2.1335871739130466e-07)*(x-14000)**2 &
+ (2.4976050434782605e-03)*(x-14000) +
(6.4649999999999999e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
qnoselow = (9.5443804347823005e-12)*(x-16000)**3 +
(2.70625000000000006e-07)*(x-16000)**2 &
+ (3.4655724782608707e-03)*(x-16000) +
(1.23900000000000001e+01)

```

```

END IF
if ((x>=0).AND.(x<9000))THEN
qnosehigh = (3.0344444444444445e-06)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnosehigh = (1.2431043478260357e-11)*(x-9000)**3 +
(2.7677343478260955e-07)*(x-9000)**2 &
+ (8.8685521739130139e-05)*(x-9000) + (2.7310000000000001
e-02)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnosehigh = (1.2431043478260991e-11)*(x-9500)**3 +
(2.9541999999999977e-07)*(x-9500)**2 &
+ (3.7478223913043495e-04)*(x-9500) + (1.4240000000000000
e-01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnosehigh = (-9.7920869565214116e-12)*(x-10000)**3 +
(3.1406656521739089e-07)*(x-10000)**2 &
+ (6.7952552173913047e-04)*(x-10000) +
(4.0520000000000000e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnosehigh = (-3.3972695652173854e-11)*(x-11000)**3 +
(2.8469030434782601e-07)*(x-11000)**2 &
+ (1.2782823913043480e-03)*(x-11000) +
(1.3890000000000000e+00)

```

```

END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (3.7802663043478152e-11)*(x-12000)**3 +
    (1.8277221739130432e-07)*(x-12000)**2 &
+ (1.7457449130434785e-03)*(x-12000) +
    (2.9180000000000001e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (-5.2785532608695600e-11)*(x-14000)**3 +
    (4.0958819565217431e-07)*(x-14000)**2 &
+ (2.9304657391304337e-03)*(x-14000) +
    (7.4429999999999996e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnosehigh = (-5.2785532608696039e-11)*(x-16000)**3 +
    (9.2875000000000281e-08)*(x-16000)**2 &
+ (3.9353921304347839e-03)*(x-16000) +
    (1.4520000000000000e+01)
END IF
  qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
  newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if

END IF !density if
!!!!!!!!!!!!!!!!!!!!!!!!NEW ALTITUDE RANGE!!!!!!!!!!!!!!!!!!!!

IF ((rho<=0.0000665930).AND.(rho>=0.0000252390)) THEN

```



```

rhodiff = (0.0000665930 - rho) / (0.0000665930 - 0.0000252390)
rndiff = 1.0 - (0.100000 - nose) / (0.100000 - 0.049999)
IF ((nose > 0.049999) .AND. (nose <= 0.10)) THEN
  IF ((x >= 0) .AND. (x < 9000)) THEN
    qnoselow = (2.9100000000000002e-08) * (x) +
      (0.0000000000000000e+00)
  END IF
  IF ((x >= 9000) .AND. (x < 9500)) THEN
    qnoselow = (1.7183663768115799e-12) * (x - 9000)**3 +
      (1.5786250434782634e-08) * (x - 9000)**2 &
    + (-2.0065168115942117e-06) * (x - 9000) +
      (2.6190000000000002e-04)
  END IF
  IF ((x >= 9500) .AND. (x < 10000)) THEN
    qnoselow = (1.7183663768115799e-12) * (x - 9500)**3 +
      (1.8363800000000001e-08) * (x - 9500)**2 &
    + (1.5068508405797105e-05) * (x - 9500) + (3.4199999999999999
      e-03)
  END IF
  IF ((x >= 10000) .AND. (x < 11000)) THEN
    qnoselow = (4.3075672463767870e-12) * (x - 10000)**3 +
      (2.0941349565217433e-08) * (x - 10000)**2 &
    + (3.4721083188405791e-05) * (x - 10000) +
      (1.5760000000000000e-02)
  END IF
  IF ((x >= 11000) .AND. (x < 12000)) THEN
    qnoselow = (3.7946463768116607e-13) * (x - 11000)**3 +
      (3.3864051304347836e-08) * (x - 11000)**2 &

```

```

+ (8.9526484057971017e-05)*(x-11000) +
  (7.5730000000000006e-02)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (2.3380322463768221e-12)*(x-12000)**3 +
    (3.5002445217391268e-08)*(x-12000)**2 &
+ (1.5839298057971019e-04)*(x-12000) +
  (1.9950000000000001e-01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnoselow = (-4.1301064492753434e-12)*(x-14000)**3 +
    (4.9030638695652153e-08)*(x-14000)**2 &
+ (3.2645914840579712e-04)*(x-14000) +
  (6.7500000000000004e-01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnoselow = (-4.1301064492754904e-12)*(x-16000)**3 +
    (2.42500000000000128e-08)*(x-16000)**2 &
+ (4.7302042579710162e-04)*(x-16000) +
  (1.4910000000000001e+00)
END IF
if ((x>=0).AND.(x<9000))THEN
  qnosehigh = (5.3211111111111107e-08)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (4.4343953623188417e-12)*(x-9000)**3 +
    (2.1550206956521730e-08)*(x-9000)**2 &

```

```

+ (-1.3435023188405755e-06)*(x-9000) +
(4.788999999999999e-04)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (4.4343953623188611e-12)*(x-9500)**3 +
(2.8201799999999993e-08)*(x-9500)**2 &
+ (2.3532501159420286e-05)*(x-9500) + (5.748999999999998
e-03)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (5.3665092753623151e-12)*(x-10000)**3 +
(3.4853393043478258e-08)*(x-10000)**2 &
+ (5.5060097681159424e-05)*(x-10000) +
(2.5120000000000000e-02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (2.2806675362318380e-12)*(x-11000)**3 +
(5.0952920869565282e-08)*(x-11000)**2 &
+ (1.4086641159420288e-04)*(x-11000) +
(1.2039999999999999e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (4.2614742753623443e-12)*(x-12000)**3 +
(5.7794923478260829e-08)*(x-12000)**2 &
+ (2.4961425594202896e-04)*(x-12000) +
(3.1450000000000000e-01)
END IF
IF ((x>=14000).AND.(x<16000))THEN

```

```

      qnosehigh = (-7.2897948550724195e-12)*(x-14000)**3 +
      (8.3363769130434670e-08)*(x-14000)**2 &
+ (5.3193164115942040e-04)*(x-14000) +
      (1.0790000000000000e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
      qnosehigh = (-7.2897948550725811e-12)*(x-16000)**3 +
      (3.96250000000000052e-08)*(x-16000)**2 &
+ (7.7790917942029006e-04)*(x-16000) +
      (2.41800000000000001e+00)
END IF
      qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<9000))THEN
      qnoselow = (5.2533333333333334e-09)*(x) +
      (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
      qnoselow = (1.0701428405797147e-12)*(x-9000)**3 +
      (2.0841457391304290e-09)*(x-9000)**2 &
+ (-3.3568579710143049e-08)*(x-9000) +
      (4.72800000000000001e-05)
END IF
IF ((x>=9500).AND.(x<10000))THEN
      qnoselow = (1.0701428405797105e-12)*(x-9500)**3 +
      (3.68936000000000006e-09)*(x-9500)**2 &
+ (2.8531842898550718e-06)*(x-9500) + (6.85300000000000002
      e-04)
END IF

```

```

IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (5.9227431884058549e-13)*(x-10000)**3 +
    (5.2945742608695606e-09)*(x-10000)**2 &
+ (7.3451514202898555e-06)*(x-10000) +
    (3.1679999999999998e-03)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (1.4747988405796433e-13)*(x-11000)**3 +
    (7.0713972173913005e-09)*(x-11000)**2 &
+ (1.9711122898550733e-05)*(x-11000) +
    (1.64000000000000001e-02)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (1.9024231884058598e-13)*(x-12000)**3 +
    (7.5138368695652108e-09)*(x-12000)**2 &
+ (3.4296356985507227e-05)*(x-12000) +
    (4.3330000000000000e-02)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnoselow = (4.8870153623188003e-13)*(x-14000)**3 +
    (8.6552907826086968e-09)*(x-14000)**2 &
+ (6.6634612289855102e-05)*(x-14000) +
    (1.4349999999999999e-01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnoselow = (4.8870153623189083e-13)*(x-16000)**3 +
    (1.1587499999999989e-08)*(x-16000)**2 &

```

```

+ (1.0712019385507245e-04)*(x-16000) +
(3.1530000000000002e-01)
END IF
if ((x>=0).AND.(x<9000))THEN
qnosehigh = (9.6877777777777771e-09)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnosehigh = (1.2076337391304395e-12)*(x-9000)**3 +
(4.2429293913043401e-09)*(x-9000)**2 &
+ (-2.8975313043477951e-07)*(x-9000) +
(8.7189999999999997e-05)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnosehigh = (1.2076337391304430e-12)*(x-9500)**3 +
(6.05437999999999987e-09)*(x-9500)**2 &
+ (4.8589015652173900e-06)*(x-9500) + (1.1540000000000001
e-03)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnosehigh = (9.6716252173912390e-13)*(x-10000)**3 +
(7.8658306086956565e-09)*(x-10000)**2 &
+ (1.1819006869565221e-05)*(x-10000) +
(5.2480000000000001e-03)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnosehigh = (4.0052617391304714e-13)*(x-11000)**3 +
(1.0767318173913045e-08)*(x-11000)**2 &

```

```

+ (3.0452155652173905e-05)*(x-11000) +
  (2.5899999999999999e-02)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (3.6595902173912203e-13)*(x-12000)**3 +
    (1.1968896695652189e-08)*(x-12000)**2 &
+ (5.3188370521739138e-05)*(x-12000) +
  (6.7519999999999997e-02)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (5.7880819565217311e-13)*(x-14000)**3 +
    (1.4164650826086976e-08)*(x-14000)**2 &
+ (1.0545546556521736e-04)*(x-14000) +
  (2.2470000000000001e-01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnosehigh = (5.7880819565213343e-13)*(x-16000)**3 +
    (1.76375000000000068e-08)*(x-16000)**2 &
+ (1.6905976721739134e-04)*(x-16000) +
  (4.9690000000000001e-01)
END IF
  qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
  newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
rndiff = 1.0-(0.230000-nose)/(0.230000-0.100000)
IF ((nose>0.1000000).AND.(nose<=0.23)) THEN
  IF ((x>=0).AND.(x<9000))THEN

```

```

qnoselow = (5.3211111111111107e-08)*(x) +
            (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (4.4343953623188417e-12)*(x-9000)**3 +
                (2.1550206956521730e-08)*(x-9000)**2 &
                + (-1.3435023188405755e-06)*(x-9000) +
                (4.788999999999999e-04)
END IF
IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (4.4343953623188611e-12)*(x-9500)**3 +
                (2.8201799999999993e-08)*(x-9500)**2 &
                + (2.3532501159420286e-05)*(x-9500) + (5.7489999999999998
                e-03)
END IF
IF ((x>=10000).AND.(x<11000))THEN
    qnoselow = (5.3665092753623151e-12)*(x-10000)**3 +
                (3.4853393043478258e-08)*(x-10000)**2 &
                + (5.5060097681159424e-05)*(x-10000) +
                (2.5120000000000000e-02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
    qnoselow = (2.2806675362318380e-12)*(x-11000)**3 +
                (5.0952920869565282e-08)*(x-11000)**2 &
                + (1.4086641159420288e-04)*(x-11000) +
                (1.2039999999999999e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN

```



```

qnoselow = (4.2614742753623443e-12)*(x-12000)**3 +
(5.7794923478260829e-08)*(x-12000)**2 &
+ (2.4961425594202896e-04)*(x-12000) +
(3.1450000000000000e-01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
qnoselow = (-7.2897948550724195e-12)*(x-14000)**3 +
(8.3363769130434670e-08)*(x-14000)**2 &
+ (5.3193164115942040e-04)*(x-14000) +
(1.0790000000000000e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
qnoselow = (-7.2897948550725811e-12)*(x-16000)**3 +
(3.96250000000000052e-08)*(x-16000)**2 &
+ (7.7790917942029006e-04)*(x-16000) +
(2.41800000000000001e+00)
END IF
if ((x>=0).AND.(x<9000))THEN
qnosehigh = (1.1155555555555554e-07)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnosehigh = (8.5983304347825429e-12)*(x-9000)**3 +
(3.6010504347826203e-08)*(x-9000)**2 &
+ (-6.4283478260873661e-07)*(x-9000) +
(1.0039999999999999e-03)
END IF
IF ((x>=9500).AND.(x<10000))THEN

```

```

qnosehigh = (8.5983304347825300e-12)*(x-9500)**3 +
(4.89080000000000010e-08)*(x-9500)**2 &
+ (4.1816417391304375e-05)*(x-9500) + (1.0760000000000000
e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnosehigh = (6.9513391304347948e-12)*(x-10000)**3 +
(6.1805495652173930e-08)*(x-10000)**2 &
+ (9.7173165217391281e-05)*(x-10000) +
(4.49700000000000003e-02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnosehigh = (6.2023130434781771e-12)*(x-11000)**3 +
(8.2659513043478314e-08)*(x-11000)**2 &
+ (2.4163817391304352e-04)*(x-11000) +
(2.10900000000000000e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
qnosehigh = (6.1757391304348561e-12)*(x-12000)**3 +
(1.0126645217391296e-07)*(x-12000)**2 &
+ (4.2556413913043468e-04)*(x-12000) +
(5.4139999999999999e-01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
qnosehigh = (-7.9701478260870283e-12)*(x-14000)**3 +
(1.3832088695652161e-07)*(x-14000)**2 &
+ (9.0473881739130480e-04)*(x-14000) +
(1.84700000000000000e+00)

```

```

END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnosehigh = (-7.9701478260865420e-12)*(x-16000)**3 +
    (9.0499999999999546e-08)*(x-16000)**2 &
+ (1.3623805913043469e-03)*(x-16000) +
    (4.1459999999999999e+00)
END IF
  qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<9000))THEN
  qnoselow = (9.6877777777777771e-09)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnoselow = (1.2076337391304395e-12)*(x-9000)**3 +
    (4.2429293913043401e-09)*(x-9000)**2 &
+ (-2.8975313043477951e-07)*(x-9000) +
    (8.7189999999999997e-05)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnoselow = (1.2076337391304430e-12)*(x-9500)**3 +
    (6.0543799999999987e-09)*(x-9500)**2 &
+ (4.8589015652173900e-06)*(x-9500) + (1.1540000000000001
    e-03)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (9.6716252173912390e-13)*(x-10000)**3 +
    (7.8658306086956565e-09)*(x-10000)**2 &

```

```

+ (1.1819006869565221e-05)*(x-10000) +
(5.2480000000000001e-03)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnoselow = (4.0052617391304714e-13)*(x-11000)**3 +
(1.0767318173913045e-08)*(x-11000)**2 &
+ (3.0452155652173905e-05)*(x-11000) +
(2.5899999999999999e-02)
END IF
IF ((x>=12000).AND.(x<14000))THEN
qnoselow = (3.6595902173912203e-13)*(x-12000)**3 +
(1.1968896695652189e-08)*(x-12000)**2 &
+ (5.3188370521739138e-05)*(x-12000) +
(6.7519999999999997e-02)
END IF
IF ((x>=14000).AND.(x<16000))THEN
qnoselow = (5.7880819565217311e-13)*(x-14000)**3 +
(1.4164650826086976e-08)*(x-14000)**2 &
+ (1.0545546556521736e-04)*(x-14000) +
(2.2470000000000001e-01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
qnoselow = (5.7880819565213343e-13)*(x-16000)**3 +
(1.76375000000000068e-08)*(x-16000)**2 &
+ (1.6905976721739134e-04)*(x-16000) +
(4.9690000000000001e-01)
END IF
if ((x>=0).AND.(x<9000))THEN

```

```

qnosehigh = (2.0366666666666667e-08)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (2.8921675362319196e-12)*(x-9000)**3 +
(6.0023486956521253e-09)*(x-9000)**2 &
+ (2.9518376811595721e-07)*(x-9000) + (1.8330000000000001
e-04)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (2.8921675362318897e-12)*(x-9500)**3 +
(1.03406000000000006e-08)*(x-9500)**2 &
+ (8.4666581159420222e-06)*(x-9500) + (2.1930000000000001
e-03)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (8.2176492753623566e-13)*(x-10000)**3 +
(1.4678851304347823e-08)*(x-10000)**2 &
+ (2.0976383768115946e-05)*(x-10000) +
(9.3729999999999994e-03)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (1.0064727536231836e-12)*(x-11000)**3 +
(1.7144146086956523e-08)*(x-11000)**2 &
+ (5.2799381159420297e-05)*(x-11000) +
(4.58500000000000002e-02)
END IF
IF ((x>=12000).AND.(x<14000))THEN

```

```

      qnosehigh = (1.2039449275362564e-12)*(x-12000)**3 +
        (2.0163564347826031e-08)*(x-12000)**2 &
+ (9.0107091594202895e-05)*(x-12000) +
        (1.1680000000000000e-01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
      qnosehigh = (-5.8328898550728823e-13)*(x-14000)**3 +
        (2.7387233913043545e-08)*(x-14000)**2 &
+ (1.8520868811594209e-04)*(x-14000) +
        (3.8729999999999998e-01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
      qnosehigh = (-5.8328898550721381e-13)*(x-16000)**3 +
        (2.3887500000000005e-08)*(x-16000)**2 &
+ (2.8775815594202881e-04)*(x-16000) +
        (8.6260000000000003e-01)
END IF
      qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
      newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
      rndiff = 1.0-(0.500000-nose)/(0.500000-0.230000)
IF ((nose>0.230000).AND.(nose<=0.50)) THEN
      IF ((x>=0).AND.(x<9000))THEN
        qnoselow = (1.1155555555555554e-07)*(x) +
          (0.0000000000000000e+00)
      END IF
      IF ((x>=9000).AND.(x<9500))THEN

```

```

qnoselow = (8.5983304347825429e-12)*(x-9000)**3 +
(3.6010504347826203e-08)*(x-9000)**2 &
+ (-6.4283478260873661e-07)*(x-9000) +
(1.0039999999999999e-03)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnoselow = (8.5983304347825300e-12)*(x-9500)**3 +
(4.89080000000000010e-08)*(x-9500)**2 &
+ (4.1816417391304375e-05)*(x-9500) + (1.0760000000000000
e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnoselow = (6.9513391304347948e-12)*(x-10000)**3 +
(6.1805495652173930e-08)*(x-10000)**2 &
+ (9.7173165217391281e-05)*(x-10000) +
(4.49700000000000003e-02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnoselow = (6.2023130434781771e-12)*(x-11000)**3 +
(8.2659513043478314e-08)*(x-11000)**2 &
+ (2.4163817391304352e-04)*(x-11000) +
(2.10900000000000000e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
qnoselow = (6.1757391304348561e-12)*(x-12000)**3 +
(1.0126645217391296e-07)*(x-12000)**2 &
+ (4.2556413913043468e-04)*(x-12000) +
(5.4139999999999999e-01)

```

```

END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnoselow = (-7.9701478260870283e-12)*(x-14000)**3 +
    (1.3832088695652161e-07)*(x-14000)**2 &
+ (9.0473881739130480e-04)*(x-14000) +
    (1.8470000000000000e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnoselow = (-7.9701478260865420e-12)*(x-16000)**3 +
    (9.0499999999999546e-08)*(x-16000)**2 &
+ (1.3623805913043469e-03)*(x-16000) +
    (4.1459999999999999e+00)
END IF
if ((x>=0).AND.(x<9000))THEN
  qnosehigh = (2.2399999999999999e-07)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (1.0452944927536205e-11)*(x-9000)**3 +
    (6.6412582608695736e-08)*(x-9000)**2 &
+ (-2.0315275362319251e-06)*(x-9000) +
    (2.0160000000000000e-03)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (1.0452944927536205e-11)*(x-9500)**3 +
    (8.2091999999999978e-08)*(x-9500)**2 &
+ (7.2220763768115969e-05)*(x-9500) + (1.8910000000000000
    e-02)

```



```

END IF
IF ((x>=10000).AND.(x<11000))THEN
    qnosehigh = (9.5261101449275267e-12)*(x-10000)**3 +
        (9.7771417391304340e-08)*(x-10000)**2 &
+ (1.6215247246376810e-04)*(x-10000) +
        (7.6850000000000002e-02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
    qnosehigh = (8.4766144927535892e-12)*(x-11000)**3 +
        (1.2634974782608699e-07)*(x-11000)**2 &
+ (3.8627363768115936e-04)*(x-11000) +
        (3.4630000000000000e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
    qnosehigh = (7.9594601449274952e-12)*(x-12000)**3 +
        (1.5177959130434802e-07)*(x-12000)**2 &
+ (6.6440297681159409e-04)*(x-12000) +
        (8.6739999999999995e-01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
    qnosehigh = (-7.1518920289855623e-12)*(x-14000)**3 +
        (1.9953635217391302e-07)*(x-14000)**2 &
+ (1.3670348637681161e-03)*(x-14000) +
        (2.8670000000000000e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
    qnosehigh = (-7.1518920289853506e-12)*(x-16000)**3 +
        (1.5662500000000009e-07)*(x-16000)**2 &

```

```

+ (2.0793575681159414e-03)*(x-16000) +
(6.3419999999999996e+00)
END IF

qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<9000))THEN
qnoselow = (2.0366666666666667e-08)*(x) +
(0.0000000000000000e+00)
END IF

IF ((x>=9000).AND.(x<9500))THEN
qnoselow = (2.8921675362319196e-12)*(x-9000)**3 +
(6.0023486956521253e-09)*(x-9000)**2 &
+ (2.9518376811595721e-07)*(x-9000) + (1.8330000000000001
e-04)
END IF

IF ((x>=9500).AND.(x<10000))THEN
qnoselow = (2.8921675362318897e-12)*(x-9500)**3 +
(1.03406000000000006e-08)*(x-9500)**2 &
+ (8.4666581159420222e-06)*(x-9500) + (2.1930000000000001
e-03)
END IF

IF ((x>=10000).AND.(x<11000))THEN
qnoselow = (8.2176492753623566e-13)*(x-10000)**3 +
(1.4678851304347823e-08)*(x-10000)**2 &
+ (2.0976383768115946e-05)*(x-10000) +
(9.3729999999999994e-03)
END IF

IF ((x>=11000).AND.(x<12000))THEN

```

```

      qnoselow = (1.0064727536231836e-12)*(x-11000)**3 +
      (1.7144146086956523e-08)*(x-11000)**2 &
+ (5.2799381159420297e-05)*(x-11000) +
      (4.5850000000000002e-02)
END IF
IF ((x>=12000).AND.(x<14000))THEN
      qnoselow = (1.2039449275362564e-12)*(x-12000)**3 +
      (2.0163564347826031e-08)*(x-12000)**2 &
+ (9.0107091594202895e-05)*(x-12000) +
      (1.1680000000000000e-01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
      qnoselow = (-5.8328898550728823e-13)*(x-14000)**3 +
      (2.7387233913043545e-08)*(x-14000)**2 &
+ (1.8520868811594209e-04)*(x-14000) +
      (3.8729999999999998e-01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
      qnoselow = (-5.8328898550721381e-13)*(x-16000)**3 +
      (2.3887500000000005e-08)*(x-16000)**2 &
+ (2.8775815594202881e-04)*(x-16000) +
      (8.6260000000000003e-01)
END IF
if ((x>=0).AND.(x<9000))THEN
      qnosehigh = (4.1088888888888891e-08)*(x) +
      (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN

```

```

qnosehigh = (4.7687501449275232e-12)*(x-9000)**3 +
(1.0706474782608712e-08)*(x-9000)**2 &
+ (8.2497507246376275e-07)*(x-9000) + (3.697999999999999
e-04)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnosehigh = (4.7687501449275168e-12)*(x-9500)**3 +
(1.78596000000000014e-08)*(x-9500)**2 &
+ (1.5108012463768117e-05)*(x-9500) + (4.0549999999999996
e-03)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnosehigh = (4.0309971014493415e-13)*(x-10000)**3 +
(2.5012725217391293e-08)*(x-10000)**2 &
+ (3.6544175072463770e-05)*(x-10000) +
(1.66700000000000001e-02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnosehigh = (2.5790510144928322e-12)*(x-11000)**3 +
(2.6222024347825996e-08)*(x-11000)**2 &
+ (8.7778924637681158e-05)*(x-11000) +
(7.86300000000000005e-02)
END IF
IF ((x>=12000).AND.(x<14000))THEN
qnosehigh = (1.5416297101449475e-12)*(x-12000)**3 +
(3.3959177391304274e-08)*(x-12000)**2 &
+ (1.4796012637681164e-04)*(x-12000) +
(1.9520999999999999e-01)

```

```

END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (2.8392405797109046e-13)*(x-14000)**3 +
    (4.3208955652173774e-08)*(x-14000)**2 &
+ (3.0229639246376810e-04)*(x-14000) +
    (6.3929999999999998e-01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnosehigh = (2.8392405797091507e-13)*(x-16000)**3 +
    (4.4912499999999979e-08)*(x-16000)**2 &
+ (4.7853930376811629e-04)*(x-16000) +
    (1.4190000000000000e+00)
END IF
  qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
  newqnr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
rndiff = 1.0-(1.000000-nose)/(1.000000-0.500000)
IF ((nose>0.5000000).AND.(nose<=1.00)) THEN
  IF ((x>=0).AND.(x<9000))THEN
    qnoselow = (2.2399999999999999e-07)*(x) +
      (0.0000000000000000e+00)
  END IF
  IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (1.0452944927536205e-11)*(x-9000)**3 +
      (6.6412582608695736e-08)*(x-9000)**2 &
+ (-2.0315275362319251e-06)*(x-9000) +
      (2.0160000000000000e-03)
  END IF

```

```

IF ((x>=9500).AND.(x<10000))THEN
  qnoselow = (1.0452944927536205e-11)*(x-9500)**3 +
    (8.2091999999999978e-08)*(x-9500)**2 &
+ (7.2220763768115969e-05)*(x-9500) + (1.8910000000000000
    e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (9.5261101449275267e-12)*(x-10000)**3 +
    (9.7771417391304340e-08)*(x-10000)**2 &
+ (1.6215247246376810e-04)*(x-10000) +
    (7.6850000000000002e-02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (8.4766144927535892e-12)*(x-11000)**3 +
    (1.2634974782608699e-07)*(x-11000)**2 &
+ (3.8627363768115936e-04)*(x-11000) +
    (3.4630000000000000e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (7.9594601449274952e-12)*(x-12000)**3 +
    (1.5177959130434802e-07)*(x-12000)**2 &
+ (6.6440297681159409e-04)*(x-12000) +
    (8.6739999999999995e-01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnoselow = (-7.1518920289855623e-12)*(x-14000)**3 +
    (1.9953635217391302e-07)*(x-14000)**2 &

```

```

+ (1.3670348637681161e-03)*(x-14000) +
(2.8670000000000000e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnoselow = (-7.1518920289853506e-12)*(x-16000)**3 +
(1.56625000000000009e-07)*(x-16000)**2 &
+ (2.0793575681159414e-03)*(x-16000) +
(6.3419999999999996e+00)
END IF
if ((x>=0).AND.(x<9000))THEN
  qnosehigh = (4.1766666666666669e-07)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (8.7932579710142953e-12)*(x-9000)**3 +
(1.0868811304347852e-07)*(x-9000)**2 &
+ (-5.4037101449283536e-07)*(x-9000) +
(3.75900000000000002e-03)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (8.7932579710145603e-12)*(x-9500)**3 +
(1.2187799999999990e-07)*(x-9500)**2 &
+ (1.1474268550724641e-04)*(x-9500) + (3.1759999999999997
e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (1.4116484057970909e-11)*(x-10000)**3 +
(1.3506788695652187e-07)*(x-10000)**2 &

```

```

+ (2.4321562898550723e-04)*(x-10000) +
  (1.2070000000000000e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (5.7818057971015191e-12)*(x-11000)**3 +
    (1.7741733913043481e-07)*(x-11000)**2 &
+ (5.5570085507246369e-04)*(x-11000) +
  (5.1310000000000000e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (7.5233840579708922e-12)*(x-12000)**3 +
    (1.9476275652173936e-07)*(x-12000)**2 &
+ (9.2788095072463786e-04)*(x-12000) +
  (1.2520000000000000e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (-6.0046768115940110e-12)*(x-14000)**3 +
    (2.3990306086956488e-07)*(x-14000)**2 &
+ (1.7972125855072461e-03)*(x-14000) +
  (3.9470000000000001e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnosehigh = (-6.0046768115943947e-12)*(x-16000)**3 +
    (2.0387500000000007e-07)*(x-16000)**2 &
+ (2.6847687072463775e-03)*(x-16000) +
  (8.4529999999999994e+00)
END IF
qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)

```



```

if ((x>=0).AND.(x<9000))THEN
qnoselow = (4.1088888888888891e-08)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnoselow = (4.7687501449275232e-12)*(x-9000)**3 +
(1.0706474782608712e-08)*(x-9000)**2 &
+ (8.2497507246376275e-07)*(x-9000) + (3.697999999999999
e-04)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnoselow = (4.7687501449275168e-12)*(x-9500)**3 +
(1.78596000000000014e-08)*(x-9500)**2 &
+ (1.5108012463768117e-05)*(x-9500) + (4.0549999999999996
e-03)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnoselow = (4.0309971014493415e-13)*(x-10000)**3 +
(2.5012725217391293e-08)*(x-10000)**2 &
+ (3.6544175072463770e-05)*(x-10000) +
(1.66700000000000001e-02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnoselow = (2.5790510144928322e-12)*(x-11000)**3 +
(2.6222024347825996e-08)*(x-11000)**2 &
+ (8.7778924637681158e-05)*(x-11000) +
(7.86300000000000005e-02)
END IF

```

```

IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (1.5416297101449475e-12)*(x-12000)**3 +
    (3.3959177391304274e-08)*(x-12000)**2 &
  + (1.4796012637681164e-04)*(x-12000) +
    (1.9520999999999999e-01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnoselow = (2.8392405797109046e-13)*(x-14000)**3 +
    (4.3208955652173774e-08)*(x-14000)**2 &
  + (3.0229639246376810e-04)*(x-14000) +
    (6.3929999999999998e-01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnoselow = (2.8392405797091507e-13)*(x-16000)**3 +
    (4.4912499999999979e-08)*(x-16000)**2 &
  + (4.7853930376811629e-04)*(x-16000) +
    (1.4190000000000000e+00)
END IF
if ((x>=0).AND.(x<9000))THEN
  qnosehigh = (7.6911111111111107e-08)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (5.4395310144927675e-12)*(x-9000)**3 +
    (2.1185103478260839e-08)*(x-9000)**2 &
  + (6.1316550724638953e-07)*(x-9000) + (6.9220000000000002
    e-04)
END IF

```

```

IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (5.4395310144927012e-12)*(x-9500)**3 +
    (2.93444000000000049e-08)*(x-9500)**2 &
+ (2.5877917246376804e-05)*(x-9500) + (6.9750000000000003
  e-03)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (2.6433797101451613e-13)*(x-10000)**3 +
    (3.7503696521739108e-08)*(x-10000)**2 &
+ (5.9301965507246378e-05)*(x-10000) +
    (2.7930000000000000e-02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (4.3009171014492190e-12)*(x-11000)**3 +
    (3.8296710434782676e-08)*(x-11000)**2 &
+ (1.3510237246376814e-04)*(x-11000) +
    (1.2500000000000000e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (1.7506329710144985e-12)*(x-12000)**3 +
    (5.1199461739130418e-08)*(x-12000)**2 &
+ (2.2459854463768115e-04)*(x-12000) +
    (3.0270000000000002e-01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (2.0987340579708281e-13)*(x-14000)**3 +
    (6.1703259565217482e-08)*(x-14000)**2 &

```

```

+ (4.5040398724637682e-04)*(x-14000) +
    (9.7070000000000001e-01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
    qnosehigh = (2.0987340579699678e-13)*(x-16000)**3 +
        (6.2962500000000125e-08)*(x-16000)**2 &
+ (6.9973550637681173e-04)*(x-16000) +
    (2.1200000000000001e+00)
END IF
    qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
    newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
    rndiff = 1.0-(2.300000-nose)/(2.300000-1.000000)
IF ((nose>1.0000000).AND.(nose<=2.30)) THEN
    IF ((x>=0).AND.(x<9000))THEN
        qnoselow = (4.1766666666666669e-07)*(x) +
            (0.0000000000000000e+00)
    END IF
    IF ((x>=9000).AND.(x<9500))THEN
        qnoselow = (8.7932579710142953e-12)*(x-9000)**3 +
            (1.0868811304347852e-07)*(x-9000)**2 &
+ (-5.4037101449283536e-07)*(x-9000) +
            (3.7590000000000002e-03)
    END IF
    IF ((x>=9500).AND.(x<10000))THEN
        qnoselow = (8.7932579710145603e-12)*(x-9500)**3 +
            (1.2187799999999990e-07)*(x-9500)**2 &

```

```

+ (1.1474268550724641e-04)*(x-9500) + (3.175999999999997
e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (1.4116484057970909e-11)*(x-10000)**3 +
    (1.3506788695652187e-07)*(x-10000)**2 &
+ (2.4321562898550723e-04)*(x-10000) +
    (1.2070000000000000e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (5.7818057971015191e-12)*(x-11000)**3 +
    (1.7741733913043481e-07)*(x-11000)**2 &
+ (5.5570085507246369e-04)*(x-11000) +
    (5.1310000000000000e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (7.5233840579708922e-12)*(x-12000)**3 +
    (1.9476275652173936e-07)*(x-12000)**2 &
+ (9.2788095072463786e-04)*(x-12000) +
    (1.2520000000000000e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnoselow = (-6.0046768115940110e-12)*(x-14000)**3 +
    (2.3990306086956488e-07)*(x-14000)**2 &
+ (1.7972125855072461e-03)*(x-14000) +
    (3.9470000000000001e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN

```

```

qnoselow = (-6.0046768115943947e-12)*(x-16000)**3 +
(2.0387500000000007e-07)*(x-16000)**2 &
+ (2.6847687072463775e-03)*(x-16000) +
(8.4529999999999994e+00)
END IF
if ((x>=0).AND.(x<9000))THEN
qnosehigh = (8.7655555555555560e-07)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnosehigh = (3.5136997101449081e-11)*(x-9000)**3 +
(1.2631250434782641e-07)*(x-9000)**2 &
+ (2.7161498550724527e-05)*(x-9000) + (7.8890000000000002
e-03)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnosehigh = (3.5136997101448971e-11)*(x-9500)**3 +
(1.79018000000000010e-07)*(x-9500)**2 &
+ (1.7982675072463774e-04)*(x-9500) + (5.7439999999999998
e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnosehigh = (-1.6020994202898443e-11)*(x-10000)**3 +
(2.3172349565217390e-07)*(x-10000)**2 &
+ (3.8519749855072457e-04)*(x-10000) +
(1.96500000000000001e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN

```

```

    qnosehigh = (2.2357979710144903e-11)*(x-11000)**3 +
        (1.8366051304347815e-07)*(x-11000)**2 &
+ (8.0058150724637703e-04)*(x-11000) +
        (7.9740000000000000e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
    qnosehigh = (-8.6134420289856410e-13)*(x-12000)**3 +
        (2.5073445217391299e-07)*(x-12000)**2 &
+ (1.2349764724637680e-03)*(x-12000) +
        (1.8040000000000000e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
    qnosehigh = (-5.9277311594203114e-12)*(x-14000)**3 +
        (2.4556638695652219e-07)*(x-14000)**2 &
+ (2.2275781507246372e-03)*(x-14000) +
        (5.2699999999999996e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
    qnosehigh = (-5.9277311594206288e-12)*(x-16000)**3 +
        (2.10000000000000011e-07)*(x-16000)**2 &
+ (3.1387109246376822e-03)*(x-16000) +
        (1.0660000000000000e+01)
END IF
    qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<9000))THEN
    qnoselow = (7.6911111111111107e-08)*(x) +
        (0.0000000000000000e+00)
END IF

```

```

IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (5.4395310144927675e-12)*(x-9000)**3 +
        (2.1185103478260839e-08)*(x-9000)**2 &
    + (6.1316550724638953e-07)*(x-9000) + (6.9220000000000002
        e-04)
END IF

IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (5.4395310144927012e-12)*(x-9500)**3 +
        (2.93444000000000049e-08)*(x-9500)**2 &
    + (2.5877917246376804e-05)*(x-9500) + (6.9750000000000003
        e-03)
END IF

IF ((x>=10000).AND.(x<11000))THEN
    qnoselow = (2.6433797101451613e-13)*(x-10000)**3 +
        (3.7503696521739108e-08)*(x-10000)**2 &
    + (5.9301965507246378e-05)*(x-10000) +
        (2.7930000000000000e-02)
END IF

IF ((x>=11000).AND.(x<12000))THEN
    qnoselow = (4.3009171014492190e-12)*(x-11000)**3 +
        (3.8296710434782676e-08)*(x-11000)**2 &
    + (1.3510237246376814e-04)*(x-11000) +
        (1.2500000000000000e-01)
END IF

IF ((x>=12000).AND.(x<14000))THEN
    qnoselow = (1.7506329710144985e-12)*(x-12000)**3 +
        (5.1199461739130418e-08)*(x-12000)**2 &

```



```

+ (2.2459854463768115e-04)*(x-12000) +
(3.0270000000000002e-01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
qnoselow = (2.0987340579708281e-13)*(x-14000)**3 +
(6.1703259565217482e-08)*(x-14000)**2 &
+ (4.5040398724637682e-04)*(x-14000) +
(9.7070000000000001e-01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
qnoselow = (2.0987340579699678e-13)*(x-16000)**3 +
(6.2962500000000125e-08)*(x-16000)**2 &
+ (6.9973550637681173e-04)*(x-16000) +
(2.1200000000000001e+00)
END IF
if ((x>=0).AND.(x<9000))THEN
qnosehigh = (1.6400000000000001e-07)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnosehigh = (1.0394666666666349e-12)*(x-9000)**3 +
(4.9872800000000022e-08)*(x-9000)**2 &
+ (-1.8082666666666690e-06)*(x-9000) +
(1.4760000000000001e-03)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnosehigh = (1.0394666666667274e-12)*(x-9500)**3 +
(5.1431999999999961e-08)*(x-9500)**2 &

```

```

+ (4.8844133333333329e-05)*(x-9500) + (1.3169999999999999
e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (2.17306666666666879e-12)*(x-10000)**3 +
    (5.2991199999999992e-08)*(x-10000)**2 &
+ (1.0105573333333334e-04)*(x-10000) +
    (5.0580000000000000e-02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (1.83226666666666100e-12)*(x-11000)**3 +
    (5.95104000000000017e-08)*(x-11000)**2 &
+ (2.1355733333333339e-04)*(x-11000) +
    (2.06800000000000001e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (7.51516666666666556e-12)*(x-12000)**3 +
    (6.50072000000000032e-08)*(x-12000)**2 &
+ (3.3807493333333326e-04)*(x-12000) +
    (4.81700000000000002e-01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (-6.24553333333333013e-12)*(x-14000)**3 +
    (1.10098200000000001e-07)*(x-14000)**2 &
+ (6.8828573333333325e-04)*(x-14000) +
    (1.47800000000000000e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN

```

```

      qnosehigh = (-6.245533333333334370e-12)*(x-16000)**3 +
      (7.26250000000000003e-08)*(x-16000)**2 &
+ (1.05373213333333337e-03)*(x-16000) +
      (3.24500000000000001e+00)
END IF

      qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
      newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if

      rndiff = 1.0-(5.000000-nose)/(5.000000-2.300000)
IF ((nose>2.3000000).AND.(nose<=5.00)) THEN
      IF ((x>=0).AND.(x<9000))THEN
      qnoselow = (8.76555555555555560e-07)*(x) +
      (0.00000000000000000e+00)
END IF
      IF ((x>=9000).AND.(x<9500))THEN
      qnoselow = (3.5136997101449081e-11)*(x-9000)**3 +
      (1.2631250434782641e-07)*(x-9000)**2 &
+ (2.7161498550724527e-05)*(x-9000) + (7.88900000000000002
      e-03)
END IF
      IF ((x>=9500).AND.(x<10000))THEN
      qnoselow = (3.5136997101448971e-11)*(x-9500)**3 +
      (1.79018000000000010e-07)*(x-9500)**2 &
+ (1.7982675072463774e-04)*(x-9500) + (5.7439999999999998
      e-02)
END IF
      IF ((x>=10000).AND.(x<11000))THEN

```

```

qnoselow = (-1.6020994202898443e-11)*(x-10000)**3 +
(2.3172349565217390e-07)*(x-10000)**2 &
+ (3.8519749855072457e-04)*(x-10000) +
(1.96500000000000001e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnoselow = (2.2357979710144903e-11)*(x-11000)**3 +
(1.8366051304347815e-07)*(x-11000)**2 &
+ (8.0058150724637703e-04)*(x-11000) +
(7.9740000000000000e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
qnoselow = (-8.6134420289856410e-13)*(x-12000)**3 +
(2.5073445217391299e-07)*(x-12000)**2 &
+ (1.2349764724637680e-03)*(x-12000) +
(1.8040000000000000e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
qnoselow = (-5.9277311594203114e-12)*(x-14000)**3 +
(2.4556638695652219e-07)*(x-14000)**2 &
+ (2.2275781507246372e-03)*(x-14000) +
(5.2699999999999996e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
qnoselow = (-5.9277311594206288e-12)*(x-16000)**3 +
(2.10000000000000011e-07)*(x-16000)**2 &
+ (3.1387109246376822e-03)*(x-16000) +
(1.0660000000000000e+01)

```

```

END IF
if ((x>=0).AND.(x<9000))THEN
qnosehigh = (1.7222222222222222e-06)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnosehigh = (1.7501826086956507e-11)*(x-9000)**3 +
(2.1318726086956514e-07)*(x-9000)**2 &
+ (4.7710913043478296e-05)*(x-9000) + (1.5500000000000000
e-02)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnosehigh = (1.7501826086956507e-11)*(x-9500)**3 +
(2.39440000000000014e-07)*(x-9500)**2 &
+ (2.7402454347826081e-04)*(x-9500) + (9.4839999999999994
e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnosehigh = (-4.1836521739131100e-12)*(x-10000)**3 +
(2.6569273913043479e-07)*(x-10000)**2 &
+ (5.2659091304347835e-04)*(x-10000) +
(2.9389999999999999e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnosehigh = (1.4327826086959417e-12)*(x-11000)**3 +
(2.5314178260869541e-07)*(x-11000)**2 &
+ (1.0454254347826086e-03)*(x-11000) +
(1.0820000000000001e+00)

```

```

END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (-7.3469021739130380e-12)*(x-12000)**3 +
    (2.5744013043478239e-07)*(x-12000)**2 &
+ (1.5560073478260873e-03)*(x-12000) +
    (2.3820000000000001e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (9.5443804347826317e-12)*(x-14000)**3 +
    (2.1335871739130466e-07)*(x-14000)**2 &
+ (2.4976050434782605e-03)*(x-14000) +
    (6.4649999999999999e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnosehigh = (9.5443804347823005e-12)*(x-16000)**3 +
    (2.7062500000000006e-07)*(x-16000)**2 &
+ (3.4655724782608707e-03)*(x-16000) +
    (1.2390000000000001e+01)
END IF
  qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
  if ((x>=0).AND.(x<9000))THEN
    qnoselow = (1.6400000000000001e-07)*(x) +
      (0.0000000000000000e+00)
  END IF
  IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (1.03946666666666349e-12)*(x-9000)**3 +
      (4.9872800000000022e-08)*(x-9000)**2 &

```

```

+ (-1.8082666666666690e-06)*(x-9000) +
    (1.4760000000000001e-03)
END IF
IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (1.03946666666667274e-12)*(x-9500)**3 +
        (5.1431999999999961e-08)*(x-9500)**2 &
+ (4.8844133333333329e-05)*(x-9500) + (1.3169999999999999
    e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
    qnoselow = (2.17306666666666879e-12)*(x-10000)**3 +
        (5.2991199999999992e-08)*(x-10000)**2 &
+ (1.0105573333333334e-04)*(x-10000) +
        (5.0580000000000000e-02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
    qnoselow = (1.83226666666666100e-12)*(x-11000)**3 +
        (5.95104000000000017e-08)*(x-11000)**2 &
+ (2.1355733333333339e-04)*(x-11000) +
        (2.06800000000000001e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
    qnoselow = (7.51516666666666556e-12)*(x-12000)**3 +
        (6.50072000000000032e-08)*(x-12000)**2 &
+ (3.3807493333333326e-04)*(x-12000) +
        (4.81700000000000002e-01)
END IF
IF ((x>=14000).AND.(x<16000))THEN

```

```

      qnoselow =  (-6.2455333333333013e-12)*(x-14000)**3 +
      (1.10098200000000001e-07)*(x-14000)**2 &
+ (6.8828573333333325e-04)*(x-14000) +
      (1.4780000000000000e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
      qnoselow =  (-6.24553333333334370e-12)*(x-16000)**3 +
      (7.26250000000000003e-08)*(x-16000)**2 &
+ (1.0537321333333337e-03)*(x-16000) +
      (3.24500000000000001e+00)
END IF
if ((x>=0).AND.(x<9000))THEN
      qnosehigh = (3.3144444444444442e-07)*(x) +
      (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
      qnosehigh =  (-8.2515420289854076e-12)*(x-9000)**3 +
      (9.1103313043478111e-08)*(x-9000)**2 &
+ (-2.3347710144926981e-06)*(x-9000) +
      (2.9830000000000000e-03)
END IF
IF ((x>=9500).AND.(x<10000))THEN
      qnosehigh =  (-8.2515420289854076e-12)*(x-9500)**3 +
      (7.87260000000000013e-08)*(x-9500)**2 &
+ (8.2579885507246356e-05)*(x-9500) + (2.35600000000000001
      e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN

```



```

qnosehigh = (4.6340840579710548e-12)*(x-10000)**3 +
(6.6348686956521597e-08)*(x-10000)**2 &
+ (1.5511722898550731e-04)*(x-10000) +
(8.3500000000000005e-02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnosehigh = (2.2322057971013796e-12)*(x-11000)**3 +
(8.0250939130434907e-08)*(x-11000)**2 &
+ (3.0171685507246368e-04)*(x-11000) +
(3.0959999999999999e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
qnosehigh = (3.6973840579710279e-12)*(x-12000)**3 +
(8.6947556521739099e-08)*(x-12000)**2 &
+ (4.6891535072463762e-04)*(x-12000) +
(6.9379999999999997e-01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
qnosehigh = (-3.3344768115941887e-12)*(x-14000)**3 +
(1.0913186086956525e-07)*(x-14000)**2 &
+ (8.6107418550724637e-04)*(x-14000) +
(2.0089999999999999e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
qnosehigh = (-3.3344768115942945e-12)*(x-16000)**3 +
(8.9125000000000012e-08)*(x-16000)**2 &
+ (1.2575879072463771e-03)*(x-16000) +
(4.1410000000000000e+00)

```

```

END IF

  qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
  newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
rndiff = 1.0-(10.000000-nose)/(10.000000-5.000000)
IF ((nose>5.0000000).AND.(nose<=10.00)) THEN
  IF ((x>=0).AND.(x<9000))THEN
    qnoselow = (1.7222222222222222e-06)*(x) +
      (0.0000000000000000e+00)
  END IF
  IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (1.7501826086956507e-11)*(x-9000)**3 +
      (2.1318726086956514e-07)*(x-9000)**2 &
      + (4.7710913043478296e-05)*(x-9000) + (1.5500000000000000
        e-02)
  END IF
  IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (1.7501826086956507e-11)*(x-9500)**3 +
      (2.39440000000000014e-07)*(x-9500)**2 &
      + (2.7402454347826081e-04)*(x-9500) + (9.4839999999999994
        e-02)
  END IF
  IF ((x>=10000).AND.(x<11000))THEN
    qnoselow = (-4.1836521739131100e-12)*(x-10000)**3 +
      (2.6569273913043479e-07)*(x-10000)**2 &
      + (5.2659091304347835e-04)*(x-10000) +
      (2.9389999999999999e-01)
  END IF

```

```

IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (1.4327826086959417e-12)*(x-11000)**3 +
    (2.5314178260869541e-07)*(x-11000)**2 &
  + (1.0454254347826086e-03)*(x-11000) +
    (1.0820000000000001e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (-7.3469021739130380e-12)*(x-12000)**3 +
    (2.5744013043478239e-07)*(x-12000)**2 &
  + (1.5560073478260873e-03)*(x-12000) +
    (2.3820000000000001e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnoselow = (9.5443804347826317e-12)*(x-14000)**3 +
    (2.1335871739130466e-07)*(x-14000)**2 &
  + (2.4976050434782605e-03)*(x-14000) +
    (6.4649999999999999e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnoselow = (9.5443804347823005e-12)*(x-16000)**3 +
    (2.7062500000000006e-07)*(x-16000)**2 &
  + (3.4655724782608707e-03)*(x-16000) +
    (1.2390000000000001e+01)
END IF
if ((x>=0).AND.(x<9000))THEN
  qnosehigh = (3.0344444444444445e-06)*(x) +
    (0.0000000000000000e+00)
END IF

```

```

IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (1.2431043478260357e-11)*(x-9000)**3 +
    (2.7677343478260955e-07)*(x-9000)**2 &
+ (8.8685521739130139e-05)*(x-9000) + (2.7310000000000001
  e-02)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (1.2431043478260991e-11)*(x-9500)**3 +
    (2.9541999999999977e-07)*(x-9500)**2 &
+ (3.7478223913043495e-04)*(x-9500) + (1.4240000000000000
  e-01)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (-9.7920869565214116e-12)*(x-10000)**3 +
    (3.1406656521739089e-07)*(x-10000)**2 &
+ (6.7952552173913047e-04)*(x-10000) +
    (4.0520000000000000e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (-3.3972695652173854e-11)*(x-11000)**3 +
    (2.8469030434782601e-07)*(x-11000)**2 &
+ (1.2782823913043480e-03)*(x-11000) +
    (1.3890000000000000e+00)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (3.7802663043478152e-11)*(x-12000)**3 +
    (1.8277221739130432e-07)*(x-12000)**2 &

```

```

+ (1.7457449130434785e-03)*(x-12000) +
(2.9180000000000001e+00)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (-5.2785532608695600e-11)*(x-14000)**3 +
(4.0958819565217431e-07)*(x-14000)**2 &
+ (2.9304657391304337e-03)*(x-14000) +
(7.4429999999999996e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnosehigh = (-5.2785532608696039e-11)*(x-16000)**3 +
(9.2875000000000281e-08)*(x-16000)**2 &
+ (3.9353921304347839e-03)*(x-16000) +
(1.4520000000000000e+01)
END IF
  qnosel = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<9000))THEN
  qnoselow = (3.3144444444444442e-07)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnoselow = (-8.2515420289854076e-12)*(x-9000)**3 +
(9.1103313043478111e-08)*(x-9000)**2 &
+ (-2.3347710144926981e-06)*(x-9000) +
(2.9830000000000000e-03)
END IF
IF ((x>=9500).AND.(x<10000))THEN

```

```

qnoselow = (-8.2515420289854076e-12)*(x-9500)**3 +
(7.87260000000000013e-08)*(x-9500)**2 &
+ (8.2579885507246356e-05)*(x-9500) + (2.3560000000000001
e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnoselow = (4.6340840579710548e-12)*(x-10000)**3 +
(6.6348686956521597e-08)*(x-10000)**2 &
+ (1.5511722898550731e-04)*(x-10000) +
(8.3500000000000005e-02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnoselow = (2.2322057971013796e-12)*(x-11000)**3 +
(8.0250939130434907e-08)*(x-11000)**2 &
+ (3.0171685507246368e-04)*(x-11000) +
(3.0959999999999999e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
qnoselow = (3.6973840579710279e-12)*(x-12000)**3 +
(8.6947556521739099e-08)*(x-12000)**2 &
+ (4.6891535072463762e-04)*(x-12000) +
(6.9379999999999997e-01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
qnoselow = (-3.3344768115941887e-12)*(x-14000)**3 +
(1.0913186086956525e-07)*(x-14000)**2 &
+ (8.6107418550724637e-04)*(x-14000) +
(2.0089999999999999e+00)

```

```

END IF
IF ((x>=16000).AND.(x<18000))THEN
    qnoselow = (-3.3344768115942945e-12)*(x-16000)**3 +
        (8.91250000000000012e-08)*(x-16000)**2 &
+ (1.2575879072463771e-03)*(x-16000) +
        (4.1410000000000000e+00)
END IF
if ((x>=0).AND.(x<9000))THEN
    qnosehigh = (6.2555555555555548e-07)*(x) +
        (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
    qnosehigh = (-1.8864840579710335e-11)*(x-9000)**3 +
        (1.3415726086956550e-07)*(x-9000)**2 &
+ (4.1775797101448188e-06)*(x-9000) + (5.6299999999999996
        e-03)
END IF
IF ((x>=9500).AND.(x<10000))THEN
    qnosehigh = (-1.8864840579709963e-11)*(x-9500)**3 +
        (1.0585999999999984e-07)*(x-9500)**2 &
+ (1.2418621014492756e-04)*(x-9500) + (3.8899999999999997
        e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
    qnosehigh = (6.4396811594203005e-12)*(x-10000)**3 +
        (7.7562739130434799e-08)*(x-10000)**2 &
+ (2.1589757971014493e-04)*(x-10000) +
        (1.2509999999999999e-01)

```

```

END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (1.9761159420287617e-12)*(x-11000)**3 +
    (9.6881782608695886e-08)*(x-11000)**2 &
+ (3.9034210144927542e-04)*(x-11000) +
    (4.249999999999999e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (-1.5635688405796161e-12)*(x-12000)**3 +
    (1.0281013043478244e-07)*(x-12000)**2 &
+ (5.9003401449275348e-04)*(x-12000) +
    (9.1420000000000001e-01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (1.0327137681158848e-12)*(x-14000)**3 +
    (9.3428717391304355e-08)*(x-14000)**2 &
+ (9.8251171014492782e-04)*(x-14000) +
    (2.4929999999999999e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnosehigh = (1.0327137681159973e-12)*(x-16000)**3 +
    (9.9625000000000100e-08)*(x-16000)**2 &
+ (1.3686191449275359e-03)*(x-16000) +
    (4.8399999999999999e+00)
END IF
  qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
  newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if

```



```

END IF !density if
!!!!!!!!!!!!!!!!!!!!NEW ALTITUDE RANGE!!!!!!!!!!!!!!!!!!!!

IF ((rho <= 0.0000252390) .AND. (rho >= 0.0000096940)) THEN
  rhodiff = (0.0000252390 - rho) / (0.0000252390 - 0.0000096940)
  rndiff = 1.0 - (0.100000 - nose) / (0.100000 - 0.049999)
  IF ((nose > 0.049999) .AND. (nose <= 0.10)) THEN
    IF ((x >= 0) .AND. (x < 9000)) THEN
      qnoselow = (5.253333333333334e-09) * (x) +
        (0.0000000000000000e+00)
    END IF
    IF ((x >= 9000) .AND. (x < 9500)) THEN
      qnoselow = (1.0701428405797147e-12) * (x - 9000)**3 +
        (2.0841457391304290e-09) * (x - 9000)**2 &
        + (-3.3568579710143049e-08) * (x - 9000) +
        (4.7280000000000001e-05)
    END IF
    IF ((x >= 9500) .AND. (x < 10000)) THEN
      qnoselow = (1.0701428405797105e-12) * (x - 9500)**3 +
        (3.6893600000000006e-09) * (x - 9500)**2 &
        + (2.8531842898550718e-06) * (x - 9500) + (6.8530000000000002
          e-04)
    END IF
    IF ((x >= 10000) .AND. (x < 11000)) THEN
      qnoselow = (5.9227431884058549e-13) * (x - 10000)**3 +
        (5.2945742608695606e-09) * (x - 10000)**2 &

```

```

+ (7.3451514202898555e-06)*(x-10000) +
(3.1679999999999998e-03)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnoselow = (1.4747988405796433e-13)*(x-11000)**3 +
(7.0713972173913005e-09)*(x-11000)**2 &
+ (1.9711122898550733e-05)*(x-11000) +
(1.64000000000000001e-02)
END IF
IF ((x>=12000).AND.(x<14000))THEN
qnoselow = (1.9024231884058598e-13)*(x-12000)**3 +
(7.5138368695652108e-09)*(x-12000)**2 &
+ (3.4296356985507227e-05)*(x-12000) +
(4.3330000000000000e-02)
END IF
IF ((x>=14000).AND.(x<16000))THEN
qnoselow = (4.8870153623188003e-13)*(x-14000)**3 +
(8.6552907826086968e-09)*(x-14000)**2 &
+ (6.6634612289855102e-05)*(x-14000) +
(1.4349999999999999e-01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
qnoselow = (4.8870153623189083e-13)*(x-16000)**3 +
(1.1587499999999989e-08)*(x-16000)**2 &
+ (1.0712019385507245e-04)*(x-16000) +
(3.15300000000000002e-01)
END IF
if ((x>=0).AND.(x<9000))THEN

```

```

qnosehigh = (9.6877777777777771e-09)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (1.2076337391304395e-12)*(x-9000)**3 +
(4.2429293913043401e-09)*(x-9000)**2 &
+ (-2.8975313043477951e-07)*(x-9000) +
(8.718999999999997e-05)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (1.2076337391304430e-12)*(x-9500)**3 +
(6.0543799999999987e-09)*(x-9500)**2 &
+ (4.8589015652173900e-06)*(x-9500) + (1.1540000000000001
e-03)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (9.6716252173912390e-13)*(x-10000)**3 +
(7.8658306086956565e-09)*(x-10000)**2 &
+ (1.1819006869565221e-05)*(x-10000) +
(5.2480000000000001e-03)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (4.0052617391304714e-13)*(x-11000)**3 +
(1.0767318173913045e-08)*(x-11000)**2 &
+ (3.0452155652173905e-05)*(x-11000) +
(2.5899999999999999e-02)
END IF
IF ((x>=12000).AND.(x<14000))THEN

```

```

      qnosehigh = (3.6595902173912203e-13)*(x-12000)**3 +
        (1.1968896695652189e-08)*(x-12000)**2 &
+ (5.3188370521739138e-05)*(x-12000) +
        (6.751999999999997e-02)
END IF
IF ((x>=14000).AND.(x<16000))THEN
      qnosehigh = (5.7880819565217311e-13)*(x-14000)**3 +
        (1.4164650826086976e-08)*(x-14000)**2 &
+ (1.0545546556521736e-04)*(x-14000) +
        (2.2470000000000001e-01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
      qnosehigh = (5.7880819565213343e-13)*(x-16000)**3 +
        (1.7637500000000068e-08)*(x-16000)**2 &
+ (1.6905976721739134e-04)*(x-16000) +
        (4.9690000000000001e-01)
END IF
      qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
      if ((x>=0).AND.(x<9000))THEN
      qnoselow = (1.0316666666666667e-09)*(x) +
        (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
      qnoselow = (2.2141118840579952e-13)*(x-9000)**3 +
        (3.4565321739130097e-10)*(x-9000)**2 &
+ (3.7850594202899638e-08)*(x-9000) + (9.2850000000000000
        e-06)
END IF

```

```

IF ((x>=9500).AND.(x<10000))THEN
  qnoselow = (2.2141118840579702e-13)*(x-9500)**3 +
    (6.77770000000000096e-10)*(x-9500)**2 &
+ (5.4956220289855023e-07)*(x-9500) + (1.4229999999999999
  e-04)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (1.6352262318840465e-13)*(x-10000)**3 +
    (1.0098867826086961e-09)*(x-10000)**2 &
+ (1.3933905942028990e-06)*(x-10000) +
    (6.1419999999999997e-04)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (3.4813318840580546e-14)*(x-11000)**3 +
    (1.5004546521739133e-09)*(x-11000)**2 &
+ (3.9037320289855052e-06)*(x-11000) +
    (3.1809999999999998e-03)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (3.4403237318840585e-13)*(x-12000)**3 +
    (1.6048946086956515e-09)*(x-12000)**2 &
+ (7.0090812898550736e-06)*(x-12000) +
    (8.6199999999999992e-03)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnoselow = (-5.6630647463768148e-13)*(x-14000)**3 +
    (3.6690888478260889e-09)*(x-14000)**2 &

```

```

+ (1.7557048202898549e-05)*(x-14000) +
(3.1809999999999998e-02)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnoselow = (-5.6630647463768148e-13)*(x-16000)**3 +
(2.71249999999999842e-10)*(x-16000)**2 &
+ (2.5437725898550728e-05)*(x-16000) +
(7.7070000000000000e-02)
END IF
if ((x>=0).AND.(x<9000))THEN
  qnosehigh = (1.9433333333333334e-09)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (2.8813721739130191e-13)*(x-9000)**3 +
(6.8997417391304635e-10)*(x-9000)**2 &
+ (3.6398608695651379e-08)*(x-9000) + (1.7490000000000000
e-05)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (2.8813721739130504e-13)*(x-9500)**3 +
(1.1221799999999991e-09)*(x-9500)**2 &
+ (9.4247569565217419e-07)*(x-9500) + (2.4420000000000003
e-04)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (2.5285556521739101e-13)*(x-10000)**3 +
(1.5543858260869575e-09)*(x-10000)**2 &

```

```

+ (2.2807586086956521e-06)*(x-10000) +
  (1.0319999999999999e-03)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (2.6895052173913178e-13)*(x-11000)**3 +
    (2.3129525217391257e-09)*(x-11000)**2 &
+ (6.1480969565217401e-06)*(x-11000) +
  (5.1200000000000004e-03)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (1.3488456521738970e-13)*(x-12000)**3 +
    (3.1198040869565256e-09)*(x-12000)**2 &
+ (1.1580853565217387e-05)*(x-12000) +
  (1.3849999999999999e-02)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (-1.5547691304347649e-13)*(x-14000)**3 +
    (3.9291114782608696e-09)*(x-14000)**2 &
+ (2.5678684695652166e-05)*(x-14000) +
  (5.0569999999999997e-02)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnosehigh = (-1.5547691304348664e-13)*(x-16000)**3 +
    (2.9962500000000069e-09)*(x-16000)**2 &
+ (3.9529407652173926e-05)*(x-16000) +
  (1.1640000000000000e-01)
END IF
qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)

```

```

    newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
rndiff = 1.0-(0.230000-nose)/(0.230000-0.100000)
IF ((nose>0.1000000).AND.(nose<=0.23)) THEN
    IF ((x>=0).AND.(x<9000))THEN
        qnoselow = (9.6877777777777771e-09)*(x) +
            (0.0000000000000000e+00)
    END IF
    IF ((x>=9000).AND.(x<9500))THEN
        qnoselow = (1.2076337391304395e-12)*(x-9000)**3 +
            (4.2429293913043401e-09)*(x-9000)**2 &
            + (-2.8975313043477951e-07)*(x-9000) +
            (8.7189999999999997e-05)
    END IF
    IF ((x>=9500).AND.(x<10000))THEN
        qnoselow = (1.2076337391304430e-12)*(x-9500)**3 +
            (6.05437999999999987e-09)*(x-9500)**2 &
            + (4.8589015652173900e-06)*(x-9500) + (1.15400000000000001
                e-03)
    END IF
    IF ((x>=10000).AND.(x<11000))THEN
        qnoselow = (9.6716252173912390e-13)*(x-10000)**3 +
            (7.8658306086956565e-09)*(x-10000)**2 &
            + (1.1819006869565221e-05)*(x-10000) +
            (5.24800000000000001e-03)
    END IF
    IF ((x>=11000).AND.(x<12000))THEN

```



```

      qnoselow = (4.0052617391304714e-13)*(x-11000)**3 +
      (1.0767318173913045e-08)*(x-11000)**2 &
+ (3.0452155652173905e-05)*(x-11000) +
      (2.5899999999999999e-02)
END IF
IF ((x>=12000).AND.(x<14000))THEN
      qnoselow = (3.6595902173912203e-13)*(x-12000)**3 +
      (1.1968896695652189e-08)*(x-12000)**2 &
+ (5.3188370521739138e-05)*(x-12000) +
      (6.7519999999999997e-02)
END IF
IF ((x>=14000).AND.(x<16000))THEN
      qnoselow = (5.7880819565217311e-13)*(x-14000)**3 +
      (1.4164650826086976e-08)*(x-14000)**2 &
+ (1.0545546556521736e-04)*(x-14000) +
      (2.24700000000000001e-01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
      qnoselow = (5.7880819565213343e-13)*(x-16000)**3 +
      (1.76375000000000068e-08)*(x-16000)**2 &
+ (1.6905976721739134e-04)*(x-16000) +
      (4.96900000000000001e-01)
END IF
if ((x>=0).AND.(x<9000))THEN
      qnosehigh = (2.0366666666666667e-08)*(x) +
      (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN

```

```

qnosehigh = (2.8921675362319196e-12)*(x-9000)**3 +
(6.0023486956521253e-09)*(x-9000)**2 &
+ (2.9518376811595721e-07)*(x-9000) + (1.8330000000000001
e-04)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnosehigh = (2.8921675362318897e-12)*(x-9500)**3 +
(1.03406000000000006e-08)*(x-9500)**2 &
+ (8.4666581159420222e-06)*(x-9500) + (2.1930000000000001
e-03)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnosehigh = (8.2176492753623566e-13)*(x-10000)**3 +
(1.4678851304347823e-08)*(x-10000)**2 &
+ (2.0976383768115946e-05)*(x-10000) +
(9.3729999999999994e-03)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnosehigh = (1.0064727536231836e-12)*(x-11000)**3 +
(1.7144146086956523e-08)*(x-11000)**2 &
+ (5.2799381159420297e-05)*(x-11000) +
(4.58500000000000002e-02)
END IF
IF ((x>=12000).AND.(x<14000))THEN
qnosehigh = (1.2039449275362564e-12)*(x-12000)**3 +
(2.0163564347826031e-08)*(x-12000)**2 &
+ (9.0107091594202895e-05)*(x-12000) +
(1.1680000000000000e-01)

```

```

END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (-5.8328898550728823e-13)*(x-14000)**3 +
    (2.7387233913043545e-08)*(x-14000)**2 &
+ (1.8520868811594209e-04)*(x-14000) +
    (3.8729999999999998e-01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnosehigh = (-5.8328898550721381e-13)*(x-16000)**3 +
    (2.3887500000000005e-08)*(x-16000)**2 &
+ (2.8775815594202881e-04)*(x-16000) +
    (8.6260000000000003e-01)
END IF
  qnosel = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<9000))THEN
  qnoselow = (1.9433333333333334e-09)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnoselow = (2.8813721739130191e-13)*(x-9000)**3 +
    (6.8997417391304635e-10)*(x-9000)**2 &
+ (3.6398608695651379e-08)*(x-9000) + (1.7490000000000000
    e-05)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnoselow = (2.8813721739130504e-13)*(x-9500)**3 +
    (1.1221799999999991e-09)*(x-9500)**2 &

```

```

+ (9.4247569565217419e-07)*(x-9500) + (2.4420000000000003
    e-04)
END IF
IF ((x>=10000).AND.(x<11000))THEN
    qnoselow = (2.5285556521739101e-13)*(x-10000)**3 +
        (1.5543858260869575e-09)*(x-10000)**2 &
+ (2.2807586086956521e-06)*(x-10000) +
        (1.0319999999999999e-03)
END IF
IF ((x>=11000).AND.(x<12000))THEN
    qnoselow = (2.6895052173913178e-13)*(x-11000)**3 +
        (2.3129525217391257e-09)*(x-11000)**2 &
+ (6.1480969565217401e-06)*(x-11000) +
        (5.1200000000000004e-03)
END IF
IF ((x>=12000).AND.(x<14000))THEN
    qnoselow = (1.3488456521738970e-13)*(x-12000)**3 +
        (3.1198040869565256e-09)*(x-12000)**2 &
+ (1.1580853565217387e-05)*(x-12000) +
        (1.3849999999999999e-02)
END IF
IF ((x>=14000).AND.(x<16000))THEN
    qnoselow = (-1.5547691304347649e-13)*(x-14000)**3 +
        (3.9291114782608696e-09)*(x-14000)**2 &
+ (2.5678684695652166e-05)*(x-14000) +
        (5.0569999999999997e-02)
END IF
IF ((x>=16000).AND.(x<18000))THEN

```

```

qnoselow = (-1.5547691304348664e-13)*(x-16000)**3 +
(2.99625000000000069e-09)*(x-16000)**2 &
+ (3.9529407652173926e-05)*(x-16000) +
(1.1640000000000000e-01)
END IF
if ((x>=0).AND.(x<9000))THEN
qnosehigh = (4.1644444444444444e-09)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnosehigh = (2.7346950724637526e-13)*(x-9000)**3 +
(1.6615557391304378e-09)*(x-9000)**2 &
+ (-4.3505246376812722e-08)*(x-9000) +
(3.7480000000000000e-05)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnosehigh = (2.7346950724637612e-13)*(x-9500)**3 +
(2.0717599999999991e-09)*(x-9500)**2 &
+ (1.8231526231884065e-06)*(x-9500) + (4.652999999999998
e-04)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnosehigh = (6.4402098550724555e-13)*(x-10000)**3 +
(2.4819642608695665e-09)*(x-10000)**2 &
+ (4.1000147536231877e-06)*(x-10000) +
(1.9289999999999999e-03)
END IF
IF ((x>=11000).AND.(x<12000))THEN

```

```

      qnosehigh = (5.4966550724634622e-14)*(x-11000)**3 +
      (4.4140272173913064e-09)*(x-11000)**2 &
+ (1.0996006231884057e-05)*(x-11000) +
      (9.154999999999999e-03)
END IF
IF ((x>=12000).AND.(x<14000))THEN
      qnosehigh = (5.8204648550724539e-13)*(x-12000)**3 +
      (4.5789268695652218e-09)*(x-12000)**2 &
+ (1.9988960318840574e-05)*(x-12000) +
      (2.4620000000000000e-02)
END IF
IF ((x>=14000).AND.(x<16000))THEN
      qnosehigh = (-6.9165929710144443e-13)*(x-14000)**3 +
      (8.0712057826086858e-09)*(x-14000)**2 &
+ (4.5289225623188406e-05)*(x-14000) +
      (8.7569999999999995e-02)
END IF
IF ((x>=16000).AND.(x<18000))THEN
      qnosehigh = (-6.9165929710144817e-13)*(x-16000)**3 +
      (3.9212499999999962e-09)*(x-16000)**2 &
+ (6.9274137188405819e-05)*(x-16000) +
      (2.0490000000000000e-01)
END IF
      qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
      newqnr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
      rndiff = 1.0-(0.500000-nose)/(0.500000-0.230000)
IF ((nose>0.2300000).AND.(nose<=0.50)) THEN

```

```

IF ((x>=0).AND.(x<9000))THEN
qnoselow = (2.0366666666666667e-08)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnoselow = (2.8921675362319196e-12)*(x-9000)**3 +
(6.0023486956521253e-09)*(x-9000)**2 &
+ (2.9518376811595721e-07)*(x-9000) + (1.8330000000000001
e-04)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnoselow = (2.8921675362318897e-12)*(x-9500)**3 +
(1.03406000000000006e-08)*(x-9500)**2 &
+ (8.4666581159420222e-06)*(x-9500) + (2.1930000000000001
e-03)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnoselow = (8.2176492753623566e-13)*(x-10000)**3 +
(1.4678851304347823e-08)*(x-10000)**2 &
+ (2.0976383768115946e-05)*(x-10000) +
(9.3729999999999994e-03)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnoselow = (1.0064727536231836e-12)*(x-11000)**3 +
(1.7144146086956523e-08)*(x-11000)**2 &
+ (5.2799381159420297e-05)*(x-11000) +
(4.58500000000000002e-02)
END IF

```

```

IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (1.2039449275362564e-12)*(x-12000)**3 +
    (2.0163564347826031e-08)*(x-12000)**2 &
  + (9.0107091594202895e-05)*(x-12000) +
    (1.1680000000000000e-01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnoselow = (-5.8328898550728823e-13)*(x-14000)**3 +
    (2.7387233913043545e-08)*(x-14000)**2 &
  + (1.8520868811594209e-04)*(x-14000) +
    (3.8729999999999998e-01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnoselow = (-5.8328898550721381e-13)*(x-16000)**3 +
    (2.3887500000000005e-08)*(x-16000)**2 &
  + (2.8775815594202881e-04)*(x-16000) +
    (8.6260000000000003e-01)
END IF
if ((x>=0).AND.(x<9000))THEN
  qnosehigh = (4.1088888888888891e-08)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (4.7687501449275232e-12)*(x-9000)**3 +
    (1.0706474782608712e-08)*(x-9000)**2 &
  + (8.2497507246376275e-07)*(x-9000) + (3.6979999999999999
    e-04)
END IF

```



```

IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (4.7687501449275168e-12)*(x-9500)**3 +
    (1.78596000000000014e-08)*(x-9500)**2 &
+ (1.5108012463768117e-05)*(x-9500) + (4.0549999999999996
  e-03)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (4.0309971014493415e-13)*(x-10000)**3 +
    (2.5012725217391293e-08)*(x-10000)**2 &
+ (3.6544175072463770e-05)*(x-10000) +
    (1.66700000000000001e-02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (2.5790510144928322e-12)*(x-11000)**3 +
    (2.6222024347825996e-08)*(x-11000)**2 &
+ (8.7778924637681158e-05)*(x-11000) +
    (7.86300000000000005e-02)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (1.5416297101449475e-12)*(x-12000)**3 +
    (3.3959177391304274e-08)*(x-12000)**2 &
+ (1.4796012637681164e-04)*(x-12000) +
    (1.9520999999999999e-01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (2.8392405797109046e-13)*(x-14000)**3 +
    (4.3208955652173774e-08)*(x-14000)**2 &

```

```

+ (3.0229639246376810e-04)*(x-14000) +
    (6.3929999999999998e-01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
    qnosehigh = (2.8392405797091507e-13)*(x-16000)**3 +
        (4.4912499999999979e-08)*(x-16000)**2 &
+ (4.7853930376811629e-04)*(x-16000) +
    (1.4190000000000000e+00)
END IF
    qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<9000))THEN
    qnoselow = (4.1644444444444444e-09)*(x) +
        (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (2.7346950724637526e-13)*(x-9000)**3 +
        (1.6615557391304378e-09)*(x-9000)**2 &
+ (-4.3505246376812722e-08)*(x-9000) +
    (3.7480000000000000e-05)
END IF
IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (2.7346950724637612e-13)*(x-9500)**3 +
        (2.0717599999999991e-09)*(x-9500)**2 &
+ (1.8231526231884065e-06)*(x-9500) + (4.6529999999999998
    e-04)
END IF
IF ((x>=10000).AND.(x<11000))THEN

```

```

      qnoselow = (6.4402098550724555e-13)*(x-10000)**3 +
        (2.4819642608695665e-09)*(x-10000)**2 &
+ (4.1000147536231877e-06)*(x-10000) +
        (1.9289999999999999e-03)
END IF
IF ((x>=11000).AND.(x<12000))THEN
      qnoselow = (5.4966550724634622e-14)*(x-11000)**3 +
        (4.4140272173913064e-09)*(x-11000)**2 &
+ (1.0996006231884057e-05)*(x-11000) +
        (9.1549999999999999e-03)
END IF
IF ((x>=12000).AND.(x<14000))THEN
      qnoselow = (5.8204648550724539e-13)*(x-12000)**3 +
        (4.5789268695652218e-09)*(x-12000)**2 &
+ (1.9988960318840574e-05)*(x-12000) +
        (2.4620000000000000e-02)
END IF
IF ((x>=14000).AND.(x<16000))THEN
      qnoselow = (-6.9165929710144443e-13)*(x-14000)**3 +
        (8.0712057826086858e-09)*(x-14000)**2 &
+ (4.5289225623188406e-05)*(x-14000) +
        (8.7569999999999995e-02)
END IF
IF ((x>=16000).AND.(x<18000))THEN
      qnoselow = (-6.9165929710144817e-13)*(x-16000)**3 +
        (3.9212499999999962e-09)*(x-16000)**2 &
+ (6.9274137188405819e-05)*(x-16000) +
        (2.0490000000000000e-01)

```

```

END IF
if ((x>=0).AND.(x<9000))THEN
qnosehigh = (8.5166666666666668e-09)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnosehigh = (6.6862318840579140e-13)*(x-9000)**3 +
(2.5291652173913127e-09)*(x-9000)**2 &
+ (1.3156159420289574e-07)*(x-9000) + (7.6650000000000006
e-05)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnosehigh = (6.6862318840580463e-13)*(x-9500)**3 +
(3.5320999999999928e-09)*(x-9500)**2 &
+ (3.1621942028985519e-06)*(x-9500) + (8.582999999999999
e-04)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnosehigh = (9.7320362318840802e-13)*(x-10000)**3 +
(4.5350347826086940e-09)*(x-10000)**2 &
+ (7.1957615942028983e-06)*(x-10000) +
(3.4060000000000002e-03)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnosehigh = (5.9912318840579036e-14)*(x-11000)**3 +
(7.4546456521739117e-09)*(x-11000)**2 &
+ (1.9185442028985511e-05)*(x-11000) +
(1.6109999999999999e-02)

```

```

END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (5.6294112318840465e-13)*(x-12000)**3 +
    (7.6343826086956554e-09)*(x-12000)**2 &
+ (3.4274470289855071e-05)*(x-12000) +
    (4.2810000000000001e-02)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (-3.7283822463768287e-13)*(x-14000)**3 +
    (1.1012029347826093e-08)*(x-14000)**2 &
+ (7.1567294202898549e-05)*(x-14000) +
    (1.4640000000000000e-01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnosehigh = (-3.7283822463768620e-13)*(x-16000)**3 +
    (8.7749999999999975e-09)*(x-16000)**2 &
+ (1.1114135289855073e-04)*(x-16000) +
    (3.3060000000000000e-01)
END IF
  qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
  newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
  rndiff = 1.0-(1.000000-nose)/(1.000000-0.500000)
IF ((nose>0.5000000).AND.(nose<=1.00)) THEN
  IF ((x>=0).AND.(x<9000))THEN
    qnoselow = (4.1088888888888891e-08)*(x) +
      (0.0000000000000000e+00)
  END IF

```

```

IF ((x>=9000).AND.(x<9500))THEN
  qnoselow = (4.7687501449275232e-12)*(x-9000)**3 +
    (1.0706474782608712e-08)*(x-9000)**2 &
  + (8.2497507246376275e-07)*(x-9000) + (3.697999999999999
    e-04)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnoselow = (4.7687501449275168e-12)*(x-9500)**3 +
    (1.78596000000000014e-08)*(x-9500)**2 &
  + (1.5108012463768117e-05)*(x-9500) + (4.0549999999999996
    e-03)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (4.0309971014493415e-13)*(x-10000)**3 +
    (2.5012725217391293e-08)*(x-10000)**2 &
  + (3.6544175072463770e-05)*(x-10000) +
    (1.66700000000000001e-02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (2.5790510144928322e-12)*(x-11000)**3 +
    (2.6222024347825996e-08)*(x-11000)**2 &
  + (8.7778924637681158e-05)*(x-11000) +
    (7.86300000000000005e-02)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (1.5416297101449475e-12)*(x-12000)**3 +
    (3.3959177391304274e-08)*(x-12000)**2 &

```

```

+ (1.4796012637681164e-04)*(x-12000) +
  (1.9520999999999999e-01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnoselow = (2.8392405797109046e-13)*(x-14000)**3 +
    (4.3208955652173774e-08)*(x-14000)**2 &
+ (3.0229639246376810e-04)*(x-14000) +
  (6.3929999999999998e-01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnoselow = (2.8392405797091507e-13)*(x-16000)**3 +
    (4.4912499999999979e-08)*(x-16000)**2 &
+ (4.7853930376811629e-04)*(x-16000) +
  (1.4190000000000000e+00)
END IF
if ((x>=0).AND.(x<9000))THEN
  qnosehigh = (7.6911111111111107e-08)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (5.4395310144927675e-12)*(x-9000)**3 +
    (2.1185103478260839e-08)*(x-9000)**2 &
+ (6.1316550724638953e-07)*(x-9000) + (6.9220000000000002
  e-04)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (5.4395310144927012e-12)*(x-9500)**3 +
    (2.93444000000000049e-08)*(x-9500)**2 &

```

```

+ (2.5877917246376804e-05)*(x-9500) + (6.9750000000000003
    e-03)
END IF
IF ((x>=10000).AND.(x<11000))THEN
    qnosehigh = (2.6433797101451613e-13)*(x-10000)**3 +
        (3.7503696521739108e-08)*(x-10000)**2 &
+ (5.9301965507246378e-05)*(x-10000) +
        (2.7930000000000000e-02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
    qnosehigh = (4.3009171014492190e-12)*(x-11000)**3 +
        (3.8296710434782676e-08)*(x-11000)**2 &
+ (1.3510237246376814e-04)*(x-11000) +
        (1.2500000000000000e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
    qnosehigh = (1.7506329710144985e-12)*(x-12000)**3 +
        (5.1199461739130418e-08)*(x-12000)**2 &
+ (2.2459854463768115e-04)*(x-12000) +
        (3.0270000000000002e-01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
    qnosehigh = (2.0987340579708281e-13)*(x-14000)**3 +
        (6.1703259565217482e-08)*(x-14000)**2 &
+ (4.5040398724637682e-04)*(x-14000) +
        (9.7070000000000001e-01)
END IF
IF ((x>=16000).AND.(x<18000))THEN

```



```

      qnosehigh = (2.0987340579699678e-13)*(x-16000)**3 +
      (6.29625000000000125e-08)*(x-16000)**2 &
+ (6.9973550637681173e-04)*(x-16000) +
      (2.12000000000000001e+00)
END IF

      qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<9000))THEN
qnoselow = (8.5166666666666668e-09)*(x) +
      (0.0000000000000000e+00)
END IF

IF ((x>=9000).AND.(x<9500))THEN
      qnoselow = (6.6862318840579140e-13)*(x-9000)**3 +
      (2.5291652173913127e-09)*(x-9000)**2 &
+ (1.3156159420289574e-07)*(x-9000) + (7.66500000000000006
      e-05)
END IF

IF ((x>=9500).AND.(x<10000))THEN
      qnoselow = (6.6862318840580463e-13)*(x-9500)**3 +
      (3.5320999999999928e-09)*(x-9500)**2 &
+ (3.1621942028985519e-06)*(x-9500) + (8.582999999999999
      e-04)
END IF

IF ((x>=10000).AND.(x<11000))THEN
      qnoselow = (9.7320362318840802e-13)*(x-10000)**3 +
      (4.5350347826086940e-09)*(x-10000)**2 &
+ (7.1957615942028983e-06)*(x-10000) +
      (3.40600000000000002e-03)
END IF

```

```

IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (5.9912318840579036e-14)*(x-11000)**3 +
    (7.4546456521739117e-09)*(x-11000)**2 &
  + (1.9185442028985511e-05)*(x-11000) +
    (1.6109999999999999e-02)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (5.6294112318840465e-13)*(x-12000)**3 +
    (7.6343826086956554e-09)*(x-12000)**2 &
  + (3.4274470289855071e-05)*(x-12000) +
    (4.2810000000000001e-02)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnoselow = (-3.7283822463768287e-13)*(x-14000)**3 +
    (1.1012029347826093e-08)*(x-14000)**2 &
  + (7.1567294202898549e-05)*(x-14000) +
    (1.4640000000000000e-01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnoselow = (-3.7283822463768620e-13)*(x-16000)**3 +
    (8.7749999999999975e-09)*(x-16000)**2 &
  + (1.1114135289855073e-04)*(x-16000) +
    (3.3060000000000000e-01)
END IF
if ((x>=0).AND.(x<9000))THEN
  qnosehigh = (1.6166666666666669e-08)*(x) +
    (0.0000000000000000e+00)
END IF

```

```

IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (1.5237565217391330e-12)*(x-9000)**3 +
    (3.5093652173913019e-09)*(x-9000)**2 &
+ (5.6737826086956591e-07)*(x-9000) + (1.4550000000000001
  e-04)
END IF

IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (1.5237565217391231e-12)*(x-9500)**3 +
    (5.79500000000000041e-09)*(x-9500)**2 &
+ (5.2195608695652177e-06)*(x-9500) + (1.4970000000000001
  e-03)
END IF

IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (1.0559869565217530e-12)*(x-10000)**3 +
    (8.0806347826086822e-09)*(x-10000)**2 &
+ (1.2157378260869564e-05)*(x-10000) +
    (5.74600000000000002e-03)
END IF

IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (2.3479565217391040e-13)*(x-11000)**3 +
    (1.1248595652173906e-08)*(x-11000)**2 &
+ (3.1486608695652188e-05)*(x-11000) +
    (2.70400000000000002e-02)
END IF

IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (4.7521195652173587e-13)*(x-12000)**3 +
    (1.1952982608695666e-08)*(x-12000)**2 &

```

```

+ (5.4688186956521731e-05)*(x-12000) +
(7.0010000000000003e-02)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (-3.9029239130434264e-13)*(x-14000)**3 +
(1.4804254347826061e-08)*(x-14000)**2 &
+ (1.0820266086956523e-04)*(x-14000) +
(2.3100000000000001e-01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnosehigh = (-3.9029239130433936e-13)*(x-16000)**3 +
(1.24625000000000025e-08)*(x-16000)**2 &
+ (1.6273616956521736e-04)*(x-16000) +
(5.0349999999999995e-01)
END IF
  qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
  newqr = qnose2*rhodiff+qnosel*(1.0-rhodiff)
END IF !nose radius if
rndiff = 1.0-(2.300000-nose)/(2.300000-1.000000)
IF ((nose>1.0000000).AND.(nose<=2.30)) THEN
  IF ((x>=0).AND.(x<9000))THEN
    qnoselow = (7.6911111111111107e-08)*(x) +
(0.0000000000000000e+00)
  END IF
  IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (5.4395310144927675e-12)*(x-9000)**3 +
(2.1185103478260839e-08)*(x-9000)**2 &

```

```

+ (6.1316550724638953e-07)*(x-9000) + (6.9220000000000002
e-04)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnoselow = (5.4395310144927012e-12)*(x-9500)**3 +
    (2.93444000000000049e-08)*(x-9500)**2 &
+ (2.5877917246376804e-05)*(x-9500) + (6.9750000000000003
e-03)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (2.6433797101451613e-13)*(x-10000)**3 +
    (3.7503696521739108e-08)*(x-10000)**2 &
+ (5.9301965507246378e-05)*(x-10000) +
    (2.7930000000000000e-02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (4.3009171014492190e-12)*(x-11000)**3 +
    (3.8296710434782676e-08)*(x-11000)**2 &
+ (1.3510237246376814e-04)*(x-11000) +
    (1.2500000000000000e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (1.7506329710144985e-12)*(x-12000)**3 +
    (5.1199461739130418e-08)*(x-12000)**2 &
+ (2.2459854463768115e-04)*(x-12000) +
    (3.0270000000000002e-01)
END IF
IF ((x>=14000).AND.(x<16000))THEN

```

```

      qnoselow = (2.0987340579708281e-13)*(x-14000)**3 +
        (6.1703259565217482e-08)*(x-14000)**2 &
+ (4.5040398724637682e-04)*(x-14000) +
      (9.70700000000000001e-01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
      qnoselow = (2.0987340579699678e-13)*(x-16000)**3 +
        (6.296250000000000125e-08)*(x-16000)**2 &
+ (6.9973550637681173e-04)*(x-16000) +
      (2.12000000000000001e+00)
END IF
if ((x>=0).AND.(x<9000))THEN
      qnosehigh = (1.64000000000000001e-07)*(x) +
        (0.00000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
      qnosehigh = (1.03946666666666349e-12)*(x-9000)**3 +
        (4.987280000000000022e-08)*(x-9000)**2 &
+ (-1.80826666666666690e-06)*(x-9000) +
      (1.476000000000000001e-03)
END IF
IF ((x>=9500).AND.(x<10000))THEN
      qnosehigh = (1.039466666666667274e-12)*(x-9500)**3 +
        (5.1431999999999961e-08)*(x-9500)**2 &
+ (4.8844133333333329e-05)*(x-9500) + (1.3169999999999999
      e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN

```

```

    qnosehigh = (2.1730666666666879e-12)*(x-10000)**3 +
        (5.2991199999999992e-08)*(x-10000)**2 &
+ (1.0105573333333334e-04)*(x-10000) +
        (5.0580000000000000e-02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
    qnosehigh = (1.8322666666666100e-12)*(x-11000)**3 +
        (5.9510400000000017e-08)*(x-11000)**2 &
+ (2.1355733333333339e-04)*(x-11000) +
        (2.0680000000000001e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
    qnosehigh = (7.5151666666666556e-12)*(x-12000)**3 +
        (6.5007200000000032e-08)*(x-12000)**2 &
+ (3.3807493333333326e-04)*(x-12000) +
        (4.8170000000000002e-01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
    qnosehigh = (-6.24553333333333013e-12)*(x-14000)**3 +
        (1.1009820000000001e-07)*(x-14000)**2 &
+ (6.8828573333333325e-04)*(x-14000) +
        (1.4780000000000000e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
    qnosehigh = (-6.24553333333334370e-12)*(x-16000)**3 +
        (7.2625000000000003e-08)*(x-16000)**2 &
+ (1.0537321333333337e-03)*(x-16000) +
        (3.2450000000000001e+00)

```

```

END IF

  qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
  if ((x>=0).AND.(x<9000))THEN
    qnoselow = (1.6166666666666669e-08)*(x) +
      (0.0000000000000000e+00)
  END IF

  IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (1.5237565217391330e-12)*(x-9000)**3 +
      (3.5093652173913019e-09)*(x-9000)**2 &
    + (5.6737826086956591e-07)*(x-9000) + (1.4550000000000001
      e-04)
  END IF

  IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (1.5237565217391231e-12)*(x-9500)**3 +
      (5.79500000000000041e-09)*(x-9500)**2 &
    + (5.2195608695652177e-06)*(x-9500) + (1.4970000000000001
      e-03)
  END IF

  IF ((x>=10000).AND.(x<11000))THEN
    qnoselow = (1.0559869565217530e-12)*(x-10000)**3 +
      (8.0806347826086822e-09)*(x-10000)**2 &
    + (1.2157378260869564e-05)*(x-10000) +
      (5.7460000000000002e-03)
  END IF

  IF ((x>=11000).AND.(x<12000))THEN
    qnoselow = (2.3479565217391040e-13)*(x-11000)**3 +
      (1.1248595652173906e-08)*(x-11000)**2 &

```



```

+ (3.1486608695652188e-05)*(x-11000) +
(2.7040000000000002e-02)
END IF
IF ((x>=12000).AND.(x<14000))THEN
qnoselow = (4.7521195652173587e-13)*(x-12000)**3 +
(1.1952982608695666e-08)*(x-12000)**2 &
+ (5.4688186956521731e-05)*(x-12000) +
(7.0010000000000003e-02)
END IF
IF ((x>=14000).AND.(x<16000))THEN
qnoselow = (-3.9029239130434264e-13)*(x-14000)**3 +
(1.4804254347826061e-08)*(x-14000)**2 &
+ (1.0820266086956523e-04)*(x-14000) +
(2.3100000000000001e-01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
qnoselow = (-3.9029239130433936e-13)*(x-16000)**3 +
(1.24625000000000025e-08)*(x-16000)**2 &
+ (1.6273616956521736e-04)*(x-16000) +
(5.0349999999999995e-01)
END IF
if ((x>=0).AND.(x<9000))THEN
qnosehigh = (3.4866666666666668e-08)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnosehigh = (2.4249472463768275e-12)*(x-9000)**3 +
(7.0821791304347650e-09)*(x-9000)**2 &

```

```

+ (1.0790736231884105e-06)*(x-9000) + (3.1379999999999998
e-04)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (2.4249472463767644e-12)*(x-9500)**3 +
    (1.07196000000000032e-08)*(x-9500)**2 &
+ (9.9799631884057957e-06)*(x-9500) + (2.9269999999999999
e-03)
END IF
IF ((x>=10000).AND.(x<11000))THEN
  qnosehigh = (1.8047055072463792e-12)*(x-10000)**3 +
    (1.4357020869565222e-08)*(x-10000)**2 &
+ (2.2518273623188401e-05)*(x-10000) +
    (1.0900000000000000e-02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (-2.9975692753623145e-12)*(x-11000)**3 +
    (1.9771137391304330e-08)*(x-11000)**2 &
+ (5.6646431884057982e-05)*(x-11000) +
    (4.9579999999999999e-02)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (3.4617855072463729e-12)*(x-12000)**3 +
    (1.0778429565217416e-08)*(x-12000)**2 &
+ (8.7195998840579699e-05)*(x-12000) +
    (1.2300000000000000e-01)
END IF
IF ((x>=14000).AND.(x<16000))THEN

```

```

      qnosehigh = (-1.9748571014492821e-12)*(x-14000)**3 +
      (3.1549142608695636e-08)*(x-14000)**2 &
+ (1.7185114318840584e-04)*(x-14000) +
      (3.6820000000000003e-01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
      qnosehigh = (-1.9748571014492821e-12)*(x-16000)**3 +
      (1.97000000000000047e-08)*(x-16000)**2 &
+ (2.7434942840579701e-04)*(x-16000) +
      (8.2230000000000003e-01)
END IF
      qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
      newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
      rndiff = 1.0-(5.000000-nose)/(5.000000-2.300000)
IF ((nose>2.3000000).AND.(nose<=5.00)) THEN
      IF ((x>=0).AND.(x<9000))THEN
      qnoselow = (1.6400000000000001e-07)*(x) +
      (0.0000000000000000e+00)
END IF
      IF ((x>=9000).AND.(x<9500))THEN
      qnoselow = (1.03946666666666349e-12)*(x-9000)**3 +
      (4.98728000000000022e-08)*(x-9000)**2 &
+ (-1.8082666666666690e-06)*(x-9000) +
      (1.4760000000000001e-03)
END IF
      IF ((x>=9500).AND.(x<10000))THEN

```

```

qnoselow = (1.0394666666667274e-12)*(x-9500)**3 +
(5.1431999999999961e-08)*(x-9500)**2 &
+ (4.8844133333333329e-05)*(x-9500) + (1.3169999999999999
e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnoselow = (2.17306666666666879e-12)*(x-10000)**3 +
(5.2991199999999992e-08)*(x-10000)**2 &
+ (1.0105573333333334e-04)*(x-10000) +
(5.0580000000000000e-02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnoselow = (1.83226666666666100e-12)*(x-11000)**3 +
(5.95104000000000017e-08)*(x-11000)**2 &
+ (2.1355733333333339e-04)*(x-11000) +
(2.06800000000000001e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
qnoselow = (7.51516666666666556e-12)*(x-12000)**3 +
(6.50072000000000032e-08)*(x-12000)**2 &
+ (3.3807493333333326e-04)*(x-12000) +
(4.81700000000000002e-01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
qnoselow = (-6.24553333333333013e-12)*(x-14000)**3 +
(1.10098200000000001e-07)*(x-14000)**2 &
+ (6.8828573333333325e-04)*(x-14000) +
(1.4780000000000000e+00)

```

```

END IF
IF ((x>=16000).AND.(x<18000))THEN
    qnoselow = (-6.24553333333334370e-12)*(x-16000)**3 +
        (7.26250000000000003e-08)*(x-16000)**2 &
+ (1.05373213333333337e-03)*(x-16000) +
        (3.24500000000000001e+00)
END IF
if ((x>=0).AND.(x<9000))THEN
    qnosehigh = (3.3144444444444442e-07)*(x) +
        (0.00000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
    qnosehigh = (-8.2515420289854076e-12)*(x-9000)**3 +
        (9.1103313043478111e-08)*(x-9000)**2 &
+ (-2.3347710144926981e-06)*(x-9000) +
        (2.98300000000000000e-03)
END IF
IF ((x>=9500).AND.(x<10000))THEN
    qnosehigh = (-8.2515420289854076e-12)*(x-9500)**3 +
        (7.87260000000000013e-08)*(x-9500)**2 &
+ (8.2579885507246356e-05)*(x-9500) + (2.35600000000000001
        e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
    qnosehigh = (4.6340840579710548e-12)*(x-10000)**3 +
        (6.6348686956521597e-08)*(x-10000)**2 &
+ (1.5511722898550731e-04)*(x-10000) +
        (8.35000000000000005e-02)

```

```

END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnosehigh = (2.2322057971013796e-12)*(x-11000)**3 +
    (8.0250939130434907e-08)*(x-11000)**2 &
+ (3.0171685507246368e-04)*(x-11000) +
    (3.0959999999999999e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (3.6973840579710279e-12)*(x-12000)**3 +
    (8.6947556521739099e-08)*(x-12000)**2 &
+ (4.6891535072463762e-04)*(x-12000) +
    (6.9379999999999997e-01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (-3.3344768115941887e-12)*(x-14000)**3 +
    (1.0913186086956525e-07)*(x-14000)**2 &
+ (8.6107418550724637e-04)*(x-14000) +
    (2.0089999999999999e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnosehigh = (-3.3344768115942945e-12)*(x-16000)**3 +
    (8.9125000000000012e-08)*(x-16000)**2 &
+ (1.2575879072463771e-03)*(x-16000) +
    (4.1410000000000000e+00)
END IF
  qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
if ((x>=0).AND.(x<9000))THEN

```

```

qnoselow = (3.4866666666666668e-08)*(x) +
            (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (2.4249472463768275e-12)*(x-9000)**3 +
                (7.0821791304347650e-09)*(x-9000)**2 &
                + (1.0790736231884105e-06)*(x-9000) + (3.1379999999999998
                e-04)
END IF
IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (2.4249472463767644e-12)*(x-9500)**3 +
                (1.0719600000000032e-08)*(x-9500)**2 &
                + (9.9799631884057957e-06)*(x-9500) + (2.9269999999999999
                e-03)
END IF
IF ((x>=10000).AND.(x<11000))THEN
    qnoselow = (1.8047055072463792e-12)*(x-10000)**3 +
                (1.4357020869565222e-08)*(x-10000)**2 &
                + (2.2518273623188401e-05)*(x-10000) +
                (1.0900000000000000e-02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
    qnoselow = (-2.9975692753623145e-12)*(x-11000)**3 +
                (1.9771137391304330e-08)*(x-11000)**2 &
                + (5.6646431884057982e-05)*(x-11000) +
                (4.9579999999999999e-02)
END IF
IF ((x>=12000).AND.(x<14000))THEN

```

```

      qnoselow = (3.4617855072463729e-12)*(x-12000)**3 +
      (1.0778429565217416e-08)*(x-12000)**2 &
+ (8.7195998840579699e-05)*(x-12000) +
      (1.2300000000000000e-01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
      qnoselow = (-1.9748571014492821e-12)*(x-14000)**3 +
      (3.1549142608695636e-08)*(x-14000)**2 &
+ (1.7185114318840584e-04)*(x-14000) +
      (3.6820000000000003e-01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
      qnoselow = (-1.9748571014492821e-12)*(x-16000)**3 +
      (1.97000000000000047e-08)*(x-16000)**2 &
+ (2.7434942840579701e-04)*(x-16000) +
      (8.2230000000000003e-01)
END IF
if ((x>=0).AND.(x<9000))THEN
      qnosehigh = (7.1000000000000000e-08)*(x) +
      (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
      qnosehigh = (3.8719594202898733e-12)*(x-9000)**3 +
      (1.3186060869565177e-08)*(x-9000)**2 &
+ (1.7529797101449446e-06)*(x-9000) + (6.3900000000000003
      e-04)
END IF
IF ((x>=9500).AND.(x<10000))THEN

```



```

qnosehigh = (3.8719594202898733e-12)*(x-9500)**3 +
(1.89940000000000005e-08)*(x-9500)**2 &
+ (1.7843010144927526e-05)*(x-9500) + (5.2960000000000004
e-03)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnosehigh = (-1.0329188405796915e-12)*(x-10000)**3 +
(2.4801939130434762e-08)*(x-10000)**2 &
+ (3.9740979710144936e-05)*(x-10000) +
(1.9449999999999999e-02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnosehigh = (2.0907159420290229e-12)*(x-11000)**3 +
(2.1703182608695588e-08)*(x-11000)**2 &
+ (8.6246101449275386e-05)*(x-11000) +
(8.29600000000000006e-02)
END IF
IF ((x>=12000).AND.(x<14000))THEN
qnosehigh = (5.3118115942030230e-13)*(x-12000)**3 +
(2.7975330434782564e-08)*(x-12000)**2 &
+ (1.3592461449275363e-04)*(x-12000) +
(1.93000000000000000e-01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
qnosehigh = (-6.2362318840698277e-15)*(x-14000)**3 +
(3.1162417391304375e-08)*(x-14000)**2 &
+ (2.5420011014492752e-04)*(x-14000) +
(5.8099999999999996e-01)

```

```

END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnosehigh = (-6.2362318840565930e-15)*(x-16000)**3 +
    (3.11250000000000036e-08)*(x-16000)**2 &
+ (3.7877494492753619e-04)*(x-16000) +
    (1.2140000000000000e+00)
END IF
  qnose2 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
  newqr = qnose2*rhodiff+qnose1*(1.0-rhodiff)
END IF !nose radius if
  rndiff = 1.0-(10.000000-nose)/(10.000000-5.000000)
IF ((nose>5.0000000).AND.(nose<=10.00)) THEN
  IF ((x>=0).AND.(x<9000))THEN
    qnoselow = (3.3144444444444442e-07)*(x) +
      (0.0000000000000000e+00)
  END IF
  IF ((x>=9000).AND.(x<9500))THEN
    qnoselow = (-8.2515420289854076e-12)*(x-9000)**3 +
      (9.1103313043478111e-08)*(x-9000)**2 &
+ (-2.3347710144926981e-06)*(x-9000) +
      (2.9830000000000000e-03)
  END IF
  IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (-8.2515420289854076e-12)*(x-9500)**3 +
      (7.87260000000000013e-08)*(x-9500)**2 &
+ (8.2579885507246356e-05)*(x-9500) + (2.35600000000000001
    e-02)
  END IF

```

```

IF ((x>=10000).AND.(x<11000))THEN
  qnoselow = (4.6340840579710548e-12)*(x-10000)**3 +
    (6.6348686956521597e-08)*(x-10000)**2 &
+ (1.5511722898550731e-04)*(x-10000) +
    (8.3500000000000005e-02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
  qnoselow = (2.2322057971013796e-12)*(x-11000)**3 +
    (8.0250939130434907e-08)*(x-11000)**2 &
+ (3.0171685507246368e-04)*(x-11000) +
    (3.0959999999999999e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnoselow = (3.6973840579710279e-12)*(x-12000)**3 +
    (8.6947556521739099e-08)*(x-12000)**2 &
+ (4.6891535072463762e-04)*(x-12000) +
    (6.9379999999999997e-01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnoselow = (-3.3344768115941887e-12)*(x-14000)**3 +
    (1.0913186086956525e-07)*(x-14000)**2 &
+ (8.6107418550724637e-04)*(x-14000) +
    (2.0089999999999999e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnoselow = (-3.3344768115942945e-12)*(x-16000)**3 +
    (8.9125000000000012e-08)*(x-16000)**2 &

```

```

+ (1.2575879072463771e-03)*(x-16000) +
(4.1410000000000000e+00)
END IF
if ((x>=0).AND.(x<9000))THEN
qnosehigh = (6.2555555555555548e-07)*(x) +
(0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
qnosehigh = (-1.8864840579710335e-11)*(x-9000)**3 +
(1.3415726086956550e-07)*(x-9000)**2 &
+ (4.1775797101448188e-06)*(x-9000) + (5.6299999999999996
e-03)
END IF
IF ((x>=9500).AND.(x<10000))THEN
qnosehigh = (-1.8864840579709963e-11)*(x-9500)**3 +
(1.0585999999999984e-07)*(x-9500)**2 &
+ (1.2418621014492756e-04)*(x-9500) + (3.8899999999999997
e-02)
END IF
IF ((x>=10000).AND.(x<11000))THEN
qnosehigh = (6.4396811594203005e-12)*(x-10000)**3 +
(7.7562739130434799e-08)*(x-10000)**2 &
+ (2.1589757971014493e-04)*(x-10000) +
(1.2509999999999999e-01)
END IF
IF ((x>=11000).AND.(x<12000))THEN
qnosehigh = (1.9761159420287617e-12)*(x-11000)**3 +
(9.6881782608695886e-08)*(x-11000)**2 &

```

```

+ (3.9034210144927542e-04)*(x-11000) +
(4.2499999999999999e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
  qnosehigh = (-1.5635688405796161e-12)*(x-12000)**3 +
(1.0281013043478244e-07)*(x-12000)**2 &
+ (5.9003401449275348e-04)*(x-12000) +
(9.14200000000000001e-01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnosehigh = (1.0327137681158848e-12)*(x-14000)**3 +
(9.3428717391304355e-08)*(x-14000)**2 &
+ (9.8251171014492782e-04)*(x-14000) +
(2.4929999999999999e+00)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnosehigh = (1.0327137681159973e-12)*(x-16000)**3 +
(9.96250000000000100e-08)*(x-16000)**2 &
+ (1.3686191449275359e-03)*(x-16000) +
(4.8399999999999999e+00)
END IF
  qnose1 = qnosehigh*rndiff+qnoselow*(1.0-rndiff)
  if ((x>=0).AND.(x<9000))THEN
    qnoselow = (7.1000000000000000e-08)*(x) +
(0.0000000000000000e+00)
  END IF
  IF ((x>=9000).AND.(x<9500))THEN

```

```

qnoselow = (3.8719594202898733e-12)*(x-9000)**3 +
            (1.3186060869565177e-08)*(x-9000)**2 &
+ (1.7529797101449446e-06)*(x-9000) + (6.3900000000000003
            e-04)
END IF
IF ((x>=9500).AND.(x<10000))THEN
    qnoselow = (3.8719594202898733e-12)*(x-9500)**3 +
                (1.8994000000000005e-08)*(x-9500)**2 &
+ (1.7843010144927526e-05)*(x-9500) + (5.2960000000000004
            e-03)
END IF
IF ((x>=10000).AND.(x<11000))THEN
    qnoselow = (-1.0329188405796915e-12)*(x-10000)**3 +
                (2.4801939130434762e-08)*(x-10000)**2 &
+ (3.9740979710144936e-05)*(x-10000) +
            (1.9449999999999999e-02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
    qnoselow = (2.0907159420290229e-12)*(x-11000)**3 +
                (2.1703182608695588e-08)*(x-11000)**2 &
+ (8.6246101449275386e-05)*(x-11000) +
            (8.2960000000000006e-02)
END IF
IF ((x>=12000).AND.(x<14000))THEN
    qnoselow = (5.3118115942030230e-13)*(x-12000)**3 +
                (2.7975330434782564e-08)*(x-12000)**2 &
+ (1.3592461449275363e-04)*(x-12000) +
            (1.9300000000000000e-01)

```

```

END IF
IF ((x>=14000).AND.(x<16000))THEN
  qnoselow = (-6.2362318840698277e-15)*(x-14000)**3 +
    (3.1162417391304375e-08)*(x-14000)**2 &
+ (2.5420011014492752e-04)*(x-14000) +
    (5.8099999999999996e-01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
  qnoselow = (-6.2362318840565930e-15)*(x-16000)**3 +
    (3.11250000000000036e-08)*(x-16000)**2 &
+ (3.7877494492753619e-04)*(x-16000) +
    (1.2140000000000000e+00)
END IF
if ((x>=0).AND.(x<9000))THEN
  qnosehigh = (1.3322222222222223e-07)*(x) +
    (0.0000000000000000e+00)
END IF
IF ((x>=9000).AND.(x<9500))THEN
  qnosehigh = (2.3490695652173456e-12)*(x-9000)**3 +
    (2.5954395652173973e-08)*(x-9000)**2 &
+ (1.9575347826086763e-06)*(x-9000) + (1.1990000000000000
    e-03)
END IF
IF ((x>=9500).AND.(x<10000))THEN
  qnosehigh = (2.3490695652173719e-12)*(x-9500)**3 +
    (2.9478000000000001e-08)*(x-9500)**2 &
+ (2.9673732608695658e-05)*(x-9500) + (8.9599999999999992
    e-03)

```

```

END IF
IF ((x>=10000).AND.(x<11000))THEN
    qnosehigh = (-2.0751391304347906e-12)*(x-10000)**3 +
        (3.3001604347826099e-08)*(x-10000)**2 &
+ (6.0913534782608690e-05)*(x-10000) +
        (3.1460000000000002e-02)
END IF
IF ((x>=11000).AND.(x<12000))THEN
    qnosehigh = (1.6324869565217235e-12)*(x-11000)**3 +
        (2.6776186956521717e-08)*(x-11000)**2 &
+ (1.2069132608695652e-04)*(x-11000) +
        (1.2330000000000001e-01)
END IF
IF ((x>=12000).AND.(x<14000))THEN
    qnosehigh = (1.5153858695652125e-12)*(x-12000)**3 +
        (3.1673647826087007e-08)*(x-12000)**2 &
+ (1.7914116086956513e-04)*(x-12000) +
        (2.7239999999999998e-01)
END IF
IF ((x>=14000).AND.(x<16000))THEN
    qnosehigh = (-1.0755771739130596e-12)*(x-14000)**3 +
        (4.0765963043478301e-08)*(x-14000)**2 &
+ (3.2402038260869571e-04)*(x-14000) +
        (7.6949999999999996e-01)
END IF
IF ((x>=16000).AND.(x<18000))THEN
    qnosehigh = (-1.0755771739130462e-12)*(x-16000)**3 +
        (3.4312500000000004e-08)*(x-16000)**2 &

```



```

!      Routine to solve density from pressure and enthalpy
      using a
!      curvefit of the Vought data. The routine requires an
      iterative
!      solution of solving pressure from enthalpy and density
      until
!      the desired pressure is found.

```

```

!*****

```

```

SUBROUTINE DENS_FIT(p_Pa, h_Jkg, rho, X, Y, Z, LX, LY, N)

```

```

  IMPLICIT NONE

```

```

  DOUBLE PRECISION p_Pa, h_Jkg, rho, X, Y, Z

```

```

  INTEGER LX, LY, N

```

```

  ! p_Pa in Pa

```

```

  ! h_Jkg in J/kg

```

```

  ! rho in kg/m3

```

```

  ! Constants

```

```

  INTEGER, PARAMETER :: IU = 0

```

```

  ! Declare Variables

```

```

  LOGICAL DONE, FIND

```

```

  INTEGER I, LX1, LX2, LY1, LY2

```

```

  DOUBLE PRECISION U, V, W, p, h

```

```

  DOUBLE PRECISION VSTEP, dens_min, dens_max, temp

```

```

! Declare Arrays
DIMENSION X(LX) , Y(LY) , Z(LX,LY) , U(N) , V(N) , W(N)

! Convert inputs
p = p_Pa / 101325.0 ! atm
h = h_Jkg / 82731.168 ! Vought non-dimensional

! Determine valid range of data from data arrays
if ((h >= 45.0) .AND. (h < 90.0)) then

    dens_min = 0.12251
    dens_max = 122.51
    LX1 = 1
    LY1 = 8
    LX2 = 24
    LY2 = 12

else if ((h >= 90.0) .AND. (h <= 6000.0)) then

    dens_min = 0.0000012251
    dens_max = 122.51
    LX1 = 2
    LY1 = 1
    LX2 = 25
    LY2 = 12

else if ((h > 6000.0) .AND. (h <= 10000.0)) then

```

```

dens_min = 0.0000012251
dens_max = 12.251
LX1 = 2
LY1 = 1
LX2 = 26
LY2 = 11

else if ((h > 10000.0) .AND. (h <= 20000.0)) then

    dens_min = 0.0000012251
    dens_max = 0.12251
    LX1 = 2
    LY1 = 1
    LX2 = 27
    LY2 = 8

else if ((h > 20000.0) .AND. (h <= 35000.0)) then

    dens_min = 0.0000012251
    dens_max = 0.000048772
    LX1 = 2
    LY1 = 1
    LX2 = 28
    LY2 = 4

else

    ! Out of range, but try to solve anyway

```

```

!write(1,*) "Enthalpy out of range at h = ", h *
      0.082731168, " MJ/kg"
dens_min = 0.0000012251
dens_max = 122.51
LX1 = 2
LY1 = 1
LX2 = 24
LY2 = 12

endif

! Set range of possible densities
V(1) = dens_min
V(N) = dens_max

! Loop until solution found
DONE = .FALSE.
do 100 while (.NOT. DONE)

! Calculate range steps
VSTEP = (V(N) - V(1))/(N-1.0)
U(1) = h
U(N) = h
do 10 I=2,N-1
  V(I) = V(1) + (I-1)*VSTEP
  U(I) = h
10 continue

```

```

! Get pressure data using curve fit code
call ITPLBV(IU, LX2-LX1+1, LY2-LY1+1, X(LX1:LX2), Y(LY1:LY2
), &
Z(LX1:LX2,LY1:LY2), N, U, V, W)

FIND = .FALSE.

! Account for out of range cases
if (p > W(N)) then
  !write(1,*) "OUT OF RANGE AT H = ", h * 0.082731168, " MJ/
    kg AND P = ",p*0.101325, " MPa"
  EXIT
  V(1) = V(N)
  V(N) = 2 * V(N)
  FIND = .TRUE.
end if

if (p < W(1)) then
  !write(1,*) "OUT OF RANGE AT H = ", h * 0.082731168, " MJ/
    kg AND P = ", P*0.101325, " MPa"
  EXIT
  V(N) = V(1)
  V(1) = dens_min
  FIND = .TRUE.
end if

! Step through new pressure values , W()
do 30 I=2,N

```

```

! Determine if correct value
temp = abs((P - W(I))/P)
if (temp < 0.001) then
  rho = V(I)
  FIND = .TRUE.
  DONE = .TRUE.
end if

! Determine new range
if (P <= W(I)) then
  if (FIND .EQV. .FALSE.) then
    V(N) = V(I)
    V(1) = V(I-1)
    FIND = .TRUE.
  end if
end if
30  end do

100 end do

RETURN
END SUBROUTINE DENS_FIT

```

```

!*****

```

```

! TEMP_FIT

```

```

!      Routine to solve temperature from pressure and enthalpy
      using a
!      curve fit of the Vought data. The routine requires an
      iterative
!      solution of solving enthalpy from temperature and
      pressure until
!      the desired enthalpy is found.

```

```

!*****

```

```

SUBROUTINE TEMP_FIT(h_Jkg, p_Pa, t, X, Y, Z, LX, LY, N)

```

```

  IMPLICIT NONE

```

```

  DOUBLE PRECISION h_Jkg, p_Pa, t, X, Y, Z

```

```

  INTEGER LX, LY, N

```

```

  ! h_Jkg in J/kg

```

```

  ! p_Pa in Pa

```

```

  ! T in Kelvin

```

```

  ! Constants

```

```

  INTEGER, PARAMETER :: IU = 0

```

```

  ! Declared Variables

```

```

  LOGICAL DONE, FIND

```

```

  INTEGER I, LX1, LX2, LY1, LY2

```

```

  DOUBLE PRECISION U, V, W, p, h

```

```

  DOUBLE PRECISION VSTEP, temp_min, temp_max, temp

```



```

! Declared Arrays
DIMENSION X(LX) , Y(LY) , Z(LX,LY) , U(N) , V(N) , W(N)

! Convert Inputs
h = h_Jkg / 82731.168 ! Vought non-dimensional
p = p_Pa / 101325.0    ! atm

! Determine valid range of data from data arrays
if ((p >= 0.00002) .AND. (p < 0.00005)) then

    temp_min = 3000.0
    temp_max = 4000.0
    LX1 = 1
    LY1 = 1
    LX2 = 24
    LY2 = 6

else if ((p >= 0.00005) .AND. (p < 0.0001)) then

    temp_min = 3000.0
    temp_max = 7000.0
    LX1 = 2
    LY1 = 1
    LX2 = 24
    LY2 = 15

else if ((p >= 0.0001) .AND. (p < 0.0002)) then

```

```

temp_min = 3000.0
temp_max = 9000.0
LX1 = 3
LY1 = 1
LX2 = 24
LY2 = 19

else if ((p >= 0.0002) .AND. (p < 0.0005)) then

temp_min = 3000.0
temp_max = 15000.0
LX1 = 4
LY1 = 1
LX2 = 24
LY2 = 24

else if ((p >= 0.0005) .AND. (p < 0.001)) then

temp_min = 3000.0
temp_max = 25000.0
LX1 = 5
LY1 = 1
LX2 = 24
LY2 = 30

else if ((p >= 0.001) .AND. (p < 0.002)) then

temp_min = 3000.0

```

```

temp_max = 40000.0
LX1 = 6
LY1 = 1
LX2 = 24
LY2 = 35

else if ((p >= 0.002) .AND. (p < 0.005)) then

temp_min = 3000.0
temp_max = 55000.0
LX1 = 7
LY1 = 1
LX2 = 24
LY2 = 38

else if ((p >= 0.005) .AND. (p <= 1000.0)) then

temp_min = 3000.0
temp_max = 100000.0
LX1 = 8
LY1 = 1
LX2 = 24
LY2 = 43

else if ((p > 1000.0) .AND. (p <= 1250.0)) then

temp_min = 3600.0
temp_max = 100000.0

```

LX1 = 8

LY1 = 4

LX2 = 25

LY2 = 43

else if ((p > 1250.0) .AND. (p <= 2000.0)) then

temp_min = 5500.0

temp_max = 100000.0

LX1 = 8

LY1 = 12

LX2 = 26

LY2 = 43

else if ((p > 2000.0) .AND. (p <= 2500.0)) then

temp_min = 6500.0

temp_max = 100000.0

LX1 = 8

LY1 = 14

LX2 = 27

LY2 = 43

else if ((p > 2500.0) .AND. (p <= 3000.0)) then

temp_min = 7500.0

temp_max = 100000.0

LX1 = 8

LY1 = 16

LX2 = 28

LY2 = 43

else if ((p > 3000.0) .AND. (p <= 5000.0)) then

temp_min = 11000.0

temp_max = 100000.0

LX1 = 8

LY1 = 22

LX2 = 29

LY2 = 43

else if ((p > 5000.0) .AND. (p <= 10000.0)) then

temp_min = 16000.0

temp_max = 100000.0

LX1 = 8

LY1 = 25

LX2 = 30

LY2 = 43

else if ((p > 10000.0) .AND. (p <= 20000.0)) then

temp_min = 25000.0

temp_max = 100000.0

LX1 = 8

LY1 = 30

LX2 = 31

LY2 = 43

else if ((p > 20000.0) .AND. (p <= 40000.0)) then

temp_min = 45000.0

temp_max = 100000.0

LX1 = 8

LY1 = 36

LX2 = 32

LY2 = 43

else if ((p > 40000.0) .AND. (p <= 60000.0)) then

temp_min = 60000.0

temp_max = 100000.0

LX1 = 8

LY1 = 39

LX2 = 33

LY2 = 43

else if ((p > 60000.0) .AND. (p <= 80000.0)) then

temp_min = 70000.0

temp_max = 100000.0

LX1 = 8

LY1 = 40

LX2 = 34

```

LY2 = 43

else if ((p > 80000.0) .AND. (p <= 100000.0)) then

temp_min = 80000.0
temp_max = 100000.0
LX1 = 8
LY1 = 41
LX2 = 35
LY2 = 43

else

! Out of range, but try to solve anyway
write(1,*) "Pressure out of range at P = ", p * 0.101325, "
      MPa"
temp_min = 3000.0
temp_max = 100000.0
LX1 = 8
LY1 = 1
LX2 = 24
LY2 = 43

endif

! Set a range of possible temperatures
V(1) = temp_min
V(N) = temp_max

```

```

! Loop until solution found
DONE = .FALSE.
do 100 while (.NOT. DONE)

    ! Calculate range steps
    VSTEP = (V(N) - V(1))/(N-1.0)
    U(1) = P
    U(N) = P
    do 10 I=2,N-1
        V(I) = V(1) + (I-1)*VSTEP
        U(I) = p
10    continue

    ! Get enthalpy data using curve fit code
    call ITPLBV(IU, LX2-LX1+1, LY2-LY1+1, X(LX1:LX2), Y(LY1:LY2)
        ), &
        Z(LX1:LX2,LY1:LY2), N, U, V, W

    FIND = .FALSE.
    ! Account for out of range cases
    if (h > W(N)) then
        write(1,*) "OUT OF RANGE AT p = ", P*0.101325, " MPa AND h
            = ", h*19.76/238.845896627, " MJ/kg"
        EXIT
    V(1) = V(N)
    V(N) = 2 * V(N)
    FIND = .TRUE.

```



```

end if

if (h < W(1)) then
  write(1,*) "OUT OF RANGE AT p = ", P*0.101325, " MPa AND h
    = ", h*19.76/238.845896627, " MJ/kg"
  EXIT
  V(N) = V(1)
  V(1) = temp_min
  FIND = .TRUE.
end if

! Step through new enthalpy values , W()
do 30 I=2,N

  ! Determine if correct value
  temp = abs((h - W(I))/h)
  if (temp < 0.001) then
    t = V(I)
    FIND = .TRUE.
    DONE = .TRUE.
  end if

  ! Determine new range
  if (h <= W(I)) then
    if (FIND .EQV. .FALSE.) then
      V(N) = V(I)
      V(1) = V(I-1)
      FIND = .TRUE.
    end if
  end if
end do

```

```

        end if
    end if
30  end do

100 end do

RETURN
END SUBROUTINE TEMP_FIT

```

```

!*****

```

```

! TEMPNEW
! This is a routine written by Rich Fought to return the
  temperature
! for a given pressure and enthalpy. It is based on Roop
  Gupta's
! work and is valid from 500 K to 32600 K and  $10^{-4}$  atm to
  1000 atm.

```

```

!*****

```

```

SUBROUTINE TEMPNEW(H,P,T)
  IMPLICIT NONE

```

```

  DOUBLE PRECISION H, P, T
  ! H in J/kg
  ! P in Pa

```

! T in K

DOUBLE PRECISION CHIS, THETAS, BPS

DOUBLE PRECISION HDATANEW, A, THETAMAX, CHI

INTEGER I, J, BP, ZN

DIMENSION HDATANEW(36,6), A(6), THETAMAX(8), CHI(2)

DIMENSION BP(2)

OPEN(UNIT=20, FILE='hdatanew.dat', STATUS='OLD')

DO 1 I=1,36

READ(20,*) (HDATANEW(I,J), J=1,6)

1 CONTINUE

DO 2 I=1,8

READ(20,*) THETAMAX(I)

2 CONTINUE

CLOSE(20)

BPS=LOG10(P/101325.0)

THETAS=LOG(H/4186800.0)

IF (BPS.GT.0) THEN

BP(2)=AINT(BPS)+1

BP(1)=AINT(BPS)

```

        IF (THETAS.GT.THETAMAX(BP(2)+5)) THEN
BP(2)=BP(1)
        BP(1)=BP(1)-1
        END IF
ELSE
BP(2)=AINT(BPS)
        BP(1)=AINT(BPS)-1
        IF (THETAS.GT.THETAMAX(BP(2)+5)) THEN
BP(2)=BP(1)
        BP(1)=BP(1)-1
        END IF
        IF (BPS.LT.-4) THEN
BP(1)=-3
        BP(2)=-4
        END IF
END IF

DO 3 I=1,2

IF (BP(I).EQ.-4) THEN
IF (THETAS.LE.-2.085) THEN
CHI(I)=THETAS-0.87547
GOTO 4
        END IF
        IF (THETAS.GT.-2.085.AND.THETAS.LE.-0.0846) ZN=1
        IF (THETAS.GT.-0.0846.AND.THETAS.LE.1.284) ZN=2
        IF (THETAS.GT.1.284.AND.THETAS.LE.2.227) ZN=3
        IF (THETAS.GT.2.227.AND.THETAS.LE.2.3065) ZN=4

```

```

        IF (THETAS.GT.2.3065.AND.THETAS.LE.3.607) ZN=5
        IF (THETAS.GT.3.607.AND.THETAS.LE.4.172) ZN=6
        IF (THETAS.GT.4.172.AND.THETAS.LE.4.652) ZN=7
        IF (THETAS.GT.4.652) THEN
WRITE(6,*) 'Enthalpy out of range '
GOTO 4
        END IF
END IF

        IF (BP(I).EQ.-3) THEN
IF (THETAS.LE.-2.087) THEN
CHI(I)=THETAS-0.87547
GOTO 4
        END IF
        IF (THETAS.GT.-2.087.AND.THETAS.LE.-0.2788) ZN=8
        IF (THETAS.GT.-0.2788.AND.THETAS.LE.0.9678) ZN=9
        IF (THETAS.GT.0.9678.AND.THETAS.LE.2.247) ZN=10
        IF (THETAS.GT.2.247.AND.THETAS.LE.3.607) ZN=11
        IF (THETAS.GT.3.607.AND.THETAS.LE.4.005) ZN=12
        IF (THETAS.GT.4.005.AND.THETAS.LE.4.686) ZN=13
        IF (THETAS.GT.4.686) THEN
WRITE(6,*) 'Enthalpy out of range '
GOTO 4
        END IF
END IF

        IF (BP(I).EQ.-2) THEN
IF (THETAS.LE.-2.086) THEN

```

```

      CHI(I)=THETAS-0.87547
      GOTO 4
    END IF
    IF (THETAS.GT. -2.086.AND.THETAS.LE.0.1125) ZN=14
    IF (THETAS.GT.0.1125.AND.THETAS.LE.1.498) ZN=15
    IF (THETAS.GT.1.498.AND.THETAS.LE.2.748) ZN=16
    IF (THETAS.GT.2.748.AND.THETAS.LE.3.779) ZN=17
    IF (THETAS.GT.3.779.AND.THETAS.LE.4.689) ZN=18
    IF (THETAS.GT.4.689) THEN
      WRITE(6,*) 'Enthalpy out of range '
      GOTO 4
    END IF
  END IF

    IF (BP(I).EQ.-1) THEN
  IF (THETAS.LE. -2.093) THEN
    CHI(I)=THETAS-0.87547
    GOTO 4
  END IF
    IF (THETAS.GT. -2.093.AND.THETAS.LE.0.352) ZN=19
    IF (THETAS.GT.0.352.AND.THETAS.LE.1.837) ZN=20
    IF (THETAS.GT.1.837.AND.THETAS.LE.3.644) ZN=21
    IF (THETAS.GT.3.644.AND.THETAS.LE.4.648) ZN=22
    IF (THETAS.GT.4.648) THEN
      WRITE(6,*) 'Enthalpy out of range '
      GOTO 4
    END IF
  END IF

```

```

        IF (BP(I).EQ.0) THEN
IF (THETAS.LE. -2.096) THEN
    CHI(I)=THETAS-0.87547
    GOTO 4
        END IF
        IF (THETAS.GT. -2.096.AND.THETAS.LE.0.485) ZN=23
        IF (THETAS.GT.0.485.AND.THETAS.LE.2.269) ZN=24
        IF (THETAS.GT.2.269.AND.THETAS.LE.3.659) ZN=25
        IF (THETAS.GT.3.659.AND.THETAS.LE.4.365) ZN=26
        IF (THETAS.GT.4.365) THEN
WRITE(6,*) 'Enthalpy out of range '
    GOTO 4
        END IF
END IF

        IF (BP(I).EQ.1) THEN
IF (THETAS.LE. -2.097) THEN
    CHI(I)=THETAS-0.87547
    GOTO 4
        END IF
        IF (THETAS.GT. -2.097.AND.THETAS.LE.0.555) ZN=27
        IF (THETAS.GT.0.555.AND.THETAS.LE.2.265) ZN=28
        IF (THETAS.GT.2.265.AND.THETAS.LE.3.468) ZN=29
        IF (THETAS.GT.3.468.AND.THETAS.LE.4.043) ZN=30
        IF (THETAS.GT.4.043) THEN
WRITE(6,*) 'Enthalpy out of range '
    GOTO 4

```

```

                END IF
END IF

                IF (BP(I).EQ.2) THEN
IF (THETAS.LE.-2.117) THEN
    CHI(I)=THETAS-0.87547
    GOTO 4
                END IF
                IF (THETAS.GT.-2.117.AND.THETAS.LE.1.041) ZN=31
                IF (THETAS.GT.1.041.AND.THETAS.LE.2.515) ZN=32
                IF (THETAS.GT.2.515.AND.THETAS.LE.3.861) ZN=33
                IF (THETAS.GT.3.861) THEN
WRITE(6,*) 'Enthalpy out of range '
    GOTO 4
                END IF
END IF

                IF (BP(I).EQ.3) THEN
IF (THETAS.LE.-2.137) THEN
    CHI(I)=THETAS-0.87547
    GOTO 4
                END IF
                IF (THETAS.GT.-2.137.AND.THETAS.LE.2.369) ZN=34
                IF (THETAS.GT.2.369.AND.THETAS.LE.2.596) ZN=35
                IF (THETAS.GT.2.596.AND.THETAS.LE.3.764) ZN=36
                IF (THETAS.GT.3.764) THEN
WRITE(6,*) 'Enthalpy out of range '
    GOTO 4

```



```

                                END IF
END IF

                                DO 5 J=1,6
A(J)=HIDATANEW(ZN,J)
5      CONTINUE

                                CHI(I)=A(1)*THETAS**5+A(2)*THETAS**4+A(3)*THETAS**3 &
                                +A(4)*THETAS**2+A(5)*THETAS+A(6)

4      CONTINUE

3      CONTINUE

                                CHIS=(BPS-BP(1))/(BP(2)-BP(1))*(CHI(2)-CHI(1))+CHI(1)

                                T=10000.0*EXP(CHIS)

RETURN
END SUBROUTINE TEMPNEW

```

!

!

!

!

U.S. STANDARD ATMOSPHERE 1976

```

!
SUBROUTINE ATMOS76(ALT,RESG,RESR,REST,RESP,RESCS)
  IMPLICIT NONE

  DOUBLE PRECISION ALT, RESG, RESR, REST, RESP, RESCS

  INTEGER I

    DOUBLE PRECISION KTSL, NSL, MUSL, M0, MSL, MFPSL, MCFSL
  DOUBLE PRECISION RESM, RESN, AVO, UGAS, PI, G0, G0P, P0, R0,
    T0
  DOUBLE PRECISION SUTH, BETA, GAMMA, ZSL, TSL, PSL, RSL
  DOUBLE PRECISION GSL, HPSL, CSSL, SIGMA, Z, RESH, RESTM,
    RESHP, RESMU
  DOUBLE PRECISION RESKT, RESMFP, RESMCF, TEMPM, WEIGHT,
    TNDENS, PRESS
  DOUBLE PRECISION THERM, VISC, FR, FPATH, SOUND, PSCALE,
    GRAVITY
  DOUBLE PRECISION DENSITY, GPALT

  DOUBLE PRECISION FI, HB, TMB, PB, MI, LMB

  DIMENSION FI(10), HB(8), TMB(7), PB(7), MI(10), LMB(10)

!  CONSTANTS  ———  CATEGORY I

  DATA AVO,UGAS,PI / 6.022169E+26, 8.31432E+03,
    3.141592654 /

```

```

DATA (MI(I),I=1,10) / 28.0134, 31.9988, 39.948,
44.00995, 20.183, &
4.0026, 83.80, 131.30, 16.04303,
2.01594 /

```

! CONSTANTS ——— CATEGORY II

```

DATA G0, G0P, P0, R0, T0, SUTH, BETA, GAMMA, SIGMA    &
/ 9.80665, 9.80665, 1.01325E+05, 6356766.0, 288.15 ,
110.4, &
1.458E-06, 1.40, 3.65E-10 /
DATA (FI(I),I=1,10) / 0.78084, 0.209476, 9.34E-03, 3.14
E-04, &
1.818E-05, 5.24E-06, 1.14E-06,
8.7E-08, &
2.0E-06, 5.0E-07 /
DATA (HB(I),I=1,8) / 0.0, 11.0, 20.0, 32.0, 47.0,
51.0, 71.0, 84.8520 /
DATA (LMB(I),I=1,7) / -6.5, 0.0, 1.0, 2.8, 0.0, -2.8,
-2.0 /

```

! COMPUTE SEALEVEL VALUES

```

ZSL    = 0.0
MSL    = WEIGHT(FI,MI)
M0     = MSL
CALL TENDS(HB,LMB,T0,TMB)
CALL PENDS(G0P,HB,LMB,M0,P0,PB,TMB,UGAS)

```

```

TSL    = TEMPM(G0,G0P,HB,LMB,R0,TMB,ZSL)
PSL    = PRESS(G0,G0P,HB,LMB,M0,PB,R0,TMB,UGAS,ZSL)
RSL    = DENSITY(MSL,PSL,TSL,UGAS,G0,G0P,R0,ZSL)
GSL    = GRAVITY(G0,G0P,R0,ZSL)
HPSL   = PSCALE(GSL,MSL,TSL,UGAS)
CSSL   = SOUND(GAMMA,MSL,TSL,UGAS)
MUSL   = VISC(BETA,SUTH,TSL)
KTSL   = THERM(TSL)
NSL    = TNDENS(AVO,PSL,TSL,UGAS)
MFPSL  = FPATH(NSL,PI,SIGMA)
MCFSL  = FR(AVO,MSL,NSL,PSL,PI,SIGMA,TSL,UGAS)

```

! COMPUTE PHYSICAL QUANTITIES FOR THE GIVEN ALTITUDE

```

Z=ALT
RESH   = GPALT(G0,G0P,R0,Z)
RESM   = M0
RESTM  = TEMPM(G0,G0P,HB,LMB,R0,TMB,Z)
REST   = RESTM
RESP   = PRESS(G0,G0P,HB,LMB,M0,PB,R0,TMB,UGAS,Z)
RESR   = DENSITY(RESM,RESP,REST,UGAS,G0,G0P,R0,Z)
RESG   = GRAVITY(G0,G0P,R0,Z)
RESHP  = PSCALE(RESG,RESM,REST,UGAS)
RESCS  = SOUND(GAMMA,RESM,REST,UGAS)
RESMU  = VISC(BETA,SUTH,REST)
RESKT  = THERM(REST)
RESN   = TNDENS(AVO,RESP,REST,UGAS)
RESMFP = FPATH(RESN,PI,SIGMA)

```

RESMCF = FR(AVO,RESM,RESN,RESP,PI,SIGMA,REST,UGAS)

RETURN

END SUBROUTINE ATMOS76

!

FUNCTION GPALT(G0,G0P,R0,Z)

IMPLICIT NONE

DOUBLE PRECISION GPALT, G0, G0P, R0, Z

DOUBLE PRECISION GAMMA

GAMMA = G0/G0P

GPALT = GAMMA*R0*Z/(R0+Z)

RETURN

END FUNCTION GPALT

!

FUNCTION WEIGHT(FI,MI)

IMPLICIT NONE

DOUBLE PRECISION WEIGHT, FI, MI

DIMENSION FI(10), MI(10)

```

INTEGER I
DOUBLE PRECISION SUM1, SUM2

SUM1 = 0.0
SUM2 = 0.0

DO 10 I = 1,10
    SUM1 = SUM1+FI(I)
    SUM2 = SUM2+MI(I)*FI(I)
10 CONTINUE
    WEIGHT = SUM2/SUM1

RETURN
END FUNCTION WEIGHT

```

```

SUBROUTINE TENDS(HB,LMB,T0,TMB)
IMPLICIT NONE

DOUBLE PRECISION HB, LMB, T0, TMB
DIMENSION LMB(7) , HB(8) , TMB(7)

```

```

INTEGER I

TMB(1) = T0
DO 10 I = 2,7

```

```

        TMB(I) = TMB(I-1) + LMB(I-1)*(HB(I)-HB(I-1))
10  CONTINUE

RETURN
END SUBROUTINE TENDS
!_____

SUBROUTINE PENDS(G0P,HB,LMB,M0,P0,PB,TMB,UGAS)
    IMPLICIT NONE

    DOUBLE PRECISION G0P, HB, LMB, M0, P0, PB, TMB, UGAS
    DIMENSION HB(8), LMB(7), PB(7), TMB(7)

    INTEGER I
    DOUBLE PRECISION AA, BB

    PB(1) = P0
    DO 10 I = 2,7
        IF (LMB(I-1) .NE. 0.0) THEN
            AA = TMB(I-1)/TMB(I)
            BB = G0P*M0/(1.E-03*UGAS*LMB(I-1))
            PB(I) = PB(I-1)*(AA**BB)
        ELSE
            AA = 1.0E+03*G0P*M0*(HB(I)-HB(I-1))/(UGAS*TMB(I
            -1))
            PB(I) = PB(I-1)*EXP(-AA)
        ENDIF
    
```

10 CONTINUE

RETURN

END SUBROUTINE PENDS

!

FUNCTION TEMPM(G0,G0P,HB,LMB,R0,TMB,Z)

IMPLICIT NONE

DOUBLE PRECISION TEMPM, G0, G0P, HB, LMB, R0, TMB, Z

DIMENSION HB(8), LMB(7), TMB(7)

DOUBLE PRECISION GAMMA, HKM

GAMMA = G0/G0P

HKM = 1.E-03*GAMMA*R0*Z/(R0+Z)

IF (HKM .LT. HB(2)) THEN

TEMPM = TMB(1) + LMB(1)*(HKM-HB(1))

ELSE IF ((HKM .GE. HB(2)) .AND. (HKM .LT. HB(3))) THEN

TEMPM = TMB(2) + LMB(2)*(HKM-HB(2))

ELSE IF ((HKM .GE. HB(3)) .AND. (HKM .LT. HB(4))) THEN

TEMPM = TMB(3) + LMB(3)*(HKM-HB(3))

ELSE IF ((HKM .GE. HB(4)) .AND. (HKM .LT. HB(5))) THEN

TEMPM = TMB(4) + LMB(4)*(HKM-HB(4))

ELSE IF ((HKM .GE. HB(5)) .AND. (HKM .LT. HB(6))) THEN


```

      TEMPM = TMB(5) + LMB(5)*(HKM-HB(5))
ELSE IF ((HKM .GE. HB(6)) .AND. (HKM .LT. HB(7))) THEN
      TEMPM = TMB(6) + LMB(6)*(HKM-HB(6))
ELSE IF ((HKM .GE. HB(7)) .AND. (HKM .LE. HB(8))) THEN
      TEMPM = TMB(7) + LMB(7)*(HKM-HB(7))
ENDIF

```

```

IF (HKM.GT.84.852.AND.HKM.LT.92.630) TEMPM=187.
IF (HKM.GT.92.63.AND.HKM.LT.97.482)TEMPM=(HKM-92.63)
    *1.1418+187.74
IF (HKM.GT.97.482.AND.HKM.LT.101.358)TEMPM=(HKM-97.482)
    *2.31+193.3
IF (HKM.GT.101.358.AND.HKM.LT.105.229)TEMPM=(HKM-101.36)
    *3.98+202.2
IF (HKM.GT.105.229.AND.HKM.LT.108.129)TEMPM=(HKM-105.23)
    *7.71+217.6
IF (HKM.GT.108.129)TEMPM=(HKM-108.129)*12.31+240.

```

```

RETURN

```

```

END FUNCTION TEMPM

```

```

!
```

```

FUNCTION PRESS(G0,G0P,HB,LMB,M0,PB,R0,TMB,UGAS,Z)

```

```

IMPLICIT NONE

```

```

DOUBLE PRECISION PRESS, G0, G0P, HB, LMB, M0, PB, R0,
    TMB, UGAS, Z

```

DIMENSION HB(8) , LMB(7) , PB(7) , TMB(7)

DOUBLE PRECISION GAMMA, HKM, AA, BB

GAMMA = G0/G0P

HKM = 1.E-03*GAMMA*R0*Z/(R0+Z)

IF (HKM .LT. HB(2)) THEN

IF (LMB(1) .NE. 0.0) THEN

AA = TMB(1)/(TMB(1)+LMB(1)*(HKM-HB(1)))

BB = G0P*M0/(1.E-03*UGAS*LMB(1))

PRESS = PB(1)*(AA**BB)

ELSE

AA = 1.E+03*G0P*M0*(HKM-HB(1))/(UGAS*TMB(1))

PRESS = PB(1)*EXP(-AA)

ENDIF

ELSE IF ((HKM .GE. HB(2)) .AND. (HKM .LT. HB(3))) THEN

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

AA = TMB(1)/(TMB(2)+LMB(1)*(HKM-HB(2)))

BB = G0P*M0/(1.E-03*UGAS*LMB(2))

PRESS = PB(2)*(AA**BB)

ELSE

AA = 1.E+03*G0P*M0*(HKM-HB(2))/(UGAS*TMB(2))

PRESS = PB(2)*EXP(-AA)

ENDIF

ELSE IF ((HKM .GE. HB(3)) .AND. (HKM .LT. HB(4))) THEN

IF (LMB(3) .NE. 0.0) THEN

```

      AA = TMB(3) / (TMB(3) + LMB(3) * (HKM - HB(3)))
      BB = G0P * M0 / (1.E-03 * UGAS * LMB(3))
      PRESS = PB(3) * (AA * BB)
ELSE
      AA = 1.E+03 * G0P * M0 * (HKM - HB(3)) / (UGAS * TMB(3))
      PRESS = PB(3) * EXP(-AA)
ENDIF
ELSE IF ((HKM .GE. HB(4)) .AND. (HKM .LT. HB(5))) THEN
  IF (LMB(4) .NE. 0.0) THEN
    AA = TMB(4) / (TMB(4) + LMB(4) * (HKM - HB(4)))
    BB = G0P * M0 / (1.E-03 * UGAS * LMB(4))
    PRESS = PB(4) * (AA * BB)
  ELSE
    AA = 1.E+03 * G0P * M0 * (HKM - HB(4)) / (UGAS * TMB(4))
    PRESS = PB(4) * EXP(-AA)
  ENDIF
ELSE IF ((HKM .GE. HB(5)) .AND. (HKM .LT. HB(6))) THEN
  IF (LMB(5) .NE. 0.0) THEN
    AA = TMB(5) / (TMB(5) + LMB(5) * (HKM - HB(5)))
    BB = G0P * M0 / (1.E-03 * UGAS * LMB(5))
    PRESS = PB(5) * (AA * BB)
  ELSE
    AA = 1.E+03 * G0P * M0 * (HKM - HB(5)) / (UGAS * TMB(5))
    PRESS = PB(5) * EXP(-AA)
  ENDIF
ELSE IF ((HKM .GE. HB(6)) .AND. (HKM .LT. HB(7))) THEN
  IF (LMB(6) .NE. 0.0) THEN
    AA = TMB(6) / (TMB(6) + LMB(6) * (HKM - HB(6)))

```

```

        BB = G0P*M0/(1.E-03*UGAS*LMB(6))
        PRESS = PB(6)*(AA**BB)
    ELSE
        AA = 1.E+03*G0P*M0*(HKM-HB(6))/(UGAS*TMB(6))
        PRESS = PB(6)*EXP(-AA)
    ENDIF
ELSE IF ((HKM .GE. HB(7)) .AND. (HKM .LE. HB(8))) THEN
    IF (LMB(7) .NE. 0.0) THEN
        AA = TMB(7)/(TMB(7)+LMB(7)*(HKM-HB(7)))
        BB = G0P*M0/(1.E-03*UGAS*LMB(7))
        PRESS = PB(7)*(AA**BB)
    ELSE
        AA = 1.E+03*G0P*M0*(HKM-HB(7))/(UGAS*TMB(7))
        PRESS = PB(7)*EXP(-AA)
    ENDIF
ELSE IF (HKM.GT.HB(8)) THEN
    PRESS=100.
ENDIF

RETURN
END FUNCTION PRESS
!_____

```

```

FUNCTION DENSITY(M,P,T,UGAS,G0,G0P,R0,Z)
    IMPLICIT NONE

```

```

    DOUBLE PRECISION M, P, T, UGAS, G0, G0P, R0, Z

```

DOUBLE PRECISION DENSITY, HKM, TEMP

DENSITY = P*M/(UGAS*T)

HKM = 0.001*G0*Z*R0/(G0P*(R0 + Z))

IF ((HKM .GT. 84.852) .AND. (HKM .LT. 112.957)) THEN

TEMP = -0.0773 * (HKM - 84.852)

DENSITY = 6.958E-6 * 10**TEMP

ENDIF

IF (HKM .GT. 112.957) THEN

TEMP = -0.0531 * (HKM - 112.957)

DENSITY = 4.289E-8 * 10**TEMP

ENDIF

IF (HKM .GT. 122.0) DENSITY = 0.0

RETURN

END FUNCTION DENSITY

!

FUNCTION GRAVITY(G0,G0P,R0,Z)

IMPLICIT NONE

DOUBLE PRECISION GRAVITY, G0, G0P, R0, Z

$$\text{GRAVITY} = G0 * ((R0 / (R0 + Z)) ** 2)$$

RETURN

END FUNCTION GRAVITY

!

FUNCTION VISC(BETA,SUTH,T)

IMPLICIT NONE

DOUBLE PRECISION VISC, BETA, SUTH, T

$$\text{VISC} = \text{BETA} * (T ** 1.5) / (T + \text{SUTH})$$

RETURN

END FUNCTION VISC

!

FUNCTION THERM(T)

IMPLICIT NONE

DOUBLE PRECISION THERM, T

$$\text{THERM} = 2.64638\text{E}-03 * (T ** 1.5) / (T + 245.4 * (10.0 ** (-12 / T)))$$

RETURN

END FUNCTION THERM

!

FUNCTION SOUND(GAMMA,M,T,UGAS)

IMPLICIT NONE

DOUBLE PRECISION SOUND, GAMMA, M, T, UGAS

SOUND = SQRT(GAMMA*UGAS*T/M)

RETURN

END FUNCTION SOUND

!

FUNCTION TNDENS(AVO,P,T,UGAS)

IMPLICIT NONE

DOUBLE PRECISION TNDENS, AVO, P, T, UGAS

TNDENS = AVO*P/(UGAS*T)

RETURN

END FUNCTION TNDENS

!

```
FUNCTION FPATH(N,PI,SIGMA)
```

```
  IMPLICIT NONE
```

```
  DOUBLE PRECISION FPATH, N, PI, SIGMA
```

```
  FPATH = SQRT(2.0) / (2.*PI*(SIGMA**2.0)*N)
```

```
  RETURN
```

```
END FUNCTION FPATH
```

```
!
```

```
FUNCTION FR(AVO,M,N,P,PI,SIGMA,T,UGAS)
```

```
  IMPLICIT NONE
```

```
  DOUBLE PRECISION FR, AVO, M, N, P, PI, SIGMA, T, UGAS
```

```
  FR = 4.0*AVO*(SIGMA**2.0)*SQRT(PI*(P**2.0)/(UGAS*T*M))
```

```
  RETURN
```

```
END FUNCTION FR
```

```
!
```

```
SUBROUTINE COMPOSIT(FI,N,NI)
```

```
  IMPLICIT NONE
```

```
  DOUBLE PRECISION FI, N, NI
```



```

        DIMENSION FI(10) , NI(10)

        INTEGER I

        DO 10 I = 1,10
            NI(I) = FI(I)*N
        10 CONTINUE

        RETURN
    END SUBROUTINE COMPOSIT
!_____

    FUNCTION PSCALE(G,M,T,UGAS)
        IMPLICIT NONE

        DOUBLE PRECISION PSCALE, G, M, T, UGAS

        PSCALE = UGAS*T/(G*M)

        RETURN
    END FUNCTION PSCALE

!*****

! DVERK

```

```

!      This program is a first order differential equation
      solver. A more
!      thorough description is given below.

```

```

!*****

```

```

SUBROUTINE DVERK (n, fcn, x, y, xend, tol, ind, c, nw, w)

```

```

  IMPLICIT NONE

```

```

      INTEGER n, ind, nw, k

```

```

      DOUBLE PRECISION x, y(n), xend, tol, c(1), w(nw,9),

```

```

         temp

```

```

!*****

```

```

!

```

```

      *

```

```

! note added 11/14/85.

```

```

                                         *

```

```

!

```

```

      *

```

```

! if you discover any errors in this subroutine, please

```

```

  contact      *

```

```

!

```

```

      *

```

```

!      kenneth r. jackson
                                           *
!      department of computer science
                                           *
!      university of toronto
                                           *
!      toronto , ontario ,
                                           *
!      canada    m5s 1a4
                                           *
!
*
!      phone: 416-978-7075
                                           *
!
*
!      electronic mail:
                                           *
!      uucp:    { cornell , decvax , ihnp4 , linus , uw-beaver }!
utcsri!krj      *
!      csnet:   krj@toronto
                                           *
!      arpa:    krj.toronto@csnet-relay
                                           *
!      bitnet:  krj%toronto@csnet-relay.arpa
                                           *

```

```

!

*
! dverk is written in fortran 66.

*

!

*
! the constants dwarf and rreb — c(10) and c(11),
    respectively — are *
! set for a vax in double precision. they should be
    reset, as *
! described below, if this program is run on another machine.

*

!

*
! the c array is declared in this subroutine to have one
    element only, *
! although more elements are referenced in this
    subroutine. this *
! causes some compilers to issue warning messages. there is,
    though, *
! no error provided c is declared sufficiently large in
    the calling *
! program, as described below.

*

```

```

!

*
! the following external statement for fcn was added to
    avoid a *
! warning message from the unix f77 compiler. the
    original dverk *
! comments and code follow it.

*

!

*
!*****

EXTERNAL FCN1, MAX
DOUBLE PRECISION MAX

!*****

!

*
! purpose — this is a runge-kutta subroutine based on
    verner's *
! fifth and sixth order pair of formulas for finding
    approximations to *

```

! the solution of a system of first order ordinary
differential *

! equations with initial conditions. it attempts to keep
the global *

! error proportional to a tolerance specified by the
user. (the *

! proportionality depends on the kind of error control that
is used, *

! as well as the differential equation and the range of
integration.) *

!

*

! various options are available to the user, including
different *

! kinds of error control, restrictions on step sizes, and
interrupts *

! which permit the user to examine the state of the
calculation (and *

! perhaps make modifications) during intermediate stages.

*

!

*

! the program is efficient for non-stiff systems.
however, a good *

! variable-order-adams method will probably be more
efficient if the *

```

! function evaluations are very costly.  such a method would
    also be *
! more suitable if one wanted to obtain a large number of
    intermediate *
! solution values by interpolation , as might be the case for
    example *
! with graphical output.

```

*

```
!
```

```

    *

```

```
!                                     hull-enright-jackson
```

```

    1/10/76      *

```

```
!
```

```

    *

```

```
!*****
```

```
!
```

```

    *

```

```
!      use — the user must specify each of the following
```

```

    *

```

```
!
```

```

    *

```

```
!      n  number of equations
```

*

```

!
*
!   fcn  name of subroutine for evaluating functions – the
subroutine *
!           itself must also be provided by the user – it
should be of *
!           the following form
*
!           subroutine fcn(n, x, y, yprime)
*
!           integer n
*
!           double precision x, y(n), yprime(n)
*
!           *** etc ***
*
!           and it should evaluate yprime, given n, x and y
*
!
*
!   x  independent variable – initial value supplied by
user      *
!
*

```



```

!      y  dependent variable — initial values of components y
      (1), y(2), *
!
      ..., y(n) supplied by user
                        *
!
*
!  xend  value of x to which integration is to be carried out
      — it may *
!
      be less than the initial value of x
                        *
!
*
!  tol  tolerance — the subroutine attempts to control a
norm of  the *
!
      local  error  in  such  a  way  that  the  global
error is *
!
      proportional to tol. in some problems there will
be enough *
!
      damping  of  errors , as well as some cancellation
, so that *
!
      the global error will be less than tol.
alternatively , the *
!
      control  can  be  viewed  as  attempting  to
provide a *
!
      calculated value of y at xend which is the exact
solution *

```

```

!           to the problem  $y' = f(x,y) + e(x)$  where the
norm of  $e(x)$  *
!           is proportional to  $\text{tol}$ . (the norm is a max
norm with *
!           weights that depend on the error control
strategy chosen *
!           by the user. the default weight for the  $k$ -th
component is *
!            $1/\max(1, \text{abs}(y(k)))$ , which therefore provides a
mixture of *
!           absolute and relative error control.)
*
!
*
!   ind indicator – on initial entry ind must be set equal
to either *
!           1 or 2. if the user does not wish to use any
options, he *
!           should set ind to 1 – all that remains for the
user to do *
!           then is to declare  $c$  and  $w$ , and to specify  $nw$ .
the user *
!           may also select various options on initial
entry by *
!           setting  $\text{ind} = 2$  and initializing the first 9
components of *

```

```

!           c as described in the next section.  he may also
      re-enter *
!           the subroutine with ind = 3 as mentioned again
      below. in *
!           any event, the subroutine returns with ind equal
      to      *
!           3 after a normal return
                        *
!           4, 5, or 6 after an interrupt (see options c
      (8), c(9)) *
!           -1, -2, or -3 after an error condition (see
      below)      *
!
      *
!           c communications vector - the dimension must be
      greater than or *
!           equal to 24, unless option c(1) = 4 or 5 is used,
      in which *
!           case the dimension must be greater than or equal
      to n+30      *
!
      *
!           nw first dimension of workspace w - must be greater
      than or *
!           equal to n

```

*

```

!
*
!      w  workspace matrix — first dimension must be nw and
second must *
!              be greater than or equal to 9
*
!
*
!      the subroutine will normally return with ind =
3, having *
! replaced the initial values of x and y with, respectively,
the value *
! of xend and an approximation to y at xend. the subroutine
can be *
! called repeatedly with new values of xend without having
to change *
! any other argument. however, changes in tol, or any of the
options *
! described below, may also be made on such a re-entry if
desired.      *
!
*
!      three error returns are also possible, in which case
x and y *

```

```

! will be the most recently accepted values -
*
! with ind = -3 the subroutine was unable to satisfy
the error *
! requirement with a particular step-size that is
less than or *
! equal to hmin, which may mean that tol is too small
*
! with ind = -2 the value of hmin is greater than
hmax, which *
! probably means that the requested tol (which is
used in the *
! calculation of hmin) is too small
*
! with ind = -1 the allowed maximum number of fcn
evaluations has *
! been exceeded, but this can only occur if option
c(7), as *
! described in the next section, has been used
*
!
*
! there are several circumstances that will cause the
calculations *
! to be terminated, along with output of information that
will help *

```

```

! the user determine the cause of the trouble. these
    circumstances *
! involve entry with illegal or inconsistent values of the
    arguments, *
! such as attempting a normal re-entry without first
    changing the *
! value of xend, or attempting to re-enter with ind less than
    zero.      *
!
    *
!*****
!
    *
!      options - if the subroutine is entered with ind = 1,
    the first 9 *
! components of the communications vector are initialized to
    zero, and *
! the subroutine uses only default values for each option.
    if the *
! subroutine is entered with ind = 2, the user must
    specify each of *
! these 9 components - normally he would first set them all
    to zero, *
! and then make non-zero those that correspond to the
    particular *

```

```

! options he wishes to select. in any event, options may be
    changed on *
! re-entry to the subroutine - but if the user changes
    any of the *
! options, or tol, in the course of a calculation he should
    be careful *
! about how such changes affect the subroutine - it may be
    better to *
! restart with ind = 1 or 2. (components 10 to 24 of c are
    used by the *
! program - the information is available to the user, but
    should not *
! normally be changed by him.)

                                *

!

*

! c(1) error control indicator - the norm of the local
    error is the *
!          max norm of the weighted error estimate
    vector, the *
!          weights being determined according to the value
    of c(1) - *
!          if c(1)=1 the weights are 1 (absolute error
    control) *
!          if c(1)=2 the weights are 1/abs(y(k)) (
    relative error *

```

```

!               control)
!
!               *
!               if c(1)=3 the weights are 1/max(abs(c(2)),
abs(y(k))) *
!               (relative error control, unless abs(y(k)
) is less *
!               than the floor value, abs(c(2)) )
!               *
!               if c(1)=4 the weights are 1/max(abs(c(k+30)),
abs(y(k))) *
!               (here individual floor values are used)
!               *
!               if c(1)=5 the weights are 1/abs(c(k+30))
!               *
!               for all other values of c(1), including c(1)
= 0, the *
!               default values of the weights are
taken to be *
!               1/max(1,abs(y(k))), as mentioned earlier
!               *
!               (in the two cases c(1) = 4 or 5 the user must
declare the *
!               dimension of c to be at least n+30 and must
initialize the *
!               components c(31), c(32), ..., c(n+30).)
!               *

```



```

!

*
! c(2) floor value — used when the indicator c(1) has the
value 3 *
!

*
! c(3) hmin specification — if not zero, the subroutine
chooses hmin *
! to be abs(c(3)) — otherwise it uses the default
value *
! 10*max(dwarf, rreb*max(weighted norm y/tol, abs(
x))), *
! where dwarf is a very small positive machine
number and *
! rreb is the relative roundoff error bound
*
!

*
! c(4) hstart specification — if not zero, the subroutine
will use *
! an initial hmag equal to abs(c(4)), except of
course for *
! the restrictions imposed by hmin and hmax —
otherwise it *

```

```

!           uses the default value of hmax*(tol)**(1/6)
                *
!
                *
! c(5)  scale specification — this is intended to be a
measure of the *
!           scale of the problem — larger values of scale
tend to make *
!           the method more reliable , first  by  possibly
restricting *
!           hmax  (as  described  below) and second , by
tightening the *
!           acceptance requirement — if c(5) is zero , a
default  value *
!           of 1  is  used.  for  linear  homogeneous
problems  with *
!           constant coefficients , an appropriate value for
scale is a *
!           norm  of  the  associated  matrix.  for other
problems , an *
!           approximation to  an  average  value  of  a  norm
of  the *
!           jacobian along the trajectory may be appropriate
                *
!
                *

```



```

! c(8) interrupt number 1 — if not zero, the
subroutine will *
!           interrupt the calculations after it has
chosen its *
!           preliminary value of hmag, and just before
choosing htrial *
!           and xtrial in preparation for taking a step (
htrial may *
!           differ from hmag in sign, and may require
adjustment if *
!           xend is near) — the subroutine returns with ind
= 4, and *
!           will resume calculation at the point of
interruption if *
!           re-entered with ind = 4
*
!
*
! c(9) interrupt number 2 — if not zero, the
subroutine will *
!           interrupt the calculations immediately after
it has *
!           decided whether or not to accept the result of
the most *
!           recent trial step, with ind = 5 if it plans to
accept, or *

```

```

!           ind = 6 if it plans to reject - y(*) is the
           previously *
!           accepted result , while w(*,9) is the newly
           computed trial *
!           value , and w(*,2) is the unweighted error
           estimate vector . *
!           the subroutine will resume calculations at the
           point of *
!           interruption on re-entry with ind = 5 or 6. (the
           user may *
!           change ind in this case if he wishes , for example
           to force *
!           acceptance of a step that would otherwise be
           rejected , or *
!           vice versa . he can also restart with ind = 1 or
           2.)      *
!
           *
!*****
!
           *
! summary of the components of the communications vector
           *

```

```

!
*
!      prescribed at the option      determined by the
program      *
!      of the user
*
!
*
!      c(10) rreb(rel roundoff
err bnd) *
!      c(1) error control indicator  c(11) dwarf (very small
mach no) *
!      c(2) floor value              c(12) weighted norm y
*
!      c(3) hmin specification        c(13) hmin
*
!      c(4) hstart specification      c(14) hmag
*
!      c(5) scale specification       c(15) scale
*
!      c(6) hmax specification        c(16) hmax
*
!      c(7) max no of fcn evals       c(17) xtrial
*
!      c(8) interrupt no 1            c(18) htrial
*

```

```

!      c(9) interrupt no 2          c(19) est
                                     *
!                                     c(20) previous xend
                                     *
!                                     c(21) flag for xend
                                     *
!                                     c(22) no of successful
steps      *
!                                     c(23) no of successive
failures   *
!                                     c(24) no of fcn evals
                                     *
!
*
!  if c(1) = 4 or 5, c(31), c(32), ... c(n+30) are floor
values      *
!
*
!*****
!
!
*
!  an overview of the program
                                     *

```

```

!

*
!      begin initialization , parameter checking , interrupt re-
entries      *
!      .....abort if ind out of range 1 to 6
                                *
!      .      cases - initial entry , normal re-entry , interrupt re-
-entries      *
!      .      case 1 - initial entry (ind .eq. 1 or 2)
                                *
!      v.....abort if n.gt.nw or tol.le.0
                                *
!      .      if initial entry without options (ind .eq. 1)
                                *
!      .      set c(1) to c(9) equal to zero
                                *
!      .      else initial entry with options (ind .eq. 2)
                                *
!      .      make c(1) to c(9) non-negative
                                *
!      .      make floor values non-negative if they are to
be used      *
!      .      end if
                                *
!      .      initialize rreb , dwarf , prev xend , flag , counts
                                *

```



```

!*****error return (with ind=-2) if hmin .gt. hmax
      .  *
!  .      calc preliminary hmag
      .  *
!*****interrupt no 1 (with ind=4) if requested.....re
      -entry.v  *
!  .      calc hmag, xtrial and htrial
      .  *
!  .      end stage 1
      .  *
!  v      stage 2 - calc ytrial (adding 7 to no of fcn evals)
      .  *
!  .      stage 3 - calc the error estimate
      .  *
!  .      stage 4 - make decisions
      .  *
!  .      set ind=5 if step acceptable, else set ind=6
      .  *
!*****interrupt no 2 if requested.....re
      -entry.v  *
!  .      if step accepted (ind .eq. 5)
      .  *
!  .      update x, y from xtrial, ytrial
      .  *
!  .      add 1 to no of successful steps
      .  *
!  .      set no of successive failures to zero
      .  *

```

```

!*****return(with ind=3, xend saved, flag set) if x
    .eq. xend *
!      else step not accepted (ind .eq. 6)
                *
!      add 1 to no of successive failures
                *
!*****error return (with ind=-3) if hmag .le. hmin
                *
!      end if
                                *
!      end stage 4
                                *
!      end loop
                                *
!
                                *
!      begin abort action
                                *
!      output appropriate message about stopping the
calculations , *
!      along with values of ind, n, nw, tol, hmin, hmax,
x, xend, *
!      previous xend, no of successful steps, no of
successive *
!      failures, no of fcn evals, and the components of y
                *

```

```

!      stop

      *
!      end abort action

                                          *

!

      *

!*****

!

*****

!      * begin initialization , parameter checking , interrupt
re-entries *
!

*****

!      .....abort if ind out of range 1 to 6
           if (ind.lt.1 .OR. ind.gt.6) goto 500

!      cases - initial entry , normal re-entry , interrupt re
-reentries
           goto (5, 5, 45, 1111, 2222, 2222), ind
!      case 1 - initial entry (ind .eq. 1 or 2)

```

```

! ..... abort if n.gt.nw or tol.le.0
5      if (n.gt.nw .OR. tol.le.0.d0) goto 500
      if (ind.eq. 2) goto 15
!      initial entry without options (ind .eq. 1)
!      set c(1) to c(9) equal to 0
      do 10 k = 1, 9
          c(k) = 0.d0
10      continue
      goto 35
15      continue
!      initial entry with options (ind .eq. 2)
!      make c(1) to c(9) non-negative
      do 20 k = 1, 9
          c(k) = abs(c(k))
20      continue
!      make floor values non-negative if they are to
be used
      if (c(1).ne.4.d0 .AND. c(1).ne.5.d0) goto 30
      do 25 k = 1, n
          c(k+30) = abs(c(k+30))
25      continue
30      continue
35      continue
!      initialize rreb, dwarf, prev xend, flag, counts
      c(10) = 2.d0**(-56)
      c(11) = 1.d-35
!      set previous xend initially to initial value of x
      c(20) = x

```



```

!      * loop through the following 4 stages , once for each
      trial  step *
!      * until the occurrence of one of the following
              *
!      *      (a) the normal return (with ind .eq. 3) on
      reaching xend in *
!      *          stage 4
              *
!      *      (b) an error return (with ind .lt. 0) in stage 1
      or stage 4 *
!      *      (c) an interrupt return (with ind .eq. 4, 5 or
      6), if *
!      *          requested , in stage 1 or stage 4
              *
!
*****

99999 continue

!
*****

!      * stage 1 – prepare – do calculations of hmin,
      hmax,  etc. , *
!      * and some parameter  checking,  and  end  up  with
      suitable *

```

```

!          * values of hmag, xtrial and htrial in preparation
      for taking *
!          * an integration step.

                                          *

!

*****

!*****error return (with ind=-1) if no of fcn evals too
      great
          if (c(7).eq.0.d0 .OR. c(24).lt.c(7)) goto 100
          ind = -1
          return
100      continue

!          calculate slope (adding 1 to no of fcn evals) if
      ind .ne. 6
          if (ind .eq. 6) goto 105
          call fcn(n, x, y, w(1,1))
          c(24) = c(24) + 1.d0
105      continue

!          calculate hmin - use default unless value
      prescribed
          c(13) = c(3)
          if (c(3) .ne. 0.d0) goto 165
!          calculate default value of hmin

```



```

!           first calculate weighted norm  $y - c(12)$  - as
      specified
!           by the error control indicator  $c(1)$ 
      temp = 0.d0
      if (c(1) .ne. 1.d0) goto 115
!           absolute error control - weights are 1
      do 110 k = 1, n
          temp = max(temp, abs(y(k)))
110      continue
          c(12) = temp
          goto 160
115      if (c(1) .ne. 2.d0) goto 120
!           relative error control - weights are  $1/abs($ 
      y(k)) so
!           weighted norm  $y$  is 1
          c(12) = 1.d0
          goto 160
120      if (c(1) .ne. 3.d0) goto 130
!           weights are  $1/\max(c(2), \text{abs}(y(k)))$ 
      do 125 k = 1, n
          temp = max(temp, abs(y(k))/c(2))
125      continue
          c(12) = min(temp, 1.0d0)
          goto 160
130      if (c(1) .ne. 4.d0) goto 140
!           weights are  $1/\max(c(k+30), \text{abs}(y(k)))$ 
      do 135 k = 1, n
          temp = max(temp, abs(y(k))/c(k+30))

```

```

135          continue
          c(12) = min(temp, 1.0d0)
          goto 160
140      if (c(1) .ne. 5.d0) goto 150
!          weights are 1/c(k+30)
          do 145 k = 1, n
              temp = max(temp, abs(y(k))/c(k+30))
145          continue
          c(12) = temp
          goto 160
150      continue
!          default case – weights are 1/max(1,abs(y(k)
          ))
          do 155 k = 1, n
              temp = max(temp, abs(y(k)))
155          continue
          c(12) = min(temp, 1.0d0)
160      continue
          c(13) = 10.d0*max(c(11),c(10)*max(c(12)/tol,
              abs(x)))
165      continue

!          calculate scale – use default unless value
prescribed
          c(15) = c(5)
          if (c(5) .eq. 0.d0) c(15) = 1.d0

!          calculate hmax – consider 4 cases

```

```

!           case 1 both hmax and scale prescribed
              if (c(6).ne.0.d0 .AND. c(5).ne.0.d0) c(16) =
                  min(c(6), 2.d0/c(5))
!           case 2 - hmax prescribed, but scale not
              if (c(6).ne.0.d0 .AND. c(5).eq.0.d0) c(16) = c
                  (6)
!           case 3 - hmax not prescribed, but scale is
              if (c(6).eq.0.d0 .AND. c(5).ne.0.d0) c(16) =
                  2.d0/c(5)
!           case 4 - neither hmax nor scale is provided
              if (c(6).eq.0.d0 .AND. c(5).eq.0.d0) c(16) =
                  2.d0

!*****error return (with ind=-2) if hmin .gt. hmax
              if (c(13) .le. c(16)) goto 170
                  ind = -2
                  return
170          continue

!           calculate preliminary hmag - consider 3 cases
              if (ind .gt. 2) goto 175
!           case 1 - initial entry - use prescribed value of
hstart, if
!           any, else default
              c(14) = c(4)
              if (c(4) .eq. 0.d0) c(14) = c(16)*tol**(1./6.)
              goto 185
175          if (c(23) .gt. 1.d0) goto 180

```

```

!           case 2 – after a successful step, or at most one
           failure ,
!           use min(2, .9*(tol/est)**(1/6))*hmag, but
           avoid possible
!           overflow. then avoid reduction by more than
           half.

           temp = 2.d0*c(14)
           if (tol .lt. (2.d0/.9d0)**6*c(19)) temp = .9d0
               *(tol/c(19))**(1./6.)*c(14)
           c(14) = max(temp, .5d0*c(14))
           goto 185
180         continue
!           case 3 – after two or more successive failures
           c(14) = .5d0*c(14)
185         continue

!           check against hmax
           c(14) = min(c(14), c(16))

!           check against hmin
           c(14) = max(c(14), c(13))

!*****interrupt no 1 (with ind=4) if requested
           if (c(8) .eq. 0.d0) goto 1111
           ind = 4
           return
!           resume here on re-entry with ind .eq. 4
           .....re-entry..

```

```

1111          continue

!              calculate hmag, xtrial - depending on preliminary
             hmag, xend
                if (c(14) .ge. abs(xend - x)) goto 190
!              do not step more than half way to xend
                c(14) = min(c(14), .5d0*abs(xend - x))
                c(17) = x + sign(c(14), xend - x)
                goto 195
190          continue
!              hit xend exactly
                c(14) = abs(xend - x)
                c(17) = xend
195          continue

!              calculate htrial
                c(18) = c(17) - x

!              end stage 1

!

*****

!              * stage 2 - calculate ytrial (adding 7 to no of fcn
             evals). *
!              * w(*,2), ... w(*,8) hold intermediate results
             needed in *

```



```

210      continue
      call fcn(n, x + c(18)*(2.0/3.0), w(1,9), w(1,4))

      do 215 k = 1, n
        w(k,9) = y(k) + temp*( - w(k,1)*3604654659375.
          d0 &
                                     + w(k,2)
                                     *12816549900000.d0 &
                                     - w(k,3)*9284716546875.
                                     d0 &
                                     + w(k,4)*1237962206250.
                                     d0 )

215      continue
      call fcn(n, x + c(18)*(5.0/6.0), w(1,9), w(1,5))

      do 220 k = 1, n
        w(k,9) = y(k) + temp*(   w(k,1)*3355605792000.
          d0 &
                                     - w(k,2)
                                     *11185352640000.d0 &
                                     + w(k,3)*9172628850000.
                                     d0 &
                                     - w(k,4)*427218330000.
                                     d0 &
                                     + w(k,5)*482505408000.
                                     d0 )

220      continue
      call fcn(n, x + c(18), w(1,9), w(1,6))

```

```

do 225 k = 1, n
    w(k,9) = y(k) + temp*( - w(k,1)*770204740536.
        d0 &
                                + w(k,2)*2311639545600.
                                d0 &
                                - w(k,3)*1322092233000.
                                d0 &
                                - w(k,4)*453006781920.
                                d0 &
                                + w(k,5)*326875481856.
                                d0 )
225    continue
    call fcn(n, x + c(18)/15.0, w(1,9), w(1,7))

do 230 k = 1, n
    w(k,9) = y(k) + temp*(    w(k,1)*2845924389000.
        d0 &
                                - w(k,2)*9754668000000.
                                d0 &
                                + w(k,3)*7897110375000.
                                d0 &
                                - w(k,4)*192082660000.
                                d0 &
                                + w(k,5)*400298976000.
                                d0 &
                                + w(k,7)*201586000000.
                                d0 )

```



```

230      continue
      call fcn(n, x + c(18), w(1,9), w(1,8))

!          calculate ytrial, the extrapolated approximation
      and store
!          in w(*,9)
do 235 k = 1, n
      w(k,9) = y(k) + temp*(      w(k,1)*104862681000.
                                d0 &
                                + w(k,3)*545186250000.
                                d0 &
                                + w(k,4)*446637345000.
                                d0 &
                                + w(k,5)*188806464000.
                                d0 &
                                + w(k,7)*15076875000.d0
                                &
                                + w(k,8)*97599465000.d0
                                )
235      continue

!          add 7 to the no of fcn evals
      c(24) = c(24) + 7.d0

!          end stage 2

```

```

!
*****

!      * stage 3 – calculate the error estimate est. first
      calculate *
!      * the unweighted absolute error estimate vector
      (per unit *
!      * step) for the unextrapolated approximation and
      store it in *
!      * w(*,2). then calculate the weighted max norm of
      w(*,2) as *
!      * specified by the error control indicator c(1).
      finally , *
!      * modify this result to produce est, the error
      estimate (per *
!      * unit step) for the extrapolated approximation
      ytrial.      *
!
*****

!      calculate the unweighted absolute error estimate
      vector

      do 300 k = 1, n
          w(k,2) = (    w(k,1)*8738556750.d0      &
                     + w(k,3)*9735468750.d0      &
                     - w(k,4)*9709507500.d0      &
                     + w(k,5)*8582112000.d0      &

```

```

+ w(k,6)*95329710000.d0      &
- w(k,7)*15076875000.d0      &
- w(k,8)*97599465000.d0)
/1398169080000.d0

300      continue

!          calculate the weighted max norm of w(*,2) as
specified by
!          the error control indicator c(1)
temp = 0.d0
if (c(1) .ne. 1.d0) goto 310
!          absolute error control
do 305 k = 1, n
temp = max(temp, abs(w(k,2)))
305      continue
goto 360
310      if (c(1) .ne. 2.d0) goto 320
!          relative error control
do 315 k = 1, n
temp = max(temp, abs(w(k,2)/y(k)))
315      continue
goto 360
320      if (c(1) .ne. 3.d0) goto 330
!          weights are 1/max(c(2),abs(y(k)))
do 325 k = 1, n
temp = max(temp, abs(w(k,2)) / max(c(2),
abs(y(k))) )
325      continue

```

```

                                goto 360
330      if (c(1) .ne. 4.d0) goto 340
!      weights are 1/max(c(k+30),abs(y(k)))
      do 335 k = 1, n
          temp = max(temp, abs(w(k,2)) / max(c(k+30),
                                abs(y(k)))) )
335      continue
      goto 360
340      if (c(1) .ne. 5.d0) goto 350
!      weights are 1/c(k+30)
      do 345 k = 1, n
          temp = max(temp, abs(w(k,2)/c(k+30)))
345      continue
      goto 360
350      continue
!      default case – weights are 1/max(1,abs(y(k)))
      do 355 k = 1, n
          temp = max(temp, abs(w(k,2)) / max(1.0, abs
                                (y(k)))) )
355      continue
360      continue

!      calculate est – (the weighted max norm of w(*,2))
      *hmag*scale
!      – est is intended to be a measure of the error
      per unit
!      step in ytrial
      c(19) = temp*c(14)*c(15)

```

```

!           end stage 3

!
*****

!           * stage 4 – make decisions.

                                           *
!
*****

!           set ind=5 if step acceptable , else set ind=6
           ind = 5
           if (c(19) .gt. tol) ind = 6

!*****interrupt no 2 if requested
           if (c(9) .eq. 0.d0) goto 2222
           return
!           resume here on re-entry with ind .eq. 5 or 6
           ...re-entry..
2222           continue

           if (ind .eq. 6) goto 410
!           step accepted (ind .eq. 5), so update x, y
           from xtrial ,
!
           ytrial , add 1 to the no of successful steps
           , and set

```

```

!               the no of successive failures to zero
x = c(17)
do 400 k = 1, n
    y(k) = w(k,9)
400    continue
    c(22) = c(22) + 1.d0
    c(23) = 0.d0
!*****return(with ind=3, xend saved, flag set) if x
    .eq. xend
        if (x .ne. xend) goto 405
        ind = 3
        c(20) = xend
        c(21) = 1.d0
        return
405    continue
        goto 420
410    continue
!               step not accepted (ind .eq. 6), so add 1 to
    the no of
!               successive failures
    c(23) = c(23) + 1.d0
!*****error return (with ind=-3) if hmag .le. hmin
    if (c(14) .gt. c(13)) goto 415
        ind = -3
        return
415    continue
420    continue

```

```

!           end stage 4

           goto 99999
!           end loop

!   begin abort action
500 continue

!           write(6,505) ind, tol, x, n, c(13), xend, nw, c(16), c
(20), &
!           c(22), c(23), c(24), (y(k), k = 1, n)
!   505 format( /// 1h0, 58hcomputation stopped in dverk with
the following values - / 1h0,
!   5hind =, i4, 5x, 6htol  =, 1pd13.6, 5x, 11hx          =,
&
!           1pd22.15          &
!           / 1h , 5hn  =, i4, 5x, 6hhmin =, 1pd13.6, 5x, 11
hxend          =, &
!           1pd22.15          &
!           / 1h , 5hnw  =, i4, 5x, 6hhmax =, 1pd13.6, 5x, 11
hprev xend =, &
!           1pd22.15          &
!           / 1h0, 14x, 27hno of successful steps      =, 0pf8.0
&
!           / 1h , 14x, 27hno of successive failures =, 0pf8.0
&
!           / 1h , 14x, 27hno of function evals        =, 0pf8.0
&

```

```

!      / 1h0, 23hthe components of y are      &
!      // (1h , 1p5d24.15)

```

```

RETURN

```

```

!  end abort action

```

```

END SUBROUTINE DVERK

```

```

!      ALGORITHM 474 COLLECTED ALGORITHMS FROM ACM.
!      ALGORITHM APPEARED IN COMM. ACM, VOL. 17, NO. 1,
!      P. 26.

```

```

SUBROUTINE ITPLBV(IU, LX, LY, X, Y, Z, N, U, V, W)
  IMPLICIT NONE

```

```

! BIVARIATE INTERPOLATION
! THIS SUBROUTINE INTERPOLATES, FROM VALUES OF THE FUNCTION
! GIVEN AT INPUT GRID POINTS IN AN X-Y PLANE AND FOR A GIVEN
! SET OF POINTS IN THE PLANE, THE VALUES OF A SINGLE-VALUED
! BIVARIATE FUNCTION  $Z = Z(X,Y)$ .
! THE METHOD IS BASED ON A PIECE-WISE FUNCTION COMPOSED OF
! A SET OF BICUBIC POLYNOMIALS IN X AND Y.  EACH POLYNOMIAL
! IS APPLICABLE TO A RECTANGLE OF THE INPUT GRID IN THE X-Y
! PLANE.  EACH POLYNOMIAL IS DETERMINED LOCALLY.
! THE INPUT PARAMETERS ARE
! IU  = LOGICAL UNIT NUMBER OF STANDARD OUTPUT UNIT

```


! LX = NUMBER OF INPUT GRID POINTS IN THE X COORDINATE
 ! (MUST BE 2 OR GREATER)
 ! LY = NUMBER OF INPUT GRID POINTS IN THE Y COORDINATE
 ! (MUST BE 2 OR GREATER)
 ! X = ARRAY OF DIMENSION LX STORING THE X COORDINATES
 ! OF INPUT GRID POINTS (IN ASCENDING ORDER)
 ! Y = ARRAY OF DIMENSION LY STORING THE Y COORDINATES
 ! OF INPUT GRID POINTS (IN ASCENDING ORDER)
 ! Z = DOUBLY-DIMENSIONED ARRAY OF DIMENSION (LX,LY)
 ! STORING THE VALUES OF THE FUNCTION (Z VALUES)
 ! AT INPUT GRID POINTS
 ! N = NUMBER OF POINTS AT WHICH INTERPOLATION OF THE
 ! Z VALUE IS DESIRED (MUST BE 1 OR GREATER)
 ! U = ARRAY OF DIMENSION N STORING THE X COORDINATES
 ! OF DESIRED POINTS
 ! V = ARRAY OF DIMENSION N STORING THE Y COORDINATES
 ! OF DESIRED POINTS
 ! THE OUTPUT PARAMETER IS
 ! W = ARRAY OF DIMENSION N WHERE THE INTERPOLATED Z
 ! VALUES AT DESIRED POINTS ARE TO BE DISPLAYED
 ! SOME VARIABLES INTERNALLY USED ARE
 ! ZA = DIVIDED DIFFERENCE OF Z WITH RESPECT TO X
 ! ZB = DIVIDED DIFFERENCE OF Z WITH RESPECT TO Y
 ! ZAB = SECOND ORDER DIVIDED DIFFERENCE OF Z WITH
 ! RESPECT TO X AND Y
 ! ZX = PARTIAL DERIVATIVE OF Z WITH RESPECT TO X
 ! ZY = PARTIAL DERIVATIVE OF Z WITH RESPECT TO Y
 ! ZXY = SECOND ORDER PARTIAL DERIVATIVE OF Z WITH

! RESPECT TO X AND Y

! DECLARATION STATEMENTS

INTEGER IU , LX, LY, N

DOUBLE PRECISION X, Y, Z, U, V, W

INTEGER IU0 , LX0, LXM1, LXM2, LXP1, LY0, LYM1, LYM2, LYP1,
N0

INTEGER IX , IY, IXPV, IYPV, K, IMN, IMX, JX, JY, JXM2, JXML
INTEGER JYM2, JYML, JX1, JY1

DOUBLE PRECISION UK, VK, X3, X4, A3, Y3, Y4, B3, Z33, Z43 ,
Z34

DOUBLE PRECISION Z44 , Z3A3, Z4A3, Z3B3, Z4B3, ZA3B3, X2, A2
DOUBLE PRECISION Z23 , Z24, Z3A2, Z4A2, X5, A4, Z53, Z54 ,
Z3A4

DOUBLE PRECISION A1, Z3A1, Z4A1, A5, Z3A5, Z4A5, Y2, B2, Z32
DOUBLE PRECISION Z42 , Z3B2, Z4B2, Y5, B4, Z35, Z45, Z3B4 ,
Z4B4

DOUBLE PRECISION ZA3B2, ZA3B4, B1, Z3B1, Z4B1, B5, Z3B5 ,
Z4B5

DOUBLE PRECISION ZA4B2, ZA4B4, ZA2B2, ZA2B4, W2, ZA, W3, SW
DOUBLE PRECISION WX2, WX3, ZX, ZB, WY2, WY3, ZY, ZXY, ZAB ,
W1

DOUBLE PRECISION W4, W5, Z4A4, ZA2B3, ZA4B3, ZX3B3

DOUBLE PRECISION ZX34, ZX33, ZX4B3, ZX44, ZX43, ZY3A3, ZY43 ,
ZY33

DOUBLE PRECISION ZY4A3, ZY44, ZY34, A, ZXY33, B, ZXY43, C ,
ZXY34

DOUBLE PRECISION D, ZXY44, E, A3SQ, B3SQ, P02, P03, P12, P13
, P20

DOUBLE PRECISION P21, P22, P23, P30, P31, P32, P33, DY, Q0,
P00

DOUBLE PRECISION P01, Q1, P10, P11, Q2, Q3, DX

DIMENSION X(LX), Y(LY), Z(LX,LY), U(N), V(N), W(N)

DIMENSION ZA(5,2), ZB(2,5), ZAB(3,3), ZX(4,4), ZY(4,4), &
ZXY(4,4)

EQUIVALENCE (Z3A1,ZA(1,1)), (Z3A2,ZA(2,1)), (Z3A3,ZA(3,1)),
&

(Z3A4,ZA(4,1)), (Z3A5,ZA(5,1)), (Z4A1,ZA(1,2)), (Z4A2,ZA
(2,2)), &

(Z4A3,ZA(3,2)), (Z4A4,ZA(4,2)), (Z4A5,ZA(5,2)), (Z3B1,ZB
(1,1)), &

(Z3B2,ZB(1,2)), (Z3B3,ZB(1,3)), (Z3B4,ZB(1,4)), (Z3B5,ZB
(1,5)), &

(Z4B1,ZB(2,1)), (Z4B2,ZB(2,2)), (Z4B3,ZB(2,3)), (Z4B4,ZB
(2,4)), &

(Z4B5,ZB(2,5)), (ZA2B2,ZAB(1,1)), (ZA3B2,ZAB(2,1)),

&

(ZA4B2,ZAB(3,1)), (ZA2B3,ZAB(1,2)), (ZA3B3,ZAB(2,2)),

&

(ZA4B3,ZAB(3,2)), (ZA2B4,ZAB(1,3)), (ZA3B4,ZAB(2,3)),

&

(ZA4B4,ZAB(3,3)), (ZX33,ZX(2,2)), (ZX43,ZX(3,2)),

&

$$\begin{aligned}
& (ZX34, ZX(2, 3)) , \quad (ZX44, ZX(3, 3)) , \quad (ZY33, ZY(2, 2)) , \\
& \quad \& \\
& (ZY43, ZY(3, 2)) , \quad (ZY34, ZY(2, 3)) , \quad (ZY44, ZY(3, 3)) , \\
& \quad \& \\
& (ZXY33, ZXY(2, 2)) , \quad (ZXY43, ZXY(3, 2)) , \quad (ZXY34, ZXY(2, 3)) , \\
& \quad \& \\
& (ZXY44, ZXY(3, 3)) , \quad (P00, Z33) , \quad (P01, ZY33) , \quad (P10, ZX33) , \\
& \quad \& \\
& (P11, ZXY33) \\
\text{EQUIVALENCE} \quad & (LX0, ZX(1, 1)) , \quad (LXM1, ZX(4, 1)) , \quad (LXM2, ZX(1, 4)) , \\
& \quad \& \\
& (LXP1, ZX(4, 4)) , \quad (LY0, ZY(1, 1)) , \quad (LYM1, ZY(4, 1)) , \quad (LYM2, ZY \\
& \quad (1, 4)) , \quad \& \\
& (LYP1, ZY(4, 4)) , \quad (IX, ZXY(1, 1)) , \quad (IY, ZXY(4, 1)) , \quad (IXPV, ZXY \\
& \quad (1, 4)) , \quad \& \\
& (IYPV, ZXY(4, 4)) , \quad (IMN, JX) , \quad (IMX, JY) , \quad (JXM2, JX1) , \\
& \quad \& \\
& (JYM2, JY1) , \quad (UK, DX) , \quad (VK, DY) , \quad (A1, A5, B1, B5, ZX(2, 1), A, Q0) , \\
& \quad \& \\
& (A2, ZX(1, 2), B, Q1) , \quad (A4, ZX(4, 2), C, Q2) , \quad (B2, ZY(2, 1), D, Q3) , \\
& \quad \& \\
& (B4, ZY(2, 4), E) , \quad (X2, ZX(3, 1), A3SQ) , \quad (X4, ZX(1, 3)) , \quad (X5, ZX \\
& \quad (4, 3)) , \quad \& \\
& (Y2, ZX(2, 4)) , \quad (Y4, ZY(3, 1), B3SQ) , \quad (Y5, ZX(3, 4), P02) , \\
& \quad \& \\
& (Z23, ZY(1, 2), P03) , \quad (Z24, ZY(4, 2), P12) , \quad (Z32, ZY(1, 3), P13) , \\
& \quad \&
\end{aligned}$$

```

(Z34,ZY(4,3),P20), (Z35,ZY(3,4),P21), (Z42,ZXY(2,1),P22),
&
(Z43,ZXY(1,2),P23), (Z44,ZXY(3,1),P30), (Z45,ZXY(4,2),P31),
&
(Z53,ZXY(1,3),P32), (Z54,ZXY(4,3),P33), (W2,WY2,W4),
&
(W3,WY3,W1,W5), (WX2,ZXY(2,4)), (WX3,ZXY(3,4))

```

! PRELIMINARY PROCESSING

! SETTING OF SOME INPUT PARAMETERS TO LOCAL VARIABLES

```

IU0 = IU
LX0 = LX
LXM1 = LX0 - 1
LXM2 = LXM1 - 1
LXP1 = LX0 + 1
LY0 = LY
LYM1 = LY0 - 1
LYM2 = LYM1 - 1
LYP1 = LY0 + 1
N0 = N

```

! ERROR CHECK

```

IF (LXM2.LT.0) GOTO 710
IF (LYM2.LT.0) GOTO 720
IF (N0.LT.1) GOTO 730

```

```

DO 10 IX=2,LX0
    IF (X(IX-1)-X(IX)) 10, 740, 750

```

```

10 CONTINUE

DO 20 IY=2,LY0
  IF (Y(IY-1)-Y(IY)) 20, 770, 780
20 CONTINUE

! INITIAL SETTING OF PREVIOUS VALUES OF IX AND IY
  IXPV = 0
  IYPV = 0

! MAIN DO-LOOP
DO 700 K=1,N0
  UK = U(K)
  VK = V(K)

! ROUTINES TO LOCATE THE DESIRED POINT
! TO FIND OUT THE IX VALUE FOR WHICH
! (U(K).GE.X(IX-1)).AND.(U(K).LT.X(IX))
  IF (LXM2.EQ.0) GOTO 80
  IF (UK.GE.X(LX0)) GOTO 70
    IF (UK.LT.X(1)) GOTO 60
IMN = 2
  IMX = LX0
30  IX = (IMN+IMX)/2
    IF (UK.GE.X(IX)) GOTO 40
  IMX = IX
  GOTO 50

```

```

40    IMN = IX + 1
50    IF (IMX.GT.IMN) GOTO 30
      IX = IMX
      GOTO 90

```

```

60    IX = 1
      GOTO 90

```

```

70    IX = LXP1
      GOTO 90

```

```

80    IX = 2

```

```

! TO FIND OUT THE IY VALUE FOR WHICH
! (V(K).GE.Y(IY-1)).AND.(V(K).LT.Y(IY))

```

```

90    IF (LYM2.EQ.0) GOTO 150
      IF (VK.GE.Y(LY0)) GOTO 140

```

```

IF (VK.LT.Y(1)) GOTO 130

```

```

IMN = 2

```

```

      IMX = LY0

```

```

100   IY = (IMN+IMX)/2
      IF (VK.GE.Y(IY)) GOTO 110
      IMX = IY
      GOTO 120

```

```

110   IMN = IY + 1

```

```

120   IF (IMX.GT.IMN) GOTO 100
      IY = IMX

```

```

        GOTO 160

130     IY = 1
        GOTO 160

140     IY = LYP1
        GOTO 160

150     IY = 2

! TO CHECK IF THE DESIRED POINT IS IN THE SAME RECTANGLE
! AS THE PREVIOUS POINT.  IF YES, SKIP TO THE COMPUTATION
! OF THE POLYNOMIAL
160     IF (IX.EQ.IXPV .AND. IY.EQ.IYPV) GOTO 690
        IXPV = IX
        IYPV = IY

! ROUTINES TO PICK UP NECESSARY X, Y, AND Z VALUES, TO
! COMPUTE THE ZA, ZB, AND ZAB VALUES, AND TO ESTIMATE THEM
! WHEN NECESSARY
JX = IX
        IF (JX.EQ.1) JX = 2
        IF (JX.EQ.LXP1) JX = LX0
        JY = IY
        IF (JY.EQ.1) JY = 2
        IF (JY.EQ.LYP1) JY = LY0
        JXM2 = JX - 2
        JXML = JX - LX0

```


$$\begin{aligned} \text{JYM2} &= \text{JY} - 2 \\ \text{JYML} &= \text{JY} - \text{LY0} \end{aligned}$$

! IN THE CORE AREA, I.E., IN THE RECTANGLE THAT CONTAINS
! THE DESIRED POINT

$$\begin{aligned} \text{X3} &= \text{X}(\text{JX}-1) \\ \text{X4} &= \text{X}(\text{JX}) \\ \text{A3} &= 1.0 / (\text{X4}-\text{X3}) \\ \text{Y3} &= \text{Y}(\text{JY}-1) \\ \text{Y4} &= \text{Y}(\text{JY}) \\ \text{B3} &= 1.0 / (\text{Y4}-\text{Y3}) \\ \text{Z33} &= \text{Z}(\text{JX}-1, \text{JY}-1) \\ \text{Z43} &= \text{Z}(\text{JX}, \text{JY}-1) \\ \text{Z34} &= \text{Z}(\text{JX}-1, \text{JY}) \\ \text{Z44} &= \text{Z}(\text{JX}, \text{JY}) \\ \text{Z3A3} &= (\text{Z43}-\text{Z33}) * \text{A3} \\ \text{Z4A3} &= (\text{Z44}-\text{Z34}) * \text{A3} \\ \text{Z3B3} &= (\text{Z34}-\text{Z33}) * \text{B3} \\ \text{Z4B3} &= (\text{Z44}-\text{Z43}) * \text{B3} \\ \text{ZA3B3} &= (\text{Z4B3}-\text{Z3B3}) * \text{A3} \end{aligned}$$

! IN THE X DIRECTION

$$\begin{aligned} \text{IF } (\text{LXM2.EQ.0}) &\text{ GOTO 230} \\ \text{IF } (\text{JXM2.EQ.0}) &\text{ GOTO 170} \\ \text{X2} &= \text{X}(\text{JX}-2) \\ \text{A2} &= 1.0 / (\text{X3}-\text{X2}) \\ \text{Z23} &= \text{Z}(\text{JX}-2, \text{JY}-1) \\ \text{Z24} &= \text{Z}(\text{JX}-2, \text{JY}) \end{aligned}$$

```

      Z3A2 = (Z33-Z23)*A2
      Z4A2 = (Z34-Z24)*A2
      IF (JXML.EQ.0) GOTO 180
170    X5 = X(JX+1)
      A4 = 1.0/(X5-X4)
      Z53 = Z(JX+1,JY-1)
      Z54 = Z(JX+1,JY)
      Z3A4 = (Z53-Z43)*A4
      Z4A4 = (Z54-Z44)*A4
      IF (JXM2.NE.0) GOTO 190
      Z3A2 = Z3A3 + Z3A3 - Z3A4
      Z4A2 = Z4A3 + Z4A3 - Z4A4
      GOTO 190

180    Z3A4 = Z3A3 + Z3A3 - Z3A2
      Z4A4 = Z4A3 + Z4A3 - Z4A2
190    ZA2B3 = (Z4A2-Z3A2)*B3
      ZA4B3 = (Z4A4-Z3A4)*B3
      IF (JX.LE.3) GOTO 200
      A1 = 1.0/(X2-X(JX-3))
      Z3A1 = (Z23-Z(JX-3,JY-1))*A1
      Z4A1 = (Z24-Z(JX-3,JY))*A1
      GOTO 210

200    Z3A1 = Z3A2 + Z3A2 - Z3A3
      Z4A1 = Z4A2 + Z4A2 - Z4A3
210    IF (JX.GE.LXM1) GOTO 220
      A5 = 1.0/(X(JX+2)-X5)

```

$$Z3A5 = (Z(JX+2,JY-1)-Z53)*A5$$

$$Z4A5 = (Z(JX+2,JY)-Z54)*A5$$

GOTO 240

$$220 \quad Z3A5 = Z3A4 + Z3A4 - Z3A3$$

$$Z4A5 = Z4A4 + Z4A4 - Z4A3$$

GOTO 240

$$230 \quad Z3A2 = Z3A3$$

$$Z4A2 = Z4A3$$

GOTO 180

! IN THE Y DIRECTION

240 IF (LYM2.EQ.0) GOTO 310

IF (JYM2.EQ.0) GOTO 250

$$Y2 = Y(JY-2)$$

$$B2 = 1.0/(Y3-Y2)$$

$$Z32 = Z(JX-1,JY-2)$$

$$Z42 = Z(JX,JY-2)$$

$$Z3B2 = (Z33-Z32)*B2$$

$$Z4B2 = (Z43-Z42)*B2$$

IF (JYML.EQ.0) GOTO 260

$$250 \quad Y5 = Y(JY+1)$$

$$B4 = 1.0/(Y5-Y4)$$

$$Z35 = Z(JX-1,JY+1)$$

$$Z45 = Z(JX,JY+1)$$

$$Z3B4 = (Z35-Z34)*B4$$

$$Z4B4 = (Z45-Z44)*B4$$

```

IF (JYM2.NE.0) GOTO 270
      Z3B2 = Z3B3 + Z3B3 - Z3B4
      Z4B2 = Z4B3 + Z4B3 - Z4B4
GOTO 270

260   Z3B4 = Z3B3 + Z3B3 - Z3B2
      Z4B4 = Z4B3 + Z4B3 - Z4B2
270   ZA3B2 = (Z4B2-Z3B2)*A3
      ZA3B4 = (Z4B4-Z3B4)*A3
      IF (JY.LE.3) GOTO 280
      B1 = 1.0/(Y2-Y(JY-3))
      Z3B1 = (Z32-Z(JX-1,JY-3))*B1
      Z4B1 = (Z42-Z(JX,JY-3))*B1
      GOTO 290

280   Z3B1 = Z3B2 + Z3B2 - Z3B3
      Z4B1 = Z4B2 + Z4B2 - Z4B3
290   IF (JY.GE.LYM1) GOTO 300
      B5 = 1.0/(Y(JY+2)-Y5)
      Z3B5 = (Z(JX-1,JY+2)-Z35)*B5
      Z4B5 = (Z(JX,JY+2)-Z45)*B5
      GOTO 320

300   Z3B5 = Z3B4 + Z3B4 - Z3B3
      Z4B5 = Z4B4 + Z4B4 - Z4B3
GOTO 320

310   Z3B2 = Z3B3

```

```

      Z4B2 = Z4B3
GOTO 260

! IN THE DIAGONAL DIRECTIONS
320  IF (LXM2.EQ.0) GOTO 400
      IF (LYM2.EQ.0) GOTO 410
      IF (JXML.EQ.0) GOTO 350
      IF (JYM2.EQ.0) GOTO 330
      ZA4B2 = ((Z53-Z(JX+1,JY-2))*B2-Z4B2)*A4
      IF (JYML.EQ.0) GOTO 340
330  ZA4B4 = ((Z(JX+1,JY+1)-Z54)*B4-Z4B4)*A4
      IF (JYM2.NE.0) GOTO 380
      ZA4B2 = ZA4B3 + ZA4B3 - ZA4B4
      GOTO 380

340  ZA4B4 = ZA4B3 + ZA4B3 - ZA4B2
      GOTO 380

350  IF (JYM2.EQ.0) GOTO 360
      ZA2B2 = (Z3B2-(Z23-Z(JX-2,JY-2))*B2)*A2
      IF (JYML.EQ.0) GOTO 370
360  ZA2B4 = (Z3B4-(Z(JX-2,JY+1)-Z24)*B4)*A2
      IF (JYM2.NE.0) GOTO 390
      ZA2B2 = ZA2B3 + ZA2B3 - ZA2B4
      GOTO 390

370  ZA2B4 = ZA2B3 + ZA2B3 - ZA2B2
      GOTO 390

```

```

380   IF (JXM2.NE.0) GOTO 350
      ZA2B2 = ZA3B2 + ZA3B2 - ZA4B2
      ZA2B4 = ZA3B4 + ZA3B4 - ZA4B4
      GOTO 420

```

```

390   IF (JXML.NE.0) GOTO 420
      ZA4B2 = ZA3B2 + ZA3B2 - ZA2B2
      ZA4B4 = ZA3B4 + ZA3B4 - ZA2B4
      GOTO 420

```

```

400   ZA2B2 = ZA3B2
      ZA4B2 = ZA3B2
      ZA2B4 = ZA3B4
      ZA4B4 = ZA3B4
      GOTO 420

```

```

410   ZA2B2 = ZA2B3
      ZA2B4 = ZA2B3
      ZA4B2 = ZA4B3
      ZA4B4 = ZA4B3

```

! NUMERICAL DIFFERENTIATION ——— TO DETERMINE PARTIAL
! DERIVATIVES ZX, ZY, AND ZXY AS WEIGHTED MEANS OF DIVIDED
! DIFFERENCES ZA, ZB, AND ZAB, RESPECTIVELY

```

420   DO 480 JY=2,3
      DO 470 JX=2,3

```

```

W2 = ABS(ZA(JX+2,JY-1)-ZA(JX+1,JY-1))
W3 = ABS(ZA(JX,JY-1)-ZA(JX-1,JY-1))
SW = W2 + W3
IF (SW.EQ.0.0) GOTO 430
WX2 = W2/SW
WX3 = W3/SW
GOTO 440

430    WX2 = 0.5
      WX3 = 0.5
440    ZX(JX,JY) = WX2*ZA(JX,JY-1) + WX3*ZA(JX+1,JY-1)
      W2 = ABS(ZB(JX-1,JY+2)-ZB(JX-1,JY+1))
      W3 = ABS(ZB(JX-1,JY)-ZB(JX-1,JY-1))
      SW = W2 + W3
      IF (SW.EQ.0.0) GOTO 450
      WY2 = W2/SW
      WY3 = W3/SW
      GOTO 460

450    WY2 = 0.5
      WY3 = 0.5
460    ZY(JX,JY) = WY2*ZB(JX-1,JY) + WY3*ZB(JX-1,JY+1)
      ZXY(JX,JY) = WY2*(WX2*ZAB(JX-1,JY-1)+WX3*ZAB(JX,JY-1)) +
      &
      WY3*(WX2*ZAB(JX-1,JY)+WX3*ZAB(JX,JY))

470    CONTINUE
480    CONTINUE

```

```

! WHEN (U(K) .LT. X(1)) .OR. (U(K) .GT. X(LX))
      IF (IX .EQ. LXP1) GOTO 530
      IF (IX .NE. 1) GOTO 590
      W2 = A4*(3.0*A3+A4)
      W1 = 2.0*A3*(A3-A4) + W2

      DO 500 JY=2,3

      ZX(1,JY) = (W1*ZA(1,JY-1)+W2*ZA(2,JY-1))/(W1+W2)
      ZY(1,JY) = ZY(2,JY) + ZY(2,JY) - ZY(3,JY)
      ZXY(1,JY) = ZXY(2,JY) + ZXY(2,JY) - ZXY(3,JY)

      DO 490 JX1=2,3

      JX = 5 - JX1
      ZX(JX,JY) = ZX(JX-1,JY)
      ZY(JX,JY) = ZY(JX-1,JY)
      ZXY(JX,JY) = ZXY(JX-1,JY)

490    CONTINUE
500    CONTINUE

      X3 = X3 - 1.0/A4
      Z33 = Z33 - Z3A2/A4

      DO 510 JY=1,5
      ZB(2,JY) = ZB(1,JY)

```


510 CONTINUE

DO 520 JY=2,4

ZB(1,JY) = ZB(1,JY) - ZAB(1,JY-1)/A4

520 CONTINUE

A3 = A4

JX = 1

GOTO 570

530 W4 = A2*(3.0*A3+A2)

W5 = 2.0*A3*(A3-A2) + W4

DO 550 JY=2,3

ZX(4,JY) = (W4*ZA(4,JY-1)+W5*ZA(5,JY-1))/(W4+W5)

ZY(4,JY) = ZY(3,JY) + ZY(3,JY) - ZY(2,JY)

ZXY(4,JY) = ZXY(3,JY) + ZXY(3,JY) - ZXY(2,JY)

DO 540 JX=2,3

ZX(JX,JY) = ZX(JX+1,JY)

ZY(JX,JY) = ZY(JX+1,JY)

ZXY(JX,JY) = ZXY(JX+1,JY)

540 CONTINUE

550 CONTINUE

```

      X3 = X4
Z33 = Z43

      DO 560 JY=1,5
      ZB(1,JY) = ZB(2,JY)
560    CONTINUE

      A3 = A2
      JX = 3
570    ZA(3,1) = ZA(JX+1,1)

      DO 580 JY=1,3
      ZAB(2,JY) = ZAB(JX,JY)
580    CONTINUE

! WHEN (V(K).LT.Y(1)).OR.(V(K).GT.Y(LY))
590    IF (IY.EQ.LYP1) GOTO 630
      IF (IY.NE.1) GOTO 680
      W2 = B4*(3.0*B3+B4)
      W1 = 2.0*B3*(B3-B4) + W2

      DO 620 JX=2,3

      IF (JX.EQ.3 .AND. IX.EQ.LXP1) GOTO 600
      IF (JX.EQ.2 .AND. IX.EQ.1) GOTO 600
      ZY(JX,1) = (W1*ZB(JX-1,1)+W2*ZB(JX-1,2))/(W1+W2)
      ZX(JX,1) = ZX(JX,2) + ZX(JX,2) - ZX(JX,3)
      ZXY(JX,1) = ZXY(JX,2) + ZXY(JX,2) - ZXY(JX,3)

```

600 DO 610 JY1=2,3

JY = 5 - JY1

ZY(JX,JY) = ZY(JX,JY-1)

ZX(JX,JY) = ZX(JX,JY-1)

ZXY(JX,JY) = ZXY(JX,JY-1)

610 CONTINUE

620 CONTINUE

Y3 = Y3 - 1.0/B4

Z33 = Z33 - Z3B2/B4

Z3A3 = Z3A3 - ZA3B2/B4

Z3B3 = Z3B2

ZA3B3 = ZA3B2

B3 = B4

GOTO 670

630 W4 = B2*(3.0*B3+B2)

W5 = 2.0*B3*(B3-B2) + W4

DO 660 JX=2,3

IF (JX.EQ.3 .AND. IX.EQ.LXP1) GOTO 640

IF (JX.EQ.2 .AND. IX.EQ.1) GOTO 640

ZY(JX,4) = (W4*ZB(JX-1,4)+W5*ZB(JX-1,5))/(W4+W5)

ZX(JX,4) = ZX(JX,3) + ZX(JX,3) - ZX(JX,2)

ZXY(JX,4) = ZXY(JX,3) + ZXY(JX,3) - ZXY(JX,2)

640 DO 650 JY=2,3

ZY(JX,JY) = ZY(JX,JY+1)

ZX(JX,JY) = ZX(JX,JY+1)

ZXY(JX,JY) = ZXY(JX,JY+1)

650 CONTINUE

660 CONTINUE

Y3 = Y4

Z33 = Z33 + Z3B3/B3

Z3A3 = Z3A3 + ZA3B3/B3

Z3B3 = Z3B4

ZA3B3 = ZA3B4

B3 = B2

670 IF (IX.NE.1 .AND. IX.NE.LXP1) GOTO 680

JX = IX/LXP1 + 2

JX1 = 5 - JX

JY = IY/LYP1 + 2

JY1 = 5 - JY

ZX(JX,JY) = ZX(JX1,JY) + ZX(JX,JY1) - ZX(JX1,JY1)

ZY(JX,JY) = ZY(JX1,JY) + ZY(JX,JY1) - ZY(JX1,JY1)

ZXY(JX,JY) = ZXY(JX1,JY) + ZXY(JX,JY1) - ZXY(JX1,JY1)

! DETERMINATION OF THE COEFFICIENTS OF THE POLYNOMIAL

680 ZX3B3 = (ZX34-ZX33)*B3

ZX4B3 = (ZX44-ZX43)*B3

ZY3A3 = (ZY43-ZY33)*A3

ZY4A3 = (ZY44-ZY34)*A3

$$\begin{aligned}
A &= ZA3B3 - ZX3B3 - ZY3A3 + ZXY33 \\
B &= ZX4B3 - ZX3B3 - ZXY43 + ZXY33 \\
C &= ZY4A3 - ZY3A3 - ZXY34 + ZXY33 \\
D &= ZXY44 - ZXY43 - ZXY34 + ZXY33 \\
E &= A + A - B - C \\
A3SQ &= A3*A3 \\
B3SQ &= B3*B3 \\
P02 &= (2.0*(Z3B3-ZY33)+Z3B3-ZY34)*B3 \\
P03 &= (-2.0*Z3B3+ZY34+ZY33)*B3SQ \\
P12 &= (2.0*(ZX3B3-ZXY33)+ZX3B3-ZXY34)*B3 \\
P13 &= (-2.0*ZX3B3+ZXY34+ZXY33)*B3SQ \\
P20 &= (2.0*(Z3A3-ZX33)+Z3A3-ZX43)*A3 \\
P21 &= (2.0*(ZY3A3-ZXY33)+ZY3A3-ZXY43)*A3 \\
P22 &= (3.0*(A+E)+D)*A3*B3 \\
P23 &= (-3.0*E-B-D)*A3*B3SQ \\
P30 &= (-2.0*Z3A3+ZX43+ZX33)*A3SQ \\
P31 &= (-2.0*ZY3A3+ZXY43+ZXY33)*A3SQ \\
P32 &= (-3.0*E-C-D)*B3*A3SQ \\
P33 &= (D+E+E)*A3SQ*B3SQ
\end{aligned}$$

! COMPUTATION OF THE POLYNOMIAL

$$\begin{aligned}
690 \quad DY &= VK - Y3 \\
Q0 &= P00 + DY*(P01+DY*(P02+DY*P03)) \\
Q1 &= P10 + DY*(P11+DY*(P12+DY*P13)) \\
Q2 &= P20 + DY*(P21+DY*(P22+DY*P23)) \\
Q3 &= P30 + DY*(P31+DY*(P32+DY*P33)) \\
DX &= UK - X3 \\
W(K) &= Q0 + DX*(Q1+DX*(Q2+DX*Q3))
\end{aligned}$$

700 CONTINUE

! NORMAL EXIT

RETURN

! ERROR EXIT

710 WRITE (IU0,99999)

GOTO 800

720 WRITE (IU0,99998)

GOTO 800

730 WRITE (IU0,99997)

GOTO 800

740 WRITE (IU0,99996)

GOTO 760

750 WRITE (IU0,99995)

760 WRITE (IU0,99994) IX, X(IX)

GOTO 800

770 WRITE (IU0,99993)

GOTO 790

780 WRITE (IU0,99992)

790 WRITE (IU0,99991) IY, Y(IY)

800 WRITE (IU0,99990) LX0, LY0, N0

RETURN

! FORMAT STATEMENTS

99999 FORMAT(1X/23H *** LX = 1 OR LESS./)

99998 FORMAT(1X/23H *** LY = 1 OR LESS./)

99997 FORMAT(1X/22H *** N = 0 OR LESS./)

```

99996 FORMAT(1X/27H   ***   IDENTICAL X VALUES./)
99995 FORMAT(1X/33H   ***   X VALUES OUT OF SEQUENCE./)
99994 FORMAT(7H   IX =, I6 , 10X, 7HX(IX) =, E12.3)
99993 FORMAT(1X/27H   ***   IDENTICAL Y VALUES./)
99992 FORMAT(1X/33H   ***   Y VALUES OUT OF SEQUENCE./)
99991 FORMAT(7H   IY =, I6 , 10X, 7HY(IY) =, E12.3)
99990 FORMAT(7H   LX =, I6 , 10X, 4HLY =, I6 , 10X, 3HN =, I7/
&
      36H ERROR DETECTED IN ROUTINE      ITPLBV)

```

```

END SUBROUTINE ITPLBV

```

```

FUNCTION MAX(x,y)
      IMPLICIT NONE

      DOUBLE PRECISION x, y, MAX

      IF (x > y) THEN

          MAX = x

      ELSE

          MAX = y

      END IF

```

```

END FUNCTION MAX

```

Vita

Jakob D. Brisby was born in Evansville, Indiana to David and Julie Brisby and raised in the small farming community in Waverly, Kentucky. While growing up Jakob always strove to succeed in his studies and developed a love for problem solving which led to his decision to pursue an engineering degree. Jakob enjoyed learning about physics and the solar system so it was easy for him to fall into the Aerospace Engineering discipline. This led him to moving to Knoxville, Tennessee where he completed his Bachelor's Degrees in Aerospace Engineering and Mechanical Engineering and his Master's in Aerospace Engineering.