A Monte-Carlo-Based Simulation of Jet Exhaust Nozzle Thermal Radiative Signatures

by

David D. Chapman 2Lt., USAF

Thesis Submitted to the Faculty of the Virginia Polytechnic Institute and State University in partial fulfillment of the requirements of the degree of Master of Science in Mechanical Engineering

APPROVED:

J.R. Mahan, Chairman

C.H. Stern

J.R. Thomas

December 1992
Blackburg, VA
LD
5655
V855
1972
CS265
C.12
A Monte-Carlo-Based Simulation of Jet Exhaust Nozzle Thermal Radiative Signatures
by

David D. Chapman 2Lt., USAF

Dr. J.R. Mahan, Chairman

(ABSTRACT)

An important consideration in the design of military aircraft is observability, or how visible an aircraft is to hostile weapons. One area of great importance to overall observability is an aircraft’s infrared signature, particularly the infrared emissions from the exhaust nozzle and plume. This creates the need for accurate modeling of the infrared signatures from these sources as a design aid or for comparison of candidate designs.

To that end, a parametric model has been developed based on the General Electric F110-GE-129 jet engine. The basis of the model is a highly flexible Monte-Carlo ray-trace formulation which is capable of simulating real surface behavior, such as specular reflections, and allows for variation of input parameters such as temperature, surface properties, and geometry. For given input parameters, the model predicts the overall infrared signature due to surface radiation from the exhaust nozzle and interior components. It also indicates the relative contribution of each interior surface to the overall signature and predicts the image that would be seen using infrared imaging equipment. The basic principles of the simulation method and the theory behind the application are discussed. Results are presented, primarily in graphical format, and recommendations are made for further work.
Acknowledgements

I wish to thank a number of individuals and organizations for their contributions to this research. I was fortunate to work with a group of such dedicated professionals as the faculty and staff of the Mechanical Engineering Department at Virginia Polytechnic Institute and State University (VPi&SU).

I would like to express my gratitude to Dr. J.R. Mahan, who served as my advisor, chief consultant, and role model throughout my graduate studies. His genuine interest in his students' work and his openness of mind on all subjects allowed me to maintain my own interest and served to enrich my overall academic experience. I would also like to extend my sincere thanks to Dr. James R. Thomas and Dr. Curtis H. Stern for serving on my advisory committee.

I also wish to thank the committee members of the Graduate Scholarship Committee, Department of Engineering Mechanics, U.S. Air Force Academy, Colorado, for giving me the opportunity to further my education. I would like to
extend my personal thanks to Lt. Col. Robert V. Pieri, who helped me in many ways both before and during my graduate education.

Mr. Thomas Hauer, manager of the Exhaust Systems Design Branch, General Electric Corporation, Evendale, Ohio, was a great help by providing me with literature and information on the General Electric F110 jet engine.

Finally, my deepest thanks to my girlfriend and fellow engineer, Heather Relation, who helped me in all aspects of my work, from technical advice to moral support. I could not have done it without her.
Table of Contents

Abstract ........................................................................................................... ii
Acknowledgements ......................................................................................... iii
Table of Contents ............................................................................................. v
List of Illustrations ........................................................................................... vii
List of Tables .................................................................................................... x
Nomenclature .................................................................................................... xi
1.0 Introduction .................................................................................................. 1
2.0 Technical Description of the Model ............................................................. 7
  2.1 The Monte-Carlo Method ........................................................................... 7
  2.2 Description of the F110-GE-129 Engine ....................................................... 14
  2.3 Geometry Modeling .................................................................................... 14
  2.4 Modeling of Radiative Properties ................................................................. 16
  2.5 Radiant Exchange Among Surfaces ............................................................. 17
  2.6 Program ANSIRS Input ............................................................................. 19
List of Illustrations

Fig. 1. Flowchart of Program ANSIRS ........................................ 38
Fig. 2. Sample random angles of emission ................................. 39
Fig. 3. Comparison of F110 and model geometry ......................... 40
Fig. 4. Flameholder model .................................................. 41
Fig. 5. Mixing duct model .................................................. 42
Fig. 6. Complete model geometry .......................................... 43
Fig. 7. Relationship between nozzle model and the virtual observer hemisphere ............................................... 44
Fig. 8. Comparison of modeled emissivity to measured data for René 41 ............................................................ 45
Fig. 9. Comparison of modeled emissivity to measured data for Haynes Alloy 188 ......................................................... 46
Fig. 10. Comparison of modeled emissivity to measured data for Inconel 718 ............................................................ 47
Fig. 11. Comparison of modeled emissivity to measured data for Inconel 625 ............................................................ 48
Fig. 12. Comparison of modeled emissivity to measured data for Hastelloy X ................................. 49

Fig. 13. Virtual observer surface locations relative to the engine nozzle ........................................ 50

Fig. 14. Sample of code convergence validation based on the reciprocity relation from one surface element of the variable nozzle to all surface elements of the variable nozzle ........... 51

Fig. 15. Sample of code convergence validation according to the symmetry requirement from two adjacent surface elements of the duct liner to all other elements of the duct liner ........... 52

Fig. 16. Effect of nozzle position on total radiative signature .............................. 53

Fig. 17. Contributions to the radiative signature when $\alpha = 0.3\alpha(T)$ for the duct liner (nozzle position of 30 degrees) ....................... 54

Fig. 18. Contributions to the radiative signature when $\alpha = 0.5\alpha(T)$ for the duct liner (nozzle position of 30 degrees) ....................... 55

Fig. 19. Contributions to the radiative signature when $\alpha = 0.65\alpha(T)$ for the duct liner (nozzle position of 30 degrees) ....................... 56

Fig. 20. Contributions to the radiative signature when $\alpha = 0.8\alpha(T)$ for the duct liner (nozzle position of 30 degrees) ....................... 57

Fig. 21. Contributions to the radiative signature when $\alpha = \alpha(T)$ for the duct liner (nozzle position of 30 degrees) ....................... 58

Fig. 22. Contributions to the radiative signature when $\alpha = 1.0$ for the duct liner (nozzle position of 30 degrees) ....................... 59

Fig. 23. Effects of duct liner specularity on radiative signature for a nozzle position of 30 degrees ....................... 60

Fig. 24. Simulated infrared image as viewed from location 1 ....................... 61

Fig. 25. Simulated infrared image as viewed from location 2 ....................... 62
Fig. 26. Simulated infrared image as viewed from location 3 ............. 63
Fig. 27. Simulated infrared image as viewed from location 4 ............. 64
Fig. 28. Simulated infrared image as viewed from location 5 ............. 65
Fig. 29. Influence of nozzle surfaces on radiative signature at location 4 .................................................. 66
Fig. 30. Influence of nozzle surfaces on radiative signature at location 5 .................................................. 67
Fig. 31. Influence of interior components on radiative signature at location 4 .................................................. 68
List of Tables

Table 1. Data used in $\chi^2$ statistical test of pseudorandom numbers ........................................... 70

Table 2. Numerical results of $\chi^2$ statistical test of pseudorandom numbers ........................................... 71

Table 3. Comparison of analytic view factors and view factors calculated by Program ANSIRS for $N = 4000$ bundles emitted from elements tested .................................................. 72

Table 4. Comparison of analytic view factors and view factors calculated by Program ANSIRS for $N = 6000$ bundles emitted from elements tested .................................................. 73

Table 5. Comparison of analytic view factors and view factors calculated by Program ANSIRS for $N = 20000$ bundles emitted from elements tested .................................................. 74

Table 6. Materials and temperature distribution assigned to nozzle structures ........................................... 75
Nomenclature

$A_i$ Surface area of element $i$ (m$^2$)

$D_{ij}$ Distribution factor from element $i$ to element $j$ (-)

$F_{ij}$ View factor from surface $i$ to surface $j$ (-)

$H_{ij}$ Influence factor from element $i$ to element $j$ (-)

$N$ Number of energy bundles emitted per unit area (-)

$Q$ Energy per unit time (W)

$q$ Radiative heat flux (W/m$^2$)

$r$ Radius (m)

$R_n$ Random number used in step $n$ of a Monte-Carlo ray trace technique (-)

$T$ Temperature (K)

Greek

$\alpha_i$ Total hemispherical absorptivity of element $i$ (-)

$\varepsilon_i$ Total hemispherical emissivity of element $i$ (-)

$\beta$ Angle from centerline of nozzle (rad)
\( \theta \)  Angle made with the surface normal (rad)

\( \kappa \)  Specular-to-total reflectivity ratio (-)

\( \rho \)  Reflectivity (-)

\( \sigma \)  Stefan-Boltzmann Constant, \( 5.6696 \times 10^{-8} \text{ W/m}^2\text{K}^4 \)

\( \phi \)  Angle made with the surface tangent (rad)

Subscripts

\( d \)  Diffuse

\( i \)  Element i

\( j \)  Element j

\( s \)  Specular

\( t \)  Total
1.0 Introduction

History has unequivocally proven the dominant role of heat-seeking weapons in combat, particularly air-to-air combat. Both the U.S. Air Force and the aerospace industry have paid heed to history's lesson and are engaged in ever increasing efforts to decrease aircraft vulnerability to infrared weapons [1]. A logical means of achieving that goal is by decreasing the infrared observability of an aircraft. This implies the need for detailed information regarding the infrared signature of a given aircraft at the design stage as well as at the operational stage. The experimental methods which currently fulfill part of this need are expensive and difficult, while the computational methods currently used are predominantly based on overly simplified models [2]. The objective of this research is the development of an accurate model of the infrared signature of a military jet exhaust nozzle. The model should be suitable for parametric studies of infrared emissions from candidate engine designs.
The importance of heat-seeking weapons has been well established and will continue to be of interest to military jet engine design engineers for the foreseeable future. Between 1958 and 1990, heat-seeking missiles accounted for 308 of the 407 known missile kills made in the air [3]. This statistic becomes more impressive when one considers that less than one-half (992 of 2014) of the missiles fired in the same period were heat seekers.

Additionally, the infrared-seeking missile responsible for most of the 308 kills, the U.S. Air Force Air Intercept Missile 9 (AIM9), is based on 1960's technology, whereas the radar-guided missiles fired were generally state-of-the-art. Finally, the U.S. Air Force currently pays ten times more for the radar-guided Advanced Medium Range Air-to-Air Missile (AMRAAM) than the AIM9 [3]. In light of shrinking defense budgets worldwide, the low cost of the AIM9 and infrared missiles in general makes them the weapon of choice for air-to-air combat.

The above discussion clearly demonstrates the proven performance of infrared-guided missiles and consequently supports the premise that such weapons will enjoy widespread use well into the next century. This establishes the need for accurate prediction of infrared signatures, and constitutes the author’s motivation for the study of jet engine nozzle infrared image simulation.

The author is not alone in the study of infrared signatures. The U.S. Air Force and the aerospace industry dedicate considerable resources to the purpose of determining jet exhaust nozzle infrared signatures. Facilities such as the altitude ground test facilities at Arnold Engineering Development Center (AEDC),

Introduction
Arnold Air Force Base, Tennessee, are used to obtain engine hot-part radiation data over a wide range of operational parameters. These tests are conducted for many types of jet engines, including the General Electric F110-GE-129 afterburning turbofan [4].

In addition to these experimental facilities, great effort has been expended developing computational models of infrared signatures. Computer simulations such as Spectral Imaging of Targets and Scenes (SPIRITS) [2] and Low Observable InfraRed (LOIR) [5] have been developed to predict observability of aircraft to heat-seeking weapons. The widespread use of these simulations by engineers and scientists is due to the fact that experimental data do not provide all of the information needed regarding infrared signatures and are often expensive to obtain and valid only for the engine being tested.

The task of the simulations cited above are to fill the void of data that either cannot be obtained or are too expensive to obtain experimentally. For example, the SPIRITS simulation accounts for all possible sources of infrared radiation, including exhaust nozzle and plume, airframe emissions, and scattered earth-, sun-, and skyshine. The SPIRITS code also allows for a variety of backgrounds [2]. The scope of such models is vast, and the simplifications needed to make such a simulation viable are substantial.

One source of infrared radiation that is frequently simplified is the exhaust nozzle. Because of the complex nature of the interior components and numerous dependent and independent parameters, current generations of models such as

**Introduction**
SPIRITS and LOIR model the exhaust nozzle as diffusely emitting, fixed-geometry cavities with constant properties [5]. Though such approximations may give adequate results in limited applications, we cannot claim a priori knowledge of exhaust nozzle infrared emissions based on results from these models. Finally, the most important drawback of these simplified models is that they preclude parametric studies of exhaust nozzle radiative signatures.

The main impetus for the current research is to meet the need for a highly accurate model based on first principles that includes few simplifications and allows for parametric studies of infrared emissions. The nozzle used as a reference in the model formulation is the F110-GE-129 afterburning turbofan. Technical information regarding the F110-GE-129 is available elsewhere [6,7]. The model developed forms a steady-state representation that includes complex geometries and realistic surface properties [8], such as specular reflections, but does not account for participating media such as hot exhaust gases and the atmosphere. Because participating media are not included, this model represents only the first step in creating a comprehensive jet engine infrared model. However, in certain wavelength intervals, particularly the long wavelength band (8-12μm), hot exhaust gases, whose radiating components are primarily carbon dioxide and water vapor, and the atmosphere are nearly transparent and may be omitted in the radiative analysis [9]. The model allows for variation of surface properties, temperature distribution and changes in geometry, such as a variable nozzle.

The numerical model developed, Program ANSIRS (Augmented turbofan

Introduction
exhaust Nozzle Simulated InfraRed Signature), is based on a Monte-Carlo ray-trace method which is widely used and well suited to complex geometries [10,11]. The Monte-Carlo approach involves simulating a large number of individual interactions between energy bundles and surfaces based on interpretation of surface radiative properties as probabilities, and predicting the overall behavior based on the outcome of all the individual interactions [12,13]. The Monte-Carlo method was first applied to radiative heat transfer in the analysis of participating media by Howell and Perlmutter and has since become a popular approach to radiant exchange between surfaces [14-16]. The method has proven to be particularly useful in the analysis of radiometer cavities, in which the effects of specularity are significant [17-21].

When compared with other methods for determining radiant exchange among surfaces, such as the projection [16] and net-exchange methods [11], the Monte-Carlo method competes favorably for use in this application. The projection methods are competitive only for diffusely emitting and reflecting surfaces or flat surfaces, whereas the Monte-Carlo method may be used to treat diffusely and specularly reflecting curved surfaces, such as those found inside a jet exhaust nozzle. The net-exchange methods, which involve solving simultaneous equations for all surfaces, become impractical in applications where many curved surfaces and blockage are present, as is the case in the F110 jet engine. This method necessitates finding the specular exchange factors, which can be a formidable task. This leads to the conclusion that a Monte-Carlo ray-trace method is the most

*Introduction*
suitable approach to a complex geometry with real surface properties [11].

The Monte-Carlo ray trace allows the distribution factor between any two interior nozzle surfaces to be determined. The distribution factor, as defined by Mahan and Eskin [20], allows for direct calculation of radiative transfer between all surfaces, including virtual observer surfaces located outside the nozzle. The total infrared signature of the nozzle consists of the intensity and direction of all thermal radiation that leaves the nozzle cavity. It may be determined by calculating the total radiative transfer between all interior nozzle surfaces and all observer surfaces. In addition, the matrix of distribution factors provides information such as the contribution of each interior surface to the overall signature. A technical discussion of the model is undertaken in Chapter 2. A discussion of the validation efforts, including the process of random number generation [22-24], is presented in Chapter 3.

The effects of changes in various parameters, such as temperature distribution, on the thermal radiative signature may be studied by altering these parameters and executing Program ANSIRS. A series of such parametric studies has been performed and the results are discussed in Chapter 4. Several improvements or additions could be made to the model which would make it more useful. In Chapter 5, a discussion of these modifications is presented along with recommendations for future research.

Introduction
2.0 Technical Description of the Model

The purpose of this chapter is to present a detailed discussion of the infrared signature model. The model is a Monte-Carlo-based simulation implemented for the General Electric F110-GE-129 jet engine. Therefore, technical descriptions of the Monte-Carlo method and the F110 engine are warranted. Additionally, the treatment of radiative surface properties and radiant exchange between surfaces is addressed. Finally, the user interface with program ANSIRS (Augmented turbofan exhaust Nozzle Simulated InfraRed Signature) is fully described.

2.1 The Monte-Carlo Method

"Monte-Carlo" refers to a statistical numerical technique that simulates actual events by interpreting physical variables as probabilities and performing a large number of simulated individual events or experiments. If enough experiments are performed, an accurate depiction of the actual behavior will result. The general theory of Monte-Carlo methods is well known, and detailed descriptions are available elsewhere [10-15]. Thus, this section focuses on the
use of the Monte-Carlo method in the current application.

The use of a Monte-Carlo technique is generally considered to be among the best of available methods for complex geometric structures such as the interior of an exhaust nozzle [11,16]. In addition, the Monte-Carlo method may be used to treat any type of surface, from a black surface to one with spectral, bidirectional properties. This consideration was of high priority in the selection of a modeling approach for the application at hand.

For this work, the Monte-Carlo method is used to simulate radiant exchange among surfaces. The mechanism of physical radiant exchange is the emission, absorption, and reflection of discrete energy bundles. The Monte-Carlo ray-trace technique simulates the emission of energy bundles from a surface and traces their life history until they are absorbed. Decisions throughout the energy bundle’s lifetime are made based on comparison of property values interpreted as probabilities with the values of random numbers. Perhaps the best way to explain the Monte-Carlo method as used in this research is to describe the life cycle of a typical energy bundle, explaining each step along the way. The flow chart of Program ANSIRS shown in Figure 1 depicts all of the steps involved in the simulation.

Before the Monte-Carlo method is applied, preliminary calculations determine the values of variables to be used in the model. This preprocessing is based on the input parameters and includes definition of the specific geometry and

Technical Description of the Model
assignment of surface properties. The details of this preprocessing are not directly related to the application of the Monte-Carlo ray-trace technique to this model. Therefore, the discussion of this aspect of the simulation is postponed until Sections 2.3 - 2.6.

Once the specifics of the model have been defined, a Monte-Carlo ray trace consisting of five distinct steps is applied to determine interactions between surfaces. Together, the five steps simulate the diffuse emission and subsequent reflection or absorption of one energy bundle. The first of these steps is to determine an exact source point for emission of an energy bundle from a given surface. This is accomplished by subroutine SOURCEPT, as depicted in the flow chart of Figure 1.

Two uniformly distributed random numbers between zero and unity, $R_1$ and $R_2$, are drawn to determine the coordinates of the emission point. The list of random numbers that are drawn is created by subroutine CHAN of Program ANSIRS, which numerically generates the random numbers as they are needed.

Because the model developed consists of planar, conical, cylindrical, and spherical surfaces, the means of source-point location differ from surface to surface. Therefore, in the following discussion Cartesian coordinates are used and a rectangular planar surface is assumed, taking $x$ and $z$ to be independent and $y$ to be dependent. For surface element $i$ with width $\Delta x$ and length $\Delta z$, as shown in
Figure 2, the $x$ and $z$ coordinates of emission are given by

\[ x_e = x_i + R_1 \Delta x \quad (1) \]

\[ z_e = z_i + R_2 \Delta z, \quad (2) \]

and $y_e$ is determined by the relationship between variables defined by the particular surface equation. The subscript "i" in Equations 1 and 2 refers to the coordinates of the reference corner of the surface element.

After establishing the point of emission, the second step is to determine the direction of emission, which is accomplished by subroutine DIRECTION shown in the flow chart in Figure 1. Two more random numbers, $R_3$ and $R_4$, are drawn to determine the angles $\theta$ and $\phi$ that define the direction of emission with respect to the surface normal and tangent, respectively, as shown in Figure 2. According to Siegel and Howell [10], for diffuse emission $\theta$ and $\phi$ are related to $R_3$ and $R_4$ by

\[ \theta = \sin^{-1}(\sqrt{R_3}) \quad (3) \]

and

\[ \phi = 2\pi R_4. \quad (4) \]

In the current work, only diffuse emission is considered. However, directional emission could be modeled if directivity relations corresponding to Equations 3 and 4 were available for the surface in question.

After determining the location and direction of emission, the third step is to
conduct a search to determine the nearest point of intersection between the emitted energy bundle and all surfaces in the enclosure. To find the nearest point of intersection, all surface equations that define the components of the model must be examined separately. Each equation must be solved simultaneously with the equation of the line defined by the direction and origin point of emission. This is done for all surface equations in turn, and solutions are accepted only if the intersection point lies on a part of the mathematical surface which represents a physical surface of the enclosure. If an intersection with a given physical surface is found, the distance between the point of emission and the point of intersection is calculated. If an intersection with another surface is subsequently found, the distance is again calculated and compared with the previous value. If the new distance is shorter than the previous one, that indicates that the energy bundle struck the enclosure at that point. The shortest distance is stored and updated as each surface equation is examined. In that way, when the search among all surfaces is completed, it is assured that the point of intersection is indeed the nearest, and thus the correct, point. This search is accomplished by subroutine INTERSECT shown in the Program ANSIRS flow chart of Figure 1.

After finding the point of intersection, the surface element containing this point must be determined so that the local surface radiative properties may be used. This is straightforward for one-sided surfaces, but for two-sided surfaces it becomes more difficult. For two sided surfaces, the side the energy bundle actually struck may be determined from the point of emission, the direction of
emission, and the distance to intersection.

A simple way to accomplish this is to calculate a point that the energy bundle passed through just before striking the surface of interest. This is done by subtracting a small vector amount from the known intersection distance and calculating the coordinates at the new distance along the original line of emission. This test point is then compared to the surface of intersection according to a rule defined by the particular surface equation. For example, if a ray strikes a two-sided planar surface, the coordinates of the test point are substituted into the equation of the plane and the result is examined. The sign of the result indicates the correct side of intersection and constitutes the rule for planar surfaces. Other types of surfaces are governed by different rules. After the rule is applied and the side of intersection is found, the corresponding surface element is found by subroutine SURFACE shown in the flow chart of Figure 1.

Once the surface element on which this intersection occurs is found, step four is to determine if the energy bundle is absorbed or reflected. This requires that another random number \( R_s \) be drawn. This number is then compared to the absorptivity of the surface of intersection by subroutine ABS. If the random number is less than or equal to the absorptivity, the incident bundle is absorbed and a counter is incremented. This counter, \( N_{ij} \), represents the "score" or number of bundles emitted by a specific element \( i \) that are absorbed by another element \( j \). If \( R_s \) is greater than the surface absorptivity, the energy bundle is reflected, in which case step five is invoked.

Technical Description of the Model
In this application, each surface has an associated reflectivity ratio, \( \kappa \), defined as

\[
\kappa = \frac{\rho_s}{\rho},
\]

(5)

where \( \rho \) is the sum of a specular component of reflectivity, \( \rho_s \), and a diffuse component of reflectivity, \( \rho_d \). In order to decide if a given reflection is diffuse or specular (step five), another random number \( R_s \), is drawn and compared to \( \kappa \) for the surface in question. If \( R_s \) is less than or equal to \( \kappa \) the reflection is specular; otherwise it is diffuse. If the reflection is diffuse, step two is repeated to determine a new random direction associated with the diffuse reflection. If the reflection is specular, the direction of reflection is determined by the angle of incidence and orientation of the surface. The reflected energy bundle is then traced to another surface of intersection and steps three through five are repeated until the bundle is absorbed, at which point the element of origin \( i \) and element of absorption \( j \) are stored. This entire process is repeated many times for each surface element.

When the process of emitting and tracing energy bundles is complete, the result is a matrix of values that represent the actual number of energy bundles emitted from a surface element that are absorbed by another element, or \( N_{ij} \). To convert this data into a meaningful form, the values may be nondimensionalized into distribution factors \( D_{ij} \) between elements. If each value represents the total number of bundles emitted from surface \( i \) that are absorbed by surface \( j \), then a nondimensional form may be calculated by dividing this value by the total number
of bundles emitted from surface i, or \( N_j \). The resulting distribution factor \( D_{ij} \) is a number between zero and unity that is equal to \( N_j / N_i \). The distribution factors may then be used to calculate radiative heat transfer between surfaces, as discussed in Section 2.5.

2.2 Description of the F110-GE-129 Engine

The General Electric F110-GE-129 engine is a two-shaft augmented turbofan which was first used in the General Dynamics F-16C in 1990 and has since been selected to power Japan’s FSX fighter [6]. This engine represents current technology for production aircraft engines. The exhaust nozzle section of the F110 is a convergent/divergent nozzle with hydraulic actuation. With the exception of the convergent/divergent portion, all structures downstream of the last stage turbine blades are stationary [25]. A cut-away view of the entire engine is shown in Figure 3. The exhaust nozzle section of the F110 is fabricated from five different materials: Inconel 718, Inconel 625, René 41, Hastelloy X, and Haynes Alloy 188 [7]. All of these materials are Nickel-Chromium alloys with various other constituents. The total length of the F110 engine is 4.83 m and the maximum diameter is 1.21 m [6].

2.3 Geometry Modeling

The model created consists of 1512 real surface elements and 552 virtual observer surface elements located on an imaginary hemispherical surface outside
the nozzle. The diameter of the virtual observer surface is five times the diameter of the nozzle opening in a fully constricted configuration, which is the configuration shown in Figure 3. There are 24 circumferential sectors for all portions of the model. Both the observer hemisphere and the model are axisymmetric about the engine centerline. The number of sectors was chosen based on the fact that the variable nozzle of the F110 is composed of 24 divergent flaps and seals. The relationship between the actual F110 engine and the radiative model is shown in Figure 3. Each surface element is defined by Cartesian coordinates to facilitate coordinate transformations. Each surface element is described by one surface equation that defines all points within the element. Elements defined by similar surface equations are grouped into sections. There are 42 such sections in the model.

To describe the entire geometry, 258 surface equations are required. The large number of equations is necessary to accurately model the intricacy of the actual engine, even though some structures are modeled as simplified surfaces. For example, surfaces such as the airfoil-shaped fuel injectors and the last-stage turbine blades are modeled as infinitely thin planar surfaces. Additionally, in order to preserve symmetry, the model includes 24 fuel injectors while the actual F110 has only 20 [25].

Complex structures within the nozzle are represented by combinations of surfaces. For example, the flameholder assembly shown in Figure 4, which stabilizes the turbulent diffusion flame during operation of the afterburner, is a
combination of eight different conic sections. Figure 5 depicts the flow mixing duct which mixes the core exhaust gases with the comparatively oxygen-rich bypass air in preparation for afterburning. An effort has been made to minimize simplification of the geometry. Various components as well as the entire model are shown in Figures 4 through 6. A depiction of the five-diameter virtual observer hemisphere and its relationship to the engine model is shown in Figure 7.

2.4 Modeling of Radiative Properties

The accuracy of the Monte-Carlo method in this application can be very high if the foundation for the decisions made during the simulation is solid. This foundation consists of the geometry of the model and the modeling of surface properties. The geometry, discussed in the last section, is very closely modeled, despite the simplification of the fuel injectors and turbine blades as plane elements. The surface properties for each material are modeled as accurately as available data permit.

In this application, all properties used are hemispherical total. This is primarily due to the fact that very little data exist for the materials involved and the data that do exist are inexact. Values of hemispherical total emissivity for the materials of interest in this work (Nickel-Chromium alloys) are generally given as a range of values at any given temperature instead of one specific value [8]. Emissivities and absorptivities, which are assumed to be equal, are modeled as linearly temperature
dependent on the basis of available data. Comparisons between property variations reported in the literature and the linear approximations for each material are shown in Figures 8 through 12. The reflectivity ratio, $\kappa$, for each material is treated as a constant in this model, again due to the lack of good experimental data. However, the model is capable of treating the reflectivity ratio as a function of temperature if the variation were known. The reflectivity ratios used for this model vary from 0.19 for Hastelloy X to 0.3 for Inconel 718 [8].

The consequences of the assumptions and approximations made for surface properties are not insignificant, but at the present time considerable uncertainty exists with regard to surface properties of such materials. If more accurate data become available in the future, the model can easily be modified to include temperature and wavelength dependence of any kind as well as directional dependence of properties. This is one of the advantages of the modeling approach used.

2.5 Radiant Exchange Among Surfaces

The treatment of radiant exchange among surfaces is relatively simple once the matrix of distribution factors is known. In the current application, all surface temperatures are kept constant. In order to obtain an accurate picture of the infrared signature then, flux incident to the hemisphere of virtual surfaces, shown in Figures 7 and 13, needs to be examined. The radiant flux from each individual
surface element i inside the nozzle to any virtual observer surface j is given by

\[ q_{ij} = \varepsilon_i A_j D_{ij} \sigma T_i^4. \] (6)

The total radiative flux through element j is obtained by summing over all 1512 surfaces i within the cavity, that is

\[ q_j = \sum_{i=1}^{1512} q_{ij}. \] (7)

If this calculation is performed for each virtual observer surface element j on the hemisphere, the entire radiative signature results.

The influence of any surface on the total radiative flux at a given surface location may also be obtained. This quantity is called the influence factor, H, and is defined by

\[ H_j = \frac{q_{ij}}{q_j}. \] (8)

This quantity may be used to determine "hot spots" or major contributors to the radiative flux at each location. This "hot spot" information is valuable to the engine designer because it shows where infrared signature reduction efforts should be focused. The influence factor differs from the distribution factor in that the distribution factor is a function of geometry and surface properties whereas the influence factor is a function of geometry, surface properties, and temperature. Care should be taken not to confuse the two quantities.

Technical Description of the Model
2.6 Program ANSIRS Input

The user inputs to program ANSIRS are the actual physical variables which define a given geometry and operating point. These inputs generally include:

1. Temperature distribution
2. Variable nozzle setting
3. Surface radiative properties

In addition, the user must input an integer as a random number seed to begin the random number sequence, as discussed later. Any one or all of these parameters may be changed for a given case study; however, the random number seed should be changed with every execution for good statistical accuracy.

2.7 Output of Program ANSIRS

Without knowledge of the output format, results from any program are simply a meaningless collection of numbers. The output contents and format of Program ANSIRS are fully described in this section.

Upon completing execution of Program ANSIRS, an output file containing the thermal radiative signature information is created. The information in this output file is in two distinct sections. The first section contains the coordinates of and radiative fluxes onto each of the virtual observer surfaces outside the nozzle. These data represent the overall radiative signature at the virtual observer

Technical Description of the Model
hemisphere for given input parameters. To determine the signature at distance other than five diameters, the radiative flux may be converted by recognizing that flux is inversely proportional to the square of the distance from the source.

The second section of the output file consists of the coordinates and influence factors $H_i$ from all 1512 surface elements i in the nozzle cavity to a given observer location j. For each execution, up to five observer locations on the virtual hemisphere may be selected, as shown in Figure 13. The corresponding influence factors for all elements are output for each of the five observer locations. This output differs from the total hemisphere signature described previously in that the influence factor matrix is a tool for finding the major contributors to the radiative flux at each observer location while the latter is a prediction of the radiative signature of the nozzle at the five-diameter virtual observer hemisphere.

The output of program ANSIRS is formatted for use in the graphical package TECPLOT [26], which allows for creation of color contour plots in which different colors represent different levels of a given variable. Such plots present a tremendous amount of information that can be understood at a glance. Also, such plots may be configured to simulate images generated by actual infrared imaging equipment, as discussed in Section 4.2.

2.8 Resource Requirements of the ANSIRS Code

As with many computer simulation methods, the Monte-Carlo method
inherently involves a tradeoff between accuracy and computing time. For this application, which is relatively complex and may involve a very large number of energy bundles (up to 51 million total bundles emitted), computing time is a factor to be considered. As the number of energy bundles emitted increases, the computing time required increases proportionally. The amount of storage required by this code is also a consideration. The distribution factors form a $1512 \times 2064$ array, requiring just under 50 Mbytes of storage for double-precision variables and 25 Mbytes for single precision. The total storage required to execute the program is approximately 65 Mbytes. The average CPU time required for execution of the program during parametric studies was 197.5 min on the IBM 3090 series 2 supercomputer. Because the maximum allowable CPU time on this machine is 6.0 hr, this simulation is limited to a maximum of 51 million total energy bundles emitted, after which allotted CPU time is exceeded.
3.0 Code Validation

A crucial step in the development and application of any computational model is testing and validation of the code. Simulations utilizing a Monte-Carlo technique may require large amounts of computer time, as discussed in the previous section. Therefore, the validation process is especially important because large amounts of valuable computer time may be wasted executing a faulty program. More importantly, engineering decisions leading to the loss of equipment and human life might result if erroneous conclusions are drawn; thus the process of validating the code is a necessary step that lends credibility to the conclusions drawn from the simulation. There are two aspects of the validation of Program ANSIRS. The first aspect is verification of the randomness of the random numbers
on which decisions are based throughout the simulation. The second aspect of validation is the comparison of the program output with known relationships from radiative heat transfer that apply to this specific geometry. Specifically, cross-diagonal symmetry of the distribution factor matrix can be checked.

3.1 Random Number Testing

The prudent observer will note that the results of any Monte-Carlo simulation is dependent on the randomness of the random numbers. In this application, the term random number is really a misnomer. A computer is a deterministic machine by nature, and so is incapable of producing truly random numbers. Instead, a pseudorandom number generator is often used which allows the computer to produce a series of simulated random numbers. A popular pseudorandom number generator given by Etter is used in the current work [24]. In this pseudorandom number generator, an integer input by the user serves as a seed upon which the sequence of numbers is based. As each pseudorandom number is calculated, the seed is automatically altered within the generator and is used to calculate the next pseudorandom number. This continues for as many
numbers as are required by the application.

A statistical test of these numbers may be performed to be certain that the numbers generated by the pseudorandom number generator behave randomly. The test performed in the current research is a $\chi^2$ test that compares observed and expected frequencies of the pseudorandom numbers. In this test, 10,000 uniformly distributed pseudorandom numbers are generated between zero and unity, and each number is placed into a bin. If the number is between 0.0 and 0.05, it is placed in bin 1; if the number is between 0.05 and 0.1 it is placed in bin 2, and so forth. When all 10,000 numbers have been sorted into their bins, the number in each bin is counted. The expected number in each bin for this case is 500 or 10,000/20. Then for each bin, the expected frequency is subtracted from the observed frequency, squared, and divided by the expected frequency. These values are summed for all bins and divided by 10,000. The result is the $\chi^2$ parameter, defined as

$$\chi^2 = \frac{1}{10000} \sum_{i=1}^{20} \frac{O_i - E_i}{E_i}. \quad (9)$$

In Equation 9, $O_i$ is the observed count of random numbers in bin $i$ and $E_i$ is the expected count of numbers in bin $i$. Sample expected and observed frequencies for each bin are compared in Table 1.

According to Whitney [22], if the $\chi^2$ parameter falls between 0.5 and 2.0, we may conclude that the numbers generated are sufficiently random. This test is

**Code Validation**
performed ten times with different values of the seed for each test. The results, shown in Table 2, indicate that the $\chi^2$ parameter falls well within the acceptable range for each test, and so it may be concluded that the pseudorandom number generator used in program ANSIRS is adequate.

### 3.2 Validation of Radiant Exchange Relationships

Certain well known relationships used in radiation heat transfer calculations can be exploited to validate parts of the code. These relationships include analytic view factors, the reciprocity relation, and relationships among surfaces in an enclosure. Additionally, for the current work, inherent relations exist due to the symmetric nature of the model. All of these relationships have been used to validate Program ANSIRS.

Perhaps the most useful relation in radiant exchange between diffuse surfaces is the reciprocity relation between diffuse view factors

$$A_i F_{ij} = A_j F_{ji}. \quad (10)$$

To be considered valid, a model must satisfy this relationship to a reasonable degree. Tests have been performed on each section of the model geometry to determine if this relationship holds. To test reciprocity, all surfaces of the nozzle

**Code Validation**
surfaces of the nozzle cavity are treated as black surfaces and the values of $F_{ij}$ are calculated between surfaces of adjacent sections. It should be noted that the distribution factors, $D_{ij}$, are actually what is calculated, but for black surfaces the distribution factor reduces to the view factor, $F_{ij}$. The ratio of $A_i F_{ij}$ to $A_j F_{ij}$ is then computed for each pair of surfaces. This ratio is found to converge to unity for all elements of all sections. A sample of this convergence is shown in Figure 14. These data are for section 1, or the outer portion of the variable nozzle, and it is clear that the ratio converges to unity, as expected. This test is performed on all sections and the results for each case are similar to the results shown for section 1 in Figure 14.

The next relationship to be tested is the summation requirement of enclosure theory, which is a statement of the conservation of energy. Simply put, the requirement is that for surfaces within an enclosure, the sum of the distribution factors from any one surface to all surfaces within the cavity must equal unity. As modeled, the exhaust nozzle can be considered an enclosure if the virtual observer hemisphere is included. This is because no energy bundle may leave the engine cavity without striking another surface within the nozzle or the observer surface, which is considered to be a black surface at zero degrees K and consequently absorbs all incident energy bundles. The tests for summation performed indicated a maximum error of no more than 0.71 percent for any surface.

Code Validation
The next test for validation of radiative relationships is a comparison between analytic view factors and results obtained using program ANSIRS. Analytic view factors between surfaces are calculated where possible and compared with distribution factors computed by program ANSIRS for diffuse surfaces. Samples of this comparison for different numbers of bundles emitted are shown in Tables 3 through 5. In these tables the view factors compared are those between rings of the duct liner. The largest portion of the duct liner is a cylindrical structure composed of seven equally spaced rings, as indicated in Figure 6. The analytic view factors between the middle, or fourth, ring and all seven rings are compared to the results obtained through Program ANSIRS. Even for a relatively small number of bundles emitted, the error is low, seldom exceeding one percent. This is partially due to the fact that the view factors are relatively large; thus, fewer bundles are required to give accurate results. Similar comparisons with various other analytic view factors for different structures in the model yielded similar results.

The final validation criterion is specific to this model. Because the model is axisymmetric, distribution factors can be expected to follow a pattern that reflects this symmetry. As noted previously, each ring of the model is divided into identical sectors for each circumferential ring of the model. This allows an analysis that treats only one of the 24 elements on each ring and defines the distribution

Code Validation 27
factors from all other elements on that ring in terms of this single sector. Exploiting such relationships in time-intensive simulations represents one of the most powerful means of decreasing resource requirements without significantly decreasing accuracy of the results. In the application at hand, this drastically reduces computing time required, but calls for verification of the symmetric relation. To verify this relation, distribution factors are computed from all elements of a given ring and subsequently compared to distribution factors between surfaces of identical mutual orientation. Such a test is performed on each section. A sample of the average results for all elements of section 4, the outer duct liner, appears in Figure 15. This is typical of results obtained for other sections. Again, the results indicate that the model converges to satisfy this requirement. This permits bundles to be emitted from only one of the 24 sectors without a decrease in overall accuracy.
4.0 Results

Now that program ANSIRS has been validated, the code may be used as described in Chapter 2 to perform any type of radiative heat transfer analysis desired. Several analyses are performed, including parametric studies of radiative signature, analysis of influence factors for several observer locations, and a series of simulations designed to create images similar to those that would be seen from infrared imaging equipment. The results of all these studies are presented in this chapter.

4.1 Results of Parametric Studies

A series of parametric studies is undertaken to demonstrate the potential application of this model. These studies consist of several trials in which one parameter is varied to determine its influence on the result. The parameters varied are: nozzle setting, duct liner absorptivity and duct liner reflectivity ratio. Temperatures used for all studies are based on an invented temperature distribution that at least qualitatively matches that of a real engine. For example, surfaces near the turbine blades are much hotter than surfaces near the nozzle.
exit. Temperatures were kept constant throughout all studies. Table 6 shows specific temperatures as well as materials used for the various parts of the engine nozzle. The values of absorptivity $\alpha$ and reflectivity ratio $\kappa$ given in Table 6 are the nominal values used unless otherwise stated.

In the first study the setting of the nozzle is varied in four increments, from fully open to fully constricted (as the angle between the second section of the variable nozzle and the engine centerline varies from zero to 30 deg respectively). The results for the total signatures calculated are shown in Figure 16. At its highest point, the radiative signature is more than doubled when comparing the open-nozzle (zero degrees) and closed-nozzle (30 deg) signatures.

Other parameters had a less pronounced effect on the total radiative signature. For example, varying the absorptivity of the nozzle liner had a relatively small impact on total signature, as trials with different absorptivities produced an insignificant variance. For this reason, only studies of contributions to the infrared signature due to the duct liner and other interior components are performed. The absorptivity of the duct liner was varied from $0.3\alpha(T)$ to $1.0\alpha(T)$ and then to 1.0, or the black case, where $\alpha(T)$ refers to the curves in Figures 8 through 12. The results shown in Figures 17 through 22 depict the effects observed. For each figure, the contribution to the signature at the given angle is divided into two parts, the contribution of the duct liner represented by the dashed lines, and the contributions of all other components, represented by the dotted lines. When the duct liner is more reflective, more of the radiation from the upstream components is reflected out of the nozzle, while at the same time the duct liner itself emits less.
radiation. Therefore, while the overall effects of varying absorptivity are inconclusive, the influence of interior components clearly dominates as absorptivity decreases.

The final series of studies involves varying the reflectivity ratio of the duct liner to determine the influence of that parameter. The ratio was varied from 0.25 to 1.0, and the results are shown in Figure 23. A more specularly reflecting duct liner results in significant signature increases. This result is not totally unexpected because of the position of the upstream engine components with respect to the observers. Thus, as the number of specular reflections on the duct liner is increased, more radiation from relatively hotter interior components would be expected to arrive at an exterior observer location.

**4.2 Simulated Infrared Images**

A series of trials was performed to predict the infrared image that would be obtained using infrared imaging equipment. For these trials, the incident radiation at a given observer location is taken to be "last bounce" radiation. This means that as energy bundles are traced through the cavity and strike the observer hemisphere, the source element for a given bundle is taken to be the last element the bundle struck (or was emitted from) before striking the observer surface. The resulting image is equivalent to that which would be detected by infrared imaging equipment. These trials were performed at the five virtual observer locations shown in Figure 13 using baseline parameters. The baseline parameters referred to include a nozzle setting of ten degrees, a temperature distribution as shown in Results
Table 6, absorptivities as shown in Figures 8 through 12, and reflectivity ratios of between 0.19 and 0.3 for the various materials. The resulting simulated images are shown in Figures 24 through 28, in which the net radiative flux is expressed in the units of W/m². From a very steep angle, as in the case of Figure 24, the radiative transfer is nearly uniform among all visible surfaces, probably due to the fact that most of the radiation observed is a result of direct emission from those surfaces, which are oriented nearly normal to the observer surface. As the viewing angle becomes more shallow, as in Figures 25 and 26, the infrared image becomes less uniformly distributed as reflected radiation from the upstream components plays an increasingly important role. As the viewing angle becomes still more shallow, as in Figure 27, the hotter portions deep in the nozzle play a greater role. For example, from this angle, some portions of the hotter upstream duct liner appear as the most prominent sources of infrared emissions. Finally, as we view the nozzle from a location near the centerline of the engine, as in Figure 28, the interior components such as the flameholder assembly and some of the turbine blades become the major contributors.

4.3 Influence Factor Images

A valuable tool in analyzing infrared signatures is the influence factor, defined by Equation 8, which relates the contribution of a given surface to the total signature at an observer location. This differs from the simulated infrared images in that the simulated images do not contain complete information about the source of the radiative signature, but rather indicate only the surfaces from which the

Results
radiation incident to the observer location was last reflected or directly emitted. Figures 29 through 31 indicate all of the original sources for the radiation incident to a given observer location and have been rotated 180 deg (with respect to the actual viewing angle) along the centerline of the engine to better reveal highly influential surfaces of the duct liner. In these figures, the values plotted are the influence factors from each surface multiplied by the total incident radiative flux and then multiplied by 100. The resulting value indicates the contribution to the actual flux incident to the observer surface from each surface (scaled up by a factor of 100 to increase graphic resolution).

Figure 29 shows the influence factors for the baseline case as viewed from location 4. From this figure, we see that the greatest influence on the radiative signature comes from upstream nozzle surfaces. Figures 30 and 31 show influence factors as viewed from location 5. Figure 31 includes the interior components. In these last two figures, there is little contribution to the image from the cooler downstream nozzle components and significant contributions by the interior components and the upstream duct liner. This is due to the fact that the upstream components are at a 50 percent higher temperature, and since radiative transfer varies as the fourth power of temperature, these sources have a greater influence on the radiative signature. The outer portion of the variable nozzle also appears to have a large impact despite its low temperature. This is probably due to the shorter distance between that structure and the observer hemisphere.

Results
5.0 Conclusions and Recommendations

5.1 Conclusions

Several conclusions can be drawn from the research described in this thesis:

1. A FORTRAN computer code, Program ANSIRS, has been created based on a Monte-Carlo ray-trace technique. The program is capable of simulating real surface properties and is suitable for parametric studies of thermal radiative signature. The code may be formatted for several different types of output, including simulated infrared images, total radiative signature, and influence factors.

2. The effects of specularity have been demonstrated to be significant. Therefore, including surface specularity in a jet exhaust nozzle radiative simulation is essential.

3. Efforts to streamline and accelerate the convergence of the Monte-Carlo method must continue. The computing time required to execute Program ANSIRS is its main disadvantage. At the extreme end, an interactive
simulation program would be most useful, as a design engineer could then explore many possibilities in a short time.

4. As expected, the major contributors to the infrared signatures examined in this model were found to be the upstream engine components. This is because they are typically at higher temperatures than the downstream nozzle surfaces.

5. Varying the absorptivity of the duct liner had a small impact on the total signature, even though the influence of the individual structures changed drastically. This leads to the conclusion that there is a tradeoff involved in changing surface properties of the duct liner and this approach may have a negligible impact on decreasing total signature.

5.2 Recommendations

The goal of this work was to demonstrate the viability of this approach for modeling infrared emissions of jet exhaust nozzles. This was accomplished by developing a highly accurate model of an F110-GE-129 class jet engine capable of handling real surface properties and suitable for use in parametric studies. The model accounts only for surface radiation and does not include gaseous radiation. The following recommendations are presented as possible topics for future research in this area.

5.2.1 Code Modifications

Modifications to this model could expand the scope of application to nearly

Conclusions and Recommendations
any situation. The treatment of participating media such as hot exhaust gases, even in a simple way, would greatly enhance the wavelength range over which this model is valid. Streamlining of Program ANSIRS could significantly decrease the time needed for execution. Vectorizing this code, or any Monte-Carlo simulation for that matter, could also significantly decrease the time requirements. Additionally, parallel processing of this code could produce great reductions in time required.

5.2.2 Simulation Applications

To be useful in the operational world, this model would need to execute in a very small amount of time, perhaps seconds. Because this is unlikely with any foreseeable modifications, an alternative approach could be taken. The model could be executed numerous times for many situations of interest and the results stored for retrieval at a later time. The retrieval would simply be a question of accessing the correct data and could certainly be accomplished in a few seconds. This would create an interactive database that involved a short waiting time for the user interface and allow "what if" situations to be examined quickly. The only drawback to this approach is the large amount of data storage required.
Figures
Figure 1. Flowchart of Program ANSIRS.
Figure 2. Geometry for angles of emission from an arbitrary surface.
Figure 3. Comparison of actual F110 engine and the numerical model studied in this thesis (cut-away view along longitudinal axis of engine).
Figure 4. Flameholder numerical model.
Figure 5. Mixing duct numerical model.
Figure 6. Complete model geometry (numbers in the center section indicate rings of the cylindrical duct liner which are used in comparison of view factors).
Figure 7. The relationship between the nozzle model and the virtual observer hemisphere.
Figure 8. Comparison of modeled emissivity to measured data for R41 [8].
$\epsilon = 0.85 + 0.025(T-589)/555$

Figure 9. Comparison of modeled emissivity to measured data for Haynes Alloy 188 [8].
Figure 10. Comparison of modeled emissivity to measured data for Inconel 718 [8].
Figure 11. Comparison of modeled emissivity to measured data for Inconel 625 [8].
Figure 12. Comparison of modeled emissivity to measured data for Hastelloy X [8].
Figure 13. Virtual observer locations relative to the exhaust nozzle model.
Figure 14. Sample of code convergence validation based on the reciprocity relation from one surface element of the variable nozzle to all elements of the variable nozzle (section 1).
Figure 15. Sample of code convergence validation according to the symmetry requirement from two adjacent surface elements of the duct liner to all other elements of the duct liner.
Figure 16. Effect of nozzle position on total radiative signature (peak at zero deg is due to extrapolation).
Figure 17. Contributions to the radiative signature when $\alpha = 0.3\alpha(T)$ for the duct liner (nozzle setting of 30 deg, peak at zero deg is due to extrapolation).
Figure 18. Contributions to the radiative signature when $\alpha = 0.5\alpha(T)$ for the duct liner (nozzle setting of 30 deg, peak at zero deg is due to extrapolation).
Figure 19. Contributions to the radiative signature when $\alpha = 0.65\alpha(T)$ for the duct liner (nozzle setting of 30 deg, peak at zero deg is due to extrapolation).
Figure 20. Contributions to the radiative signature when $\alpha = 0.8\alpha(T)$ for the duct liner (nozzle setting of 30 deg, peak at zero deg is due to extrapolation).
Figure 21. Contributions to the radiative signature when $\alpha = \alpha(T)$ for the duct liner (nozzle setting of 30 deg, peak at zero deg is due to extrapolation).
Figure 22. Contributions to the radiative signature when $\alpha = 1.0$ for the duct liner (nozzle setting of 30 deg, peak at zero deg is due to extrapolation of data).
Figure 23. Effects of duct liner specularity on radiative signature for a nozzle setting of 30 deg (peak at zero deg is due to extrapolation of data).
Figure 24. Simulated infrared image viewed from location 1 in Figure 13 (W/m²).

Figures 61
Figure 25. Simulated infrared image viewed from location 2 in Figure 13 (W/m²).

Figures 62
Figure 26. Simulated infrared image viewed from location 3 in Figure 13 (W/m²).

Figures
Figure 27. Simulated infrared image viewed from location 4 in Figure 13 (W/m²).
Figure 28. Simulated infrared image viewed from location 5 in Figure 13 (W/m²).

Figures 65
Figure 29. Influence of nozzle surfaces on radiative signature at location 4 in Figure 13 (W/m²).

Figures 66
Figure 30. Influence of nozzle surfaces on radiative signature at location 5 in Figure 13 (W/m²).
Figure 31. Influence of interior components on radiative signature at location 4 in Figure 13 (W/m²).
Tables
Table 1. Data used in the $\chi^2$ statistical test for randomness of 10000 pseudorandom numbers between zero and unity generated and used by Program ANSIRS.

<table>
<thead>
<tr>
<th>Interval of Interest</th>
<th>Observed Frequency Within Interval</th>
<th>Expected Frequency Within Interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.00 - 0.05</td>
<td>513</td>
<td>500</td>
</tr>
<tr>
<td>0.05 - 0.10</td>
<td>487</td>
<td>500</td>
</tr>
<tr>
<td>0.10 - 0.15</td>
<td>485</td>
<td>500</td>
</tr>
<tr>
<td>0.15 - 0.20</td>
<td>509</td>
<td>500</td>
</tr>
<tr>
<td>0.20 - 0.25</td>
<td>508</td>
<td>500</td>
</tr>
<tr>
<td>0.25 - 0.30</td>
<td>494</td>
<td>500</td>
</tr>
<tr>
<td>0.30 - 0.35</td>
<td>501</td>
<td>500</td>
</tr>
<tr>
<td>0.35 - 0.40</td>
<td>492</td>
<td>500</td>
</tr>
<tr>
<td>0.40 - 0.45</td>
<td>518</td>
<td>500</td>
</tr>
<tr>
<td>0.45 - 0.50</td>
<td>487</td>
<td>500</td>
</tr>
<tr>
<td>0.50 - 0.55</td>
<td>496</td>
<td>500</td>
</tr>
<tr>
<td>0.55 - 0.60</td>
<td>486</td>
<td>500</td>
</tr>
<tr>
<td>0.60 - 0.65</td>
<td>496</td>
<td>500</td>
</tr>
<tr>
<td>0.65 - 0.70</td>
<td>516</td>
<td>500</td>
</tr>
<tr>
<td>0.70 - 0.75</td>
<td>502</td>
<td>500</td>
</tr>
<tr>
<td>0.75 - 0.80</td>
<td>469</td>
<td>500</td>
</tr>
<tr>
<td>0.80 - 0.85</td>
<td>500</td>
<td>500</td>
</tr>
<tr>
<td>0.85 - 0.90</td>
<td>521</td>
<td>500</td>
</tr>
<tr>
<td>0.90 - 0.95</td>
<td>494</td>
<td>500</td>
</tr>
<tr>
<td>0.95 - 1.00</td>
<td>526</td>
<td>500</td>
</tr>
</tbody>
</table>
Table 2. Numerical results of the $\chi^2$ statistical test for randomness of the pseudorandom numbers generated by Program ANSIRS for ten values of the random number generator seed.

<table>
<thead>
<tr>
<th>Trial</th>
<th>Seed</th>
<th>$\chi^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>223</td>
<td>1.306</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>0.860</td>
</tr>
<tr>
<td>3</td>
<td>54</td>
<td>0.934</td>
</tr>
<tr>
<td>4</td>
<td>4299</td>
<td>1.036</td>
</tr>
<tr>
<td>5</td>
<td>1069</td>
<td>1.250</td>
</tr>
<tr>
<td>6</td>
<td>12058</td>
<td>0.837</td>
</tr>
<tr>
<td>7</td>
<td>9478</td>
<td>1.080</td>
</tr>
<tr>
<td>8</td>
<td>77114</td>
<td>0.998</td>
</tr>
<tr>
<td>9</td>
<td>121791</td>
<td>1.125</td>
</tr>
<tr>
<td>10</td>
<td>864</td>
<td>0.828</td>
</tr>
</tbody>
</table>
### Table 3.
Comparison of analytic and calculated view factors between the fourth ring and all seven rings of the cylindrical duct liner for \( N = 4000 \) bundles emitted from each surface element.

<table>
<thead>
<tr>
<th>View Factor</th>
<th>Analytic</th>
<th>Program ANSIRS</th>
<th>Error (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>( F_{4.1} )</td>
<td>0.05993</td>
<td>0.06055</td>
<td>1.03</td>
</tr>
<tr>
<td>( F_{4.2} )</td>
<td>0.07496</td>
<td>0.07487</td>
<td>0.12</td>
</tr>
<tr>
<td>( F_{4.3} )</td>
<td>0.09180</td>
<td>0.09159</td>
<td>0.23</td>
</tr>
<tr>
<td>( F_{4.4} )</td>
<td>0.10367</td>
<td>0.10338</td>
<td>0.28</td>
</tr>
<tr>
<td>( F_{4.5} )</td>
<td>0.09180</td>
<td>0.09364</td>
<td>1.96</td>
</tr>
<tr>
<td>( F_{4.6} )</td>
<td>0.07496</td>
<td>0.07459</td>
<td>0.50</td>
</tr>
<tr>
<td>( F_{4.7} )</td>
<td>0.05993</td>
<td>0.06078</td>
<td>1.42</td>
</tr>
</tbody>
</table>
Table 4. Comparison of analytic and calculated view factors between the fourth ring and all seven rings of the cylindrical duct liner for \( N = 8000 \) bundles emitted from each surface element.

<table>
<thead>
<tr>
<th>View Factor</th>
<th>Analytic</th>
<th>Program ANSIRS</th>
<th>Error (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>( F_{41} )</td>
<td>0.05993</td>
<td>0.05949</td>
<td>0.73</td>
</tr>
<tr>
<td>( F_{42} )</td>
<td>0.07496</td>
<td>0.07540</td>
<td>0.59</td>
</tr>
<tr>
<td>( F_{43} )</td>
<td>0.09180</td>
<td>0.09088</td>
<td>1.00</td>
</tr>
<tr>
<td>( F_{44} )</td>
<td>0.10367</td>
<td>0.10469</td>
<td>0.98</td>
</tr>
<tr>
<td>( F_{45} )</td>
<td>0.09180</td>
<td>0.09196</td>
<td>0.19</td>
</tr>
<tr>
<td>( F_{46} )</td>
<td>0.07496</td>
<td>0.07455</td>
<td>0.55</td>
</tr>
<tr>
<td>( F_{47} )</td>
<td>0.05993</td>
<td>0.06136</td>
<td>2.39</td>
</tr>
</tbody>
</table>
Table 5. Comparison of analytic and calculated view factors between the fourth ring and all seven rings of the cylindrical duct liner for $N = 20000$ bundles emitted from each surface element.

<table>
<thead>
<tr>
<th>View Factor</th>
<th>Analytic</th>
<th>Program ANSIRS</th>
<th>Error (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$F_{41}$</td>
<td>0.05993</td>
<td>0.05993</td>
<td>0.067</td>
</tr>
<tr>
<td>$F_{42}$</td>
<td>0.07496</td>
<td>0.07432</td>
<td>0.855</td>
</tr>
<tr>
<td>$F_{43}$</td>
<td>0.09180</td>
<td>0.09197</td>
<td>0.013</td>
</tr>
<tr>
<td>$F_{44}$</td>
<td>0.10367</td>
<td>0.10397</td>
<td>0.269</td>
</tr>
<tr>
<td>$F_{45}$</td>
<td>0.09180</td>
<td>0.09195</td>
<td>0.161</td>
</tr>
<tr>
<td>$F_{46}$</td>
<td>0.07496</td>
<td>0.07433</td>
<td>0.840</td>
</tr>
<tr>
<td>$F_{47}$</td>
<td>0.05993</td>
<td>0.06093</td>
<td>1.669</td>
</tr>
</tbody>
</table>
Table 6. Materials, surface properties and temperature distribution assigned to nozzle structures.

<table>
<thead>
<tr>
<th>Structure</th>
<th>Material</th>
<th>( \alpha(T) )</th>
<th>( \kappa )</th>
<th>T (K)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Outer Nozzle</td>
<td>René 41</td>
<td>0.77</td>
<td>0.2</td>
<td>800.0</td>
</tr>
<tr>
<td>Inner Nozzle</td>
<td>René 41</td>
<td>0.78</td>
<td>0.2</td>
<td>850.0</td>
</tr>
<tr>
<td>Outer Duct Liner</td>
<td>Inconel 625</td>
<td>0.77</td>
<td>0.3</td>
<td>830.0</td>
</tr>
<tr>
<td>Middle Duct Liner</td>
<td>Inconel 625</td>
<td>0.77</td>
<td>0.3</td>
<td>860.0</td>
</tr>
<tr>
<td>Inner Duct Liner</td>
<td>Inconel 625</td>
<td>0.79</td>
<td>0.3</td>
<td>950.0</td>
</tr>
<tr>
<td>Turbine Plane</td>
<td>René 41</td>
<td>0.85</td>
<td>0.2</td>
<td>1200.0</td>
</tr>
<tr>
<td>Outer Pilot Can</td>
<td>Hastelloy X</td>
<td>0.88</td>
<td>0.19</td>
<td>1150.0</td>
</tr>
<tr>
<td>Inner Pilot Can</td>
<td>Hastelloy X</td>
<td>0.87</td>
<td>0.19</td>
<td>1100.0</td>
</tr>
<tr>
<td>Bearing Cover</td>
<td>Hastelloy X</td>
<td>0.86</td>
<td>0.19</td>
<td>1000.0</td>
</tr>
<tr>
<td>Turbine Frame</td>
<td>René 41</td>
<td>0.85</td>
<td>0.2</td>
<td>1200.0</td>
</tr>
<tr>
<td>Turbine Frame Support</td>
<td>René 41</td>
<td>0.84</td>
<td>0.2</td>
<td>1170.0</td>
</tr>
<tr>
<td>Fuel Injectors</td>
<td>Haynes Alloy 188</td>
<td>0.88</td>
<td>0.24</td>
<td>1130.0</td>
</tr>
<tr>
<td>Inner Flameholder</td>
<td>Inconel 718</td>
<td>0.83</td>
<td>0.3</td>
<td>1190.0</td>
</tr>
<tr>
<td>1st Flameholder Brace</td>
<td>René 41</td>
<td>0.85</td>
<td>0.2</td>
<td>1195.0</td>
</tr>
<tr>
<td>Middle Flameholder</td>
<td>Inconel 718</td>
<td>0.83</td>
<td>0.3</td>
<td>1180.0</td>
</tr>
<tr>
<td>2nd Flameholder Brace</td>
<td>Haynes Alloy 188</td>
<td>0.82</td>
<td>0.24</td>
<td>1120.0</td>
</tr>
<tr>
<td>Outer Flameholder</td>
<td>Hastelloy X</td>
<td>0.87</td>
<td>0.19</td>
<td>1100.0</td>
</tr>
<tr>
<td>Flow Mixing Duct</td>
<td>Inconel 718</td>
<td>0.81</td>
<td>0.3</td>
<td>1100.0</td>
</tr>
</tbody>
</table>
References


References
1984.


Appendix A  ANSIRS Code Listing
?PROGRAM ANSIRS

(AFTERBURNING TURBOFAN NOZZLE SIMULATED INFRARED SIGNATURE)

THIS PROGRAM WILL IMPLEMENT A MONTE-CARLO RAY TRACE TECHNIQUE TO DETERMINE THE RADIATIVE CHARACTERISTICS OF AN AFTERBURNING TURBOFAN JET ENGINE.

THE GEOMETRY USED FOR THIS PROGRAM IS BASED ON THE GENERAL ELECTRIC F110 ENGINE CURRENTLY USED IN F-15 AND F-16 AIRCRAFT.

PROGRAM LANGUAGE - FORTRAN 77 ON VM1
PROGRAM AUTHOR - 2LT DAVE CHAPMAN
ADVISOR - DR. J. R. MAHAN

WHAT WE NEED TO KNOW TO USE THIS PROGRAM:

PROGRAM INPUTS:
- RANDOM NUMBER SEED (AN INTEGER)
- NOZZLE SETTING (0-30; 0 BEING FULLY OPEN)
- SURFACE PROPERTIES (DEFINED IN PROGRAM)
- TEMPERATURE DISTRIBUTION ADDED BY USER

VARIABLE DESCRIPTIONS:

NOZSET = Setting of nozzle (angle in degrees w/centerline)
X(I) = X coordinate, I = surface number
Y = Y coordinate
Z = Z coordinate
CHN = Subroutine that returns random numbers
XJ,YJ,ZJ = Coordinates of emission
THET, PHI = Angles of emission
NSHOTS(I) = Number of shots fired from surface I
ALP(I) = Absorptivity of surface I
XAP(I) = Ratio of diffuse/total reflectivity
FOR SURFACE I (1-KAPPA)
XNORMAL(I) = Surface normal for surface I in the X-DIRECTION (MAY BE A PXN)
T(I) = Temperature of surface I (ASSUMED CONSTANT)
AREA(I) = Total surface area of element I
Q1-Q4 (I) = Flux from element I to an observer location
XPLANE, YPLANE, AND ZPLANE WILL BE USED LATER TO FIND INTERSECTIONS WITH PLANAR SURFACES

DIMENSION X(2064), Y(2064), Z(2064)
DIMENSION ALP(2064), XAP(2064)
DIMENSION DF(1512,2064), T(2100), TMP(2100)
DIMENSION Q1(2100), Q2(2100), Q3(2100), Q4(2100), Q5(2100)
DIMENSION AREA(2064), NSHOTS(2064)
DIMENSION XPLANE(1500), YPLANE(1500), ZPLANE(1500)
DIMENSION U(1500), V(1500), W(1500)
INTEGER SEED
REAL PI
REAL R
REAL NOZSET
COMMON /BLOCK1/ DF
PARAMETER (PI = 3.141592653589793, NOZSET =30.0, L = 670)
PARAMETER (SIGMA = 5.66966-08)

Appendix A  ANSIRS Code Listing
L = OVERALL LENGTH OF ENGINE
R = RADIUS (FROM CENTERLINE OF ENGINE)

*NOTE IT IS ASSUMED THAT REFLECTIVITY = 1 - ABSORPTIVITY


THE FIRST LOOP INITIALIZES THE DISTRIBUTION FACTORS

DO 3 I = 1, 1512
DO 7 J = 1, 2064
DF(I, J) = 0.0

3 CONTINUE
7 CONTINUE

DO 10 K = 1, 96
  T(K) = 800
  ALP(K) = ALPHA(T(K), 2)
  XAP(K) = SPEC(T(K), 2)

10 CONTINUE

DO 20 K = 97, 168
  T(K) = 850
  ALP(K) = ALPHA(T(K), 2)
  XAP(K) = SPEC(T(K), 2)

20 CONTINUE

DO 25 K = 169, 240
  T(K) = 830
  ALP(K) = ALPHA(T(K), 4)
  XAP(K) = SPEC(T(K), 4)

25 CONTINUE

DO 30 K = 241, 288
  T(K) = 870

30 CONTINUE
\[
ALP(K) = \text{ALPHA}(T(K), 4) \\
XAP(K) = \text{SPEC}(T(K), 4)
\]

30 CONTINUE
DO 35 K = 289, 456
   T(K) = 860 \\
   ALP(K) = \text{ALPHA}(T(K), 4) \\
   XAP(K) = \text{SPEC}(T(K), 4)
35 CONTINUE
DO 40 K = 457, 480
   T(K) = 950 \\
   ALP(K) = \text{ALPHA}(T(K), 4) \\
   XAP(K) = \text{SPEC}(T(K), 4)
40 CONTINUE
DO 45 K = 481, 552
   T(K) = 990 \\
   ALP(K) = \text{ALPHA}(T(K), 4) \\
   XAP(K) = \text{SPEC}(T(K), 4)
45 CONTINUE
DO 50 K = 553, 660
   T(K) = 1200 \\
   ALP(K) = \text{ALPHA}(T(K), 2) \\
   XAP(K) = \text{SPEC}(T(K), 2)
50 CONTINUE
DO 55 K = 601, 672
   T(K) = 1150 \\
   ALP(K) = \text{ALPHA}(T(K), 3) \\
   XAP(K) = \text{SPEC}(T(K), 3)
55 CONTINUE
DO 60 K = 673, 744
   T(K) = 1100 \\
   ALP(K) = \text{ALPHA}(T(K), 3) \\
   XAP(K) = \text{SPEC}(T(K), 3)
60 CONTINUE
DO 65 K = 745, 792
   T(K) = 1000 \\
   ALP(K) = \text{ALPHA}(T(K), 3) \\
   XAP(K) = \text{SPEC}(T(K), 3)
65 CONTINUE
DO 70 K = 793, 840
   T(K) = 1200 \\
   ALP(K) = \text{ALPHA}(T(K), 2) \\
   XAP(K) = \text{SPEC}(T(K), 2)
70 CONTINUE
DO 75 K = 841, 888
   T(K) = 1170 \\
   ALP(K) = \text{ALPHA}(T(K), 2) \\
   XAP(K) = \text{SPEC}(T(K), 2)
75 CONTINUE
DO 80 K = 889, 936
   T(K) = 1130 \\
   ALP(K) = \text{ALPHA}(T(K), 5) \\
   XAP(K) = \text{SPEC}(T(K), 5)
80 CONTINUE
DO 85 K = 937, 1032
   T(K) = 1190 \\
   ALP(K) = \text{ALPHA}(T(K), 1) \\
   XAP(K) = \text{SPEC}(T(K), 1)
85 CONTINUE
DO 90 K = 1033, 1080
   T(K) = 1199 \\
   ALP(K) = \text{ALPHA}(T(K), 2) \\
   XAP(K) = \text{SPEC}(T(K), 2)
90 CONTINUE
DO 95 K = 1081, 1176
   T(K) = 1180

Appendix A  ANSIRS Code Listing
ALP(K) = ALPHA(T(K),1)
XAP(K) = SPEC(T(K),1)
95 CONTINUE
DO 100 K = 1177, 1224
   T(K) = 1126
   ALP(K) = ALPHA(T(K),5)
   XAP(K) = SPEC(T(K),5)
100 CONTINUE
DO 105 K = 1225, 1320
   T(K) = 1100
   ALP(K) = ALPHA(T(K),3)
   XAP(K) = SPEC(T(K),3)
105 CONTINUE
DO 110 K = 1321, 1368
   T(Y) = 1190
   ALP(K) = ALPHA(T(K),1)
   XAP(K) = SPEC(T(K),1)
110 CONTINUE
DO 115 K = 1369, 1464
   T(K) = 1100
   ALP(K) = ALPHA(T(K),2)
   XAP(K) = SPEC(T(K),2)
115 CONTINUE
DO 120 K = 1465, 1512
   T(K) = 1010
   ALP(K) = ALPHA(T(K),1)
   XAP(K) = SPEC(T(K),1)
120 CONTINUE
DO 130 K = 1513, 2064
   T(K) = 0.0
   ALP(K) = 1.0
   XAP(K) = 0.0
130 CONTINUE

THE FOLLOWING SECTION ASSIGN X, Y, AND Z COORDINATES
TO ALL STRUCTURES IN THE MODEL (SUBROUTINE GEOM). NEXT,
THE AREA OF EACH ELEMENT IS CALCULATED (FINDAREA). THEN,
THE NUMBER OF BUNDLES EMITTED FROM EACH SURFACE IS FOUND
(NUMSHOTS) AND THE RANDOM NUMBER SEED IS INPUT MANUALLY.
FUNCTION STIME IS NOT COMPATIBLE WITH BATCH OPERATION
THUS THE RANDOM NUMBER SEED MUST BE ENTERED BY HAND.
THE SEED MAY BE ANY INTEGER BETWEEN 0 AND 10000.

CALL GEOM(X,Y,Z,R,L,PI,SECTION,NOZSET,ZMAX,ZSTART)
CALL FINDAREA(X,Y,Z,PL,AREA)
CALL NUMSHOTS(AREA,NSHOTS)
SEED = 59

THIS LOOP CALCULATES SURFACE NORMAL COMPONENTS
AND A SURFACE POINT FOR ALL PLANAR SURFACES. THIS DATA
WILL BE NECESSARY TO FIND INTERSECTION POINTS LATER.

THIS LOOP IS FOR THE PLANAR VARIABLE NOZZLE SECTIONS

DO 135 I = 1,168
   IELEMENT = I
   CALL SOURCEPT(X,Y,Z,SEED,NSHOTS,PI,SECTION,1,1,AREA,
     & IELEMENT,UNO1,VO1,WO1,NOZSET,L,ZSTART,ZMAX)
   CALL NORMAL(UO1,VO1,WO1,X,Y,Z,SECTION,XNORMAL,YNORMAL,
     & ZNORMAL,PI,IELEMENT,XTANGENT,YTANGENT,ZTANGENT,
     & XCROSS,YCROSS,ZCROSS)
   XPLANE(I) = XNORMAL
   YPLANE(I) = YNORMAL
   ZPLANE(I) = ZNORMAL
   U(I) = UO1

Appendix A  ANSIRS Code Listing  84
V(I) = V01
W(I) = W01

135 CONTINUE

THIS LOOP IS FOR THE TURBINE FRAME MEMBERS AND FUEL
INJECTORS

DO 137 I = 841, 936
  IELEMENT = I
  CALL SOURCECT(X, Y, Z, SEED, NSHOTS, PI, SECTION, 1, 1, AREA,
   &     IELEMENT, U01, V01, W01, NOSET, L, ZSTART, ZMAX)
  CALL NORMAL(U01, V01, W01, X, Y, Z, SECTION, XNORMAL, YNORMAL,
   &     ZNORMAL, PI, IELEMENT, XTANGENT, YTANGENT, ZTANGENT,
   &     XCROSS, YCROSS, ZCROSS)
  XPLANE(I) = XNORMAL
  YPLANE(I) = YNORMAL
  ZPLANE(I) = ZNORMAL
  U(I) = U01
  V(I) = V01
  W(I) = W01

137 CONTINUE

THIS LOOP IS FOR THE PLANAR ELEMENTS OF THE FLOW MIXING
DUCT

DO 139 I = 1369, 1464
  IELEMENT = I
  CALL SOURCECT(X, Y, Z, SEED, NSHOTS, PI, SECTION, 1, 1, AREA,
   &     IELEMENT, U01, V01, W01, NOSET, L, ZSTART, ZMAX)
  CALL NORMAL(U01, V01, W01, X, Y, Z, SECTION, XNORMAL, YNORMAL,
   &     ZNORMAL, PI, IELEMENT, XTANGENT, YTANGENT, ZTANGENT,
   &     XCROSS, YCROSS, ZCROSS)
  XPLANE(I) = XNORMAL
  YPLANE(I) = YNORMAL
  ZPLANE(I) = ZNORMAL
  U(I) = U01
  V(I) = V01
  W(I) = W01

139 CONTINUE

THE FIRST LOOP BELOW INDEXED AS MSURF REPRESENTS
THE OPERATIONS THAT OCCUR FOR RAYS ORIGINATING FROM EACH
SURFACE ELEMENT. MSURF IS THE ELEMENT NUMBER. FOR EACH
ELEMENT WE EXECUTE AN INNER LOOP A NUMBER OF TIMES. THIS
INNER LOOP CALLS SUBROUTINES THAT EXECUTE THE FIRING AND
TRACING OF RAYS FROM THE ELEMENT. THIS IS DONE FOR AS
MANY SHOTS AS ARE FIRED FROM THE SPECIFIC ELEMENT, WHICH
IS A FUNCTION OF SURFACE AREA AND PROPERTIES. SUBROUTINE
SOURCECT LOCATES A RANDOM EMISSION POINT ON THE SURFACE
SUBROUTINE DIRECTION CALCULATES THE RANDOM ANGLES OF
EMISSION AND NORMAL DEFINES THE LOCAL COORDINATES AT THE
EMISSION POINT. SUBROUTINE INTERSECT FINDS AN INTERSECTION
POINT WHERE THE EMITTED RAY STRIKES A SURFACE WITHIN THE
ENGINE. SUBROUTINE SURFACE DETERMINES WHICH ELEMENT
CONTAINS THE POINT OF INTERSECTION. SHOTS ARE ONLY FIRED
FROM ONE ELEMENT ON EACH RING BECAUSE OF THE SYMMETRY
OF THE GEOMETRY, THUS THE INCREMENT OF 24 IN THE LOOP.

DO 160 MSURF = 1, 1489, 24
  NUM = NSHOTS(MSURF)
  M = MSURF
  IELEMENT = M
  DO 150 N = 1, NUM
    CALL SOURCECT(X, Y, Z, SEED, NSHOTS, PI, SECTION, NUM, N,
     &
 THE FOLLOWING LOOP ASSIGN VALUES TO THE DISTRIBUTION
FACTORS OF ELEMENTS ON EACH RING. THROUGH SYMMETRY, I CAN
DO THIS AND CUT REQUIRED TIME BY A FACTOR OF 24. AS AN
EXAMPLE THE DISTRIBUTION FACTOR FROM ELEMENT 5 TO
ELEMENT 34 (DF(5,34)) IS EQUAL TO THE DISTRIBUTION
FACTOR FROM ELEMENT 1 TO ELEMENT 30 (DF(1,30)).

DO 173 M = 1,1489,24
   J = M+1
   K = M+23
   DO 177 JDIM = J,K
   JDIM = JDIM - IFIX((JDIM-1)/24.0)*24.0
   KDIM = JDIM - M
   IF (JDIFF.GT.KDIFF) JDUM = JDIM - (KDIFF)
   IF (JDIFF.LE.KDIFF) THEN
     KDIFF = KDIFF - JDIFF
     JDUM = (JDIM - JDIFF) + 24 - KDIFF
   ENDIF
   DF(KDIM,JDIM) = DF(M,JDUM)
177 CONTINUE
175 CONTINUE
173 CONTINUE

THE FOLLOWING LOOP PERFORMS CALCULATIONS THAT DETERMINE
THE RADIATIVE EXCHANGE BETWEEN SURFACES IN THE ENCLOSURE.

THE GOVERNING EQUATION FOR EXCHANGE BETWEEN INDIVIDUAL
SURFACES AND THE SURFACE OF INTEREST IS:

AI*EI*DIJ*SIGMA*T^4=QIN (TO SURFACE OF INTEREST) WATTS/S
TO CONVERT TO FLUX, DIVIDE QIN BY AJ TO GET WATTS/M^2/S

SINCE OUR SURFACE OF INTEREST HAS ABSORPTIVITY OF 1.0,
WE CAN DETERMINE THE TOTAL FLUX INTO THE SURFACE BY:

QJ = ((1.0)*SUM OVER I(DIJ*T^4*EI*SIGMA*AI))/AJ
AJ = 1600.675605 (FOR A 5-DIAMETER HEMISPHERE WITH 552
ELEMENTS)

THOUGH IT MAY BE CONFUSING, I WILL USE T(J) AS A DUMMY
variable FOR Q(J) SO THAT IT WILL NOT BE CONFUSED WITH
Q1, Q2, Q3, AND Q4 USED LATER.

I AM ALSO ASSUMING THAT EMISSIVITY = ABSORPTIVITY, AN
ASSUMPTION THAT I WILL DISCUSS IN MY THESIS.

DO 180 I = 1,1512
   DO 190 J = 1,2064
      DF(I,J) = DF(I,J)/FLOAT(NSHOTS(I))
190 CONTINUE
180 CONTINUE
   DO 200 I = 1513, 2064
      SUM = 0.0
   DO 210 J = 1,1512
      SUM = SUM +
      & ALP(I)*T(I)*T(I)*T(I)*T(I)*DF(I,J)*SIGMA*AREA(I)
200 CONTINUE
   T(J) = (SUM)/1600.675605
200 CONTINUE
   DO 220 I = 1537,2088
      TMP(I) = T(I-24)
220 CONTINUE
   DO 230 I = 1537,2088
      T(I) = TMP(I)
230 CONTINUE

   THE LINES OF CODE BELOW GENERATE ONE FORM OF PROGRAM
   OUTPUT, IN THE FORM OF A 5-DIAMETER HEMISPHERE AROUND
   THE EXHAUST NOZZLE. IN THIS FORM, THE OUTPUT WILL BE THE
   NET RADIATIVE FLUX THROUGH EACH EQUAL AREA SECTOR
   OF THE HEMISPHERE, GIVING US AN IDEA OF THE OVERALL
   RADIATIVE SIGNATURE. THE OUTPUT WILL EITHER BE IN THIS
   FORM OR IN THE FORMAT OF HOT SURFACES, FROM THE PART OF
   CODE THAT FOLLOWS THIS PORTION.

   WRITE(8,*) 'TITLE = 'HEMISPHERE'
   WRITE(8,*) 'VARIABLES = X, Y, Z, TEMP'
   WRITE(8,*) 'ZONE T=ZONE 42*, I = 25, J=24,F=POINT'

240 FORMAT(1X,F7.1,
   & 1X, F7.1,1X,F7.1,1X,F8.2,1X,F8.2,1X,F8.2,1X,F8.2,1X,F8.2)
242 FORMAT (1X,F8.2,1X,F8.2,1X,F8.2,1X,F8.2,1X,F8.2)
   DO 250 I = 1,24
      ZVAL = 669.0 + FLOAT(I-1)*(374.999/23.0)
      R = SQRT((140525.0 - (ZVAL - 669.0)*(ZVAL-669.0)))
   DO 260 J = 1,125
      N = J
      IF (J.EQ.25) N = 1
      K = 1513 + 24*(I-1) + N
      XVAL = R*COS((FLOAT(J-1)*15.0*PI)/180.0)
      YVAL = R*SIN((FLOAT(J-1)*15.0*PI)/180.0)
      WRITE (8,242) XVAL,YVAL,ZVAL,T(K)
260 CONTINUE
250 CONTINUE

   THE FOLLOWING LINES OF CODE PROVIDE THE OUTPUT OF THE
   PROGRAM IN THE CORRECT FORMAT TO BE READ INTO THE
   SOFTWARE PACKAGE TECPLOT, WHICH CAN GENERATE FLOODED
   CONTOUR PLOTS INDICATING "HOT" REGIONS, ETC.

   DO 265 I = 1, 1512
      J1 = 1681
      J2 = 1777
      J3 = 1873
      J4 = 1969
      J5 = 2064
      AREAJ = 16.00675605
      Q1(I) = AREA(I)*
      & ALP(I)*SIGMA*T(I)*T(I)*T(I)*T(I)*DF(I,J1)/AREAJ
      Q2(I) = AREA(I)*
      & ALP(I)*SIGMA*T(I)*T(I)*T(I)*T(I)*DF(I,J2)/AREAJ
      Q3(I) = AREA(I)*
      & ALP(I)*SIGMA*T(I)*T(I)*T(I)*T(I)*DF(I,J3)/AREAJ
      Q4(I) = AREA(I)*
      & ALP(I)*SIGMA*T(I)*T(I)*T(I)*T(I)*DF(I,J4)/AREAJ

Appendix A  ANSIRS Code Listing 87
Q5(I) = AREA(I)*
& ALP(I)*SIGMA**(I)T(I)*T(I)*T(I)*T(I)*DF(I,J5)/AREAJ
265 CONTINUE
WRITE (8,*)' TITLE = *SHEL SECT*'
WRITE (8,*)' VARIABLES = X,Y,Z,FLUX1,FLUX2,FLUX3,FLUX4'
WRITE (8,*)' ZONE T=*ZONE-2*, I = 25, J = 24, F=POINT'
DO 270 I = 1,24
   DO 280 J = 1,25
      N = J
      IF (J.EQ.25) N = 1
      K = 24*(I-1) + N
      WRITE (8,240) X(K),Y(K),Z(K),Q1(K),Q2(K),Q3(K),Q4(K)
280 CONTINUE
270 CONTINUE

THE PORTION BELOW INCLUDES THE TURBINE PLANE, PILOT CAN
AND BEARING COVER.

WRITE (8,*)' ZONE T=*ZONE-2*, I = 25, J = 11, F=POINT'
DO 290 I = 1,11
   DO 300 J = 1,25
      N = J
      IF (J.EQ.25) N = 1
      K = 152 + 24*(I-1) + N
      IF (I.EQ.11) THEN
         WRITE (8,240) X(K),Y(K),Z(K),Q1(K),Q2(K),Q3(K),Q4(K)
      ENDIF
      IF (I.EQ.11) THEN
         WRITE(8,240)0,0,49.0,
         & Q1(K-24),Q2(K-24),Q3(K-24),Q4(K-24)
      ENDIF
300 CONTINUE
290 CONTINUE

THIS SECTION IS OUTPUT FOR THE TURBINE FRAME SHROUD

WRITE (8,*)' ZONE T=*ZONE-3*, I = 25, J = 3, F=POINT'
DO 310 I = 1,25
      N = I
      IF (I.EQ.25) N = 1
      K = 576 + N
      K1 = K+240
      WRITE(8,240) X(K),Y(K),
      & Z(K),Q1(K1),Q2(K1),Q3(K1),Q4(K1)
310 CONTINUE
DO 320 I = 1,2
   DO 330 J = 1,25
      N = J
      IF (J.EQ.25) N = 1
      K = 792 + 24*(I-1) + N
      WRITE (8,240) X(K), Y(K), Z(K),Q1(K),Q2(K),Q3(K),Q4(K)
330 CONTINUE
320 CONTINUE

THIS PORTION IS THE TURBINE FRAME MEMBERS

WRITE (8,*)' ZONE T=*ZONE-4*, I = 96, J = 24, F=FEPOINT'
DO 340 J = 1,24
      N = 1
      K = 576 + N
      K1=K+288
      WRITE(8,240) X(K), Y(K),
      & Z(K),Q1(K1),Q2(K1),Q3(K1),Q4(K1)
      WRITE(8,240)
      & X(K+24),Y(K+24),Z(K+24),Q1(K1),Q2(K1),Q3(K1),Q4(K1)
340 CONTINUE
WRITE (8, 240) & X(K+48), Y(K+46), Z(K+48), Q1(K1), Q2(K1), Q3(K1), Q4(K1)
WRITE (8, 240) X(K+216), & Y(K+216), Z(K+216), Q1(K1), Q2(K1), Q3(K1), Q4(K1)
340 CONTINUE
DO 342 I = 1, 93, 4
WRITE (8, *) I, I+1, I+2, I+3
342 CONTINUE

THE LINES BELOW REFER TO THE FUEL INJECTOR ARRAY

WRITE (8, *)' ZONE T="ZONE-5", I = 96, J = 24, F=FEPOINT'
DO 350 I = 1, 24
N = I
IF (I.EQ.25) N = 1
K = 888 + N
WRITE (8, 240) X(K), Y(K), Z(K), Q1(K), Q2(K), Q3(K), Q4(K)
WRITE (8, 240) X(K), Y(K), Z(K+24), Q1(K), Q2(K), Q3(K), Q4(K)
WRITE (8, 240) & X(K+24), Y(K+24), Z(K+24), Q1(K), Q2(K), Q3(K), Q4(K)
WRITE (8, 240) X(K+24), Y(K+24), Z(K), Q1(K), Q2(K), Q3(K), Q4(K)
350 CONTINUE
DO 352 I = 1, 93, 4
WRITE (8, *) I, I+1, I+2, I+3
352 CONTINUE


WRITE (8, *)' ZONE T="ZONE-6", I = 25, J = 4, F = POINT'
DO 360 I = 1, 4
DO 370 J = 1, 25
N = J
IF (J.EQ.25) N = 1
K = 936 + 24*(I-1) + N
IF (I.EQ.1) K = K+24
IF (I.EQ.4) K = K-24
WRITE (8, 240) X(K), Y(K), Z(K), Q1(K), Q2(K), Q3(K), Q4(K)
370 CONTINUE
360 CONTINUE
WRITE (8, *)' ZONE T="ZONE-7", I=96,J=24,F=FEPOINT'
DO 380 I = 1, 24
K = 1032 + I
K1 = K+24
X1 = COS((FLOT(I-1)*15.0*PI)/180.0)
X2 = COS((FLOT(I-1)*15.0+6.0)*PI/180.0)
Y1 = SIN((FLOT(I-1)*15.0*PI)/180.0)
Y2 = SIN((FLOT(I-1)*15.0+6.0)*PI/180.0)
WRITE (8, 240) 36.0*X1, 36.0*Y1,
& Z(K), Q1(K1), Q2(K1), Q3(K1), Q4(K1)
WRITE (8, 240) 36.0*X2, 36.0*Y2,
& Z(K), Q1(K1), Q2(K1), Q3(K1), Q4(K1)
WRITE (8, 240) & 77.0*X2, 77.0*Y2, Z(K+24), Q1(K1), Q2(K1), Q3(K1), Q4(K1)
WRITE (8, 240) & 77.0*X1, 77.0*Y1, Z(K+24), Q1(K1), Q2(K1), Q3(K1), Q4(K1)
380 CONTINUE
DO 382 I = 1, 93, 4
WRITE (8, *) I, I+1, I+2, I+3
382 CONTINUE
WRITE (8, *)' ZONE T="ZONE-8", I = 25, J = 4, F = POINT'
DO 400 I = 1, 4
DO 410 J = 1, 25

Appendix A  ANSIRIS Code Listing
N = J
IF (J .EQ. 25) N = 1
K = 1080 + 24*(I-1) + N
IF (I .EQ. 1) K = K+24
IF (I .EQ. 4) K = K-24
WRITE (8,240) X(K), Y(K), Z(K), Q1(K), Q2(K), Q3(K), Q4(K)
410 CONTINUE
409 CONTINUE
WRITE (8,’(A)’), ’ZONE T=*ZONE-10*, I=96,J=24, F=FEPOINT’
DO 420 I = 1, 24
   K = 1176 + I
   K1 = K+24
   X1 = COS((FLOAT(I-1)*15.0*PI)/180.0)
   X2 = COS((FLOAT(I-1)*15.0+6.0)*PI/180.0)
   Y1 = SIN((FLOAT(I-1)*15.0*PI)/180.0)
   Y2 = SIN((FLOAT(I-1)*15.0+6.0)*PI/180.0)
WRITE (8,240) 77.0*X1,
& 77.0*Y1,Z(K),Q1(K),Q2(K),Q3(K),Q4(K)
WRITE (8,240) 77.0*X2,
& 77.0*Y2,Z(K),Q1(K),Q2(K),Q3(K),Q4(K)
WRITE (8,240) 114.0*X1,
& 114.0*Y1,Z(K),Q1(K),Q2(K),Q3(K),Q4(K)
WRITE (8,240) 114.0*X2,
& 114.0*Y2,Z(K),Q1(K),Q2(K),Q3(K),Q4(K)
420 CONTINUE
DO 422 I = 1, 93, 4
   WRITE (8,’(A)’), I,I+1,I+2,I+3
422 CONTINUE
WRITE (8,’(A)’), ’ZONE T=*ZONE-10*, I = 25, J = 4, F = POINT’
DO 440 I = 1, 4
   DO 450 J = 1, 25
      N = J
      IF (J .EQ. 25) N = 1
      K = 1224 + 24*(I-1) + N
      IF (I .EQ. 1) K = K+24
      IF (I .EQ. 4) K = K-24
      WRITE (8,240) X(K), Y(K), Z(K), Q1(K), Q2(K), Q3(K), Q4(K)
440 CONTINUE
440 CONTINUE
C THE LAST FOUR ZONES COMPRISE THE FLOW MIXING DUCT
C
WRITE (8,’(A)’), ’ZONE T=*ZONE-11*, I=96, J = 24, F = FEPOINT’
DO 470 I = 1, 24
   K = 1320 + I
   K1 = K+24
   X1 = COS((FLOAT(I-1)*15.0*PI)/180.0)
   X2 = COS((FLOAT(I-1)*15.0+7.5)*PI/180.0)
   Y1 = SIN((FLOAT(I-1)*15.0*PI)/180.0)
   Y2 = SIN((FLOAT(I-1)*15.0+7.5)*PI/180.0)
WRITE (8,240) 90.0*X1,
& 90.0*Y1,Z(K),Q1(K),Q2(K),Q3(K),Q4(K)
WRITE (8,240) 90.0*X2,
& 90.0*Y2,Z(K),Q1(K),Q2(K),Q3(K),Q4(K)
WRITE (8,240) 80.0*X1,
& 80.0*Y1,Z(K),Q1(K),Q2(K),Q3(K),Q4(K)
WRITE (8,240) 80.0*X2,
& 80.0*Y2,Z(K),Q1(K),Q2(K),Q3(K),Q4(K)
470 CONTINUE
DO 472 I = 1, 93, 4
   WRITE (8,’(A)’), I,I+1,I+2,I+3
472 CONTINUE
WRITE (8,’(A)’), ’ZONE T=*ZONE-12*, I = 72, J=24 , F=FEPOINT’
DO 480 I = 1, 24
   K = 1368 + I
Appendix A  ANSIRS Code Listing 90
X1 = COS((FLOAT(I-1)*15.0*PI)/180.0)
X2 = COS((FLOAT(I-1)*15.0+7.5)*PI/180.0)
Y1 = SIN((FLOAT(I-1)*15.0*PI)/180.0)
Y2 = SIN((FLOAT(I-1)*15.0+7.5)*PI/180.0)
WRITE (8,240) 90.0*X2, 90.0*Y2,
& 27.0,Q1(K),Q2(K),Q3(K),Q4(K)
WRITE (8,240) 80.0*X2, 80.0*Y2,
& 44.32,Q1(K),Q2(K),Q3(K),Q4(K)
WRITE (8,240) 100.0*X2, 100.0*Y2,
& 44.32,Q1(K),Q2(K),Q3(K),Q4(K)
CONTINUE
DO 482 I = 1, 70, 3
WRITE (*,*) I, I+1, I+2, 1
482 CONTINUE
WRITE (*,*) 'ZONE T=*ZONE-13*, I = 72, J=24, F=PEPOINT'
DO 490 I = 1,24
K = 1464 + I
X1 = COS((FLOAT(I-1)*15.0*PI)/180.0)
X2 = COS((FLOAT(I-1)*15.0+15.0)*PI/180.0)
Y1 = SIN((FLOAT(I-1)*15.0*PI)/180.0)
Y2 = SIN((FLOAT(I-1)*15.0+15.0)*PI/180.0)
WRITE (8,240) 90.0*X2, 90.0*Y2,
& 27.0,Q1(K),Q2(K),Q3(K),Q4(K)
WRITE (8,240) 80.0*X2, 80.0*Y2,
& 44.32,Q1(K),Q2(K),Q3(K),Q4(K)
WRITE (8,240) 100.0*X2, 100.0*Y2,
& 44.32,Q1(K),Q2(K),Q3(K),Q4(K)
CONTINUE
DO 492 I = 1, 70, 3
WRITE (*,*) I, I+1, I+2, 1
492 CONTINUE
WRITE (*,*) 'ZONE T=*ZONE-14*, I=96,J = 24, F=PEPOINT'
DO 500 I = 1,24
K = 1464 + I
X1 = COS((FLOAT(I-1)*15.0+7.5)*PI/180.0)
X2 = COS((FLOAT(I-1)*15.0+15.0)*PI/180.0)
Y1 = SIN((FLOAT(I-1)*15.0+7.5)*PI/180.0)
Y2 = SIN((FLOAT(I-1)*15.0+15.0)*PI/180.0)
WRITE (8,240) 90.0*X1, 90.0*Y1,
& 2(K),Q1(K),Q2(K),Q3(K),Q4(K)
WRITE (8,240) 80.0*X2, 80.0*Y2,
& 2(K),Q1(K),Q2(K),Q3(K),Q4(K)
WRITE (8,240) 100.0*X2,
& 100.0*Y2,Z(K+24),Q1(K),Q2(K),Q3(K),Q4(K)
WRITE (8,240) 100.0*X1,
& 100.0*Y1,Z(K+24),Q1(K),Q2(K),Q3(K),Q4(K)
CONTINUE
DO 505 I = 1, 93, 4
WRITE (*,*) I, I+1, I+2, I+3
505 CONTINUE
CONTINUE
STOP
END

THE FOLLOWING FUNCTIONS WILL BE USED BY THE PROGRAM TO
DETERMINE THE SURFACE PROPERTIES OF A GIVEN SURFACE
AT A SPECIFIED TEMPERATURE. THIS IS NECESSARY BECAUSE
PROPERTIES SUCH AS EMISSIVITY CAN VARY GREATLY WITH
TEMPERATURE. THERE WILL BE ONE FUNCTION FOR EACH
PROPERTY. THE INPUT PARAMETER WILL BE TEMPERATURE AND
SURFACE MATERIAL, AND THE OUTPUT WILL BE THE PROPERTY
VALUE.

MATERIAL 1 = INCONEL 718
MATERIAL 2 = RENE 41

Appendix A ANSIRS Code Listing
MATERIAL 3 = HASTELLOY X
MATERIAL 4 = INCONEL 625
MATERIAL 5 = HAYNES ALLOY 188

REAL FUNCTION ALPHA(TEMPERATURE, MATERIAL)
  MAT = MATERIAL
  IF (MAT.EQ.1) ALPHA = 0.74 + 0.09 * (TEMPERATURE - 700.0) / 389.0
  IF (MAT.EQ.2) ALPHA = 0.75 + 0.105 * (TEMPERATURE - 711.0) / 500.0
  IF (MAT.EQ.3) ALPHA = 0.84 + 0.037 * (TEMPERATURE - 624.0) / 580.0
  IF (MAT.EQ.4) ALPHA = 0.74 + 0.09 * (TEMPERATURE - 700.0) / 389.0
  IF (MAT.EQ.5) ALPHA = 0.85 + 0.025 * (TEMPERATURE - 589.0) / 555.0
  RETURN
END

REAL FUNCTION SPEC(TEMPERATURE, MATERIAL)
  IF (MATERIAL.EQ.1) SPEC = 0.7
  IF (MATERIAL.EQ.2) SPEC = 0.8
  IF (MATERIAL.EQ.3) SPEC = 0.81
  IF (MATERIAL.EQ.4) SPEC = 0.7
  IF (MATERIAL.EQ.5) SPEC = 0.76
  RETURN
END

SUBROUTINE FINDRADIUS COMPUTES THE VALUE OF THE RADIUS
OF A PART OF THE GEOMETRY WHEN GIVEN THE Z-COORDINATE
AND THE SURFACE SECTION NUMBER (SECTION)

SUBROUTINE FINDRADIUS(ZVAL,
&
  ZVAL, NOZSET, L, R, SECTION, PI, ZSTART)

REAL NOZSET
REAL PI
SETTING = (NOZSET*PI) / 180
ZSTART = 549.0 - (NOZSET*11) / 30
RSTART = 106.133 - TAN(SETTING) * (ZSTART - 466)
IF (ZVAL.GE.ZSTART) THEN
  T1 = ZVAL - ZSTART
  T2 = 1 - NOZSET / 30
  T3 = RSTART + ((11) * T1) / 131
  R = RSTART + ((T2 * T1 * 15) / 131) + (11 * T1) / 131
ELSE
  IF (ZVAL.GE.465) THEN
    R = 106.133 - TAN(SETTING) * (ZVAL - 466)
  ELSE
    IF (ZVAL.GE.445) THEN
      R = 106.133
    ELSE
      IF (ZVAL.GE.417) THEN
        R = 106.133 + SIN(PI * ((ZVAL - 417) / 4.5))
      ELSE
        IF (ZVAL.GE.400) THEN
          R = 106.133
        ELSE
          IF (ZVAL.GE.333) THEN
            R = 120.5 - ((ZVAL - 333) * (13.5 / 66)) - SIN(PI * ((ZVAL - 333) / 4.5))
          ELSE
            IF (ZVAL.GT.147.0) THEN
              R = 120.5
            ELSE
              IF (ZVAL.GE.118.0) AND. ZVAL.LE.127.0) THEN
                IF (SECTION.EQ.18.0 OR. SECTION.EQ.19.0) THEN
                  R = 36.0 - ((ZVAL - 118.0) / 9.0) * 2.5
              ENDIF

Appendix A  ANSIRS Code Listing
IF (SECT.EQ.20.0.OR.SECTION.EQ.21.0) THEN
    R = 36.0 + ((ZVAL-118.0)/9.0)*2.5
ENDIF

ENDIF

IF (ZVAL.GE.94.0.AND.ZVAL.LE.118.0) THEN
    IF (SECTION.EQ.22.0.OR.SECTION.EQ.23.0) THEN
        R = 77.0 - ((ZVAL-94.0)/24.0)*41.0
    ENDIF
    IF (SECTION.EQ.26.0.OR.SECTION.EQ.27.0) THEN
        R = 77.0 + ((ZVAL-94.0)/9.0)*2.5
    ENDIF
ENDIF

ENDIF

IF (ZVAL.GE.94.0.AND.ZVAL.LE.99.0) THEN
    IF (SECTION.EQ.28.0.OR.SECTION.EQ.29.0) THEN
        R = 77.0 + ((ZVAL-94.0)/5.0)*37.0
    ENDIF
ENDIF

ENDIF

IF (ZVAL.GE.99.0.AND.ZVAL.LE.108.0) THEN
    IF (SECTION.EQ.30.0.OR.SECTION.EQ.31.0) THEN
        R = 114.0 - ((ZVAL - 99.0)/9.0)*2.5
    ENDIF
    IF (SECTION.EQ.32.0.OR.SECTION.EQ.33.0) THEN
        R = 114.0 + ((ZVAL - 99.0)/9.0)*2.5
    ENDIF
ENDIF

ENDIF

IF (ZVAL.GE.10.0.AND.ZVAL.LE.27.0) THEN
    IF (SECTION.EQ.12.0.OR.SECTION.EQ.13.0) THEN
        R = 90.0
    ENDIF
ENDIF

ENDIF

IF (ZVAL.GE.121.AND.SECTION.EQ.6.0) THEN
    R = 120.5 + SIN(PI*((ZVAL-121)/4.5))
ELSE
    IF (ZVAL.GE.61.AND.SECTION.EQ.7.0) THEN
        R = 120.5
    ELSE
        IF (ZVAL.GE.61.AND.SECTION.EQ.9.0) THEN
            R = 39.0 - ((10*(ZVAL - 10))/98)
        ELSE
            IF (ZVAL.GE.61.AND.SECTION.EQ.10.0) THEN
                R = 39.0 - ((10*(ZVAL - 10))/98)
            ENDIF
        ENDIF
    ENDIF
ELSE
    IF (ZVAL.GE.10) THEN
        IF (SECTION.EQ.7.0) R = 120.5
        IF (SECTION.EQ.9.0) R = 39 - ((10*(ZVAL - 10))/98)
        IF (SECTION.EQ.10.0) R = 39 - ((10*(ZVAL - 10))/98)
        IF (SECTION.EQ.8.0.AND.ZVAL.LE.49.0) THEN
            R = SQRT(1521 - (ZVAL - 10)*(ZVAL - 10))
        ENDIF
    ENDIF
    IF (ZVAL.GE.27.0.AND.ZVAL.LE.44.32) THEN
        IF (SECTION.EQ.34.0.OR.SECTION.EQ.35.0) THEN
            R = 90.0 - ((ZVAL-27.0)*(10.0/17.32))
        ENDIF
    ENDIF
    IF (SECTION.EQ.40.0.OR.SECTION.EQ.41.0) THEN
        R = 90.0 + ((ZVAL-27.0)*(10.0/17.32))
    ENDIF
ENDIF
ENDIF

ENDIF

ENDIF
ENDIF
ENDIF
ENDIF
ENDIF
ENDIF
ENDIF
ENDIF
ENDIF
ENDIF
ENDIF
ENDIF
RETURN
END

SUBROUTINE GEOM(X,Y,Z,R,L,PI,SECTION,NOZSET, ZMAX, &
ZSTART)

DIMENSION X(2064) , Y(2064) , Z(2064)
REAL NOZSET
REAL PI

ZSTART = 549.0 - (NOZSET*11)/30
ZVAL = ZSTART + 131.0
ZMAX = ZVAL
CALL FINDRADIUS(ZVAL,NOZSET,L,R,SECTION,PI,ZSTART)
DO 10 I = 1,24
   DUM1 = FLOAT(I-1)
   DUM2 = (DUM1*PI*15)/180.0
   X(I)  = R*COS(DUM2)
   Y(I)  = R*SIN(DUM2)
   Z(I)  = ZVAL
10 CONTINUE
DUM3 = 131.0/4.0
ZVAL = ZVAL - DUM3
CALL FINDRADIUS(ZVAL,NOZSET,L,R,SECTION,PI,ZSTART)
DO 20 I = 1,24
   DUM4 = FLOAT(I-1)
   DUM5 = (DUM4*PI*15)/180.0
   X(I+24)  = R*COS(DUM5)
   Y(I+24)  = R*SIN(DUM5)
   Z(I+24)  = ZVAL
20 CONTINUE
ZVAL = ZVAL - DUM3
CALL FINDRADIUS(ZVAL,NOZSET,L,R,SECTION,PI,ZSTART)
DO 30 I = 1,24
   DUM6 = FLOAT(I-1)
   DUM7 = (DUM6*PI*15)/180.0
   X(I+48)  = R*COS(DUM7)
   Y(I+48)  = R*SIN(DUM7)
   Z(I+48)  = ZVAL
30 CONTINUE
ZVAL = ZVAL - DUM3
CALL FINDRADIUS(ZVAL,NOZSET,L,R,SECTION,PI,ZSTART)
DO 40 I = 1,24
   DUM8 = FLOAT(I-1)
   DUM9 = (DUM8*PI*15)/180.0
   X(I+72)  = R*COS(DUM9)
   Y(I+72)  = R*SIN(DUM9)
   Z(I+72)  = ZVAL
40 CONTINUE
DUM10 = ZSTART - 465.0
DO 50 J = 1,3
   ZVAL = ZSTART - ((FLOAT(J-1)*DUM10)/3.0)
CALL FINDRADIUS(ZVAL, NOZSET, L, R, SECTION, PI, ZSTART)
DO 60 I = 1, 24
DUM1 = FLOAT(I-1)
DUM2 = (DUM1*PI*15)/180.0
IDUM1 = J*24 + 72 + I
X(IDUM1) = R*COS(DUM2)
Y(IDUM1) = R*SIN(DUM2)
Z(IDUM1) = ZVAL
60 CONTINUE
50 CONTINUE
ZVAL = 465.0
CALL FINDRADIUS(ZVAL, NOZSET, L, R, SECTION, PI, ZSTART)
DO 70 I = 1, 24
DUM3 = FLOAT(I-1)
DUM4 = (DUM3*PI*15)/180.0
X(I+168) = R*COS(DUM4)
Y(I+168) = R*SIN(DUM4)
Z(I+168) = ZVAL
70 CONTINUE
ZVAL = 444.0
CALL FINDRADIUS(ZVAL, NOZSET, L, R, SECTION, PI, ZSTART)
DO 80 I = 1, 24
DUM5 = FLOAT(I-1)
DUM6 = (DUM5*PI*15)/180.0
X(I+192) = R*COS(DUM6)
Y(I+192) = R*SIN(DUM6)
Z(I+192) = ZVAL
80 CONTINUE
ZVAL = 416.0
CALL FINDRADIUS(ZVAL, NOZSET, L, R, SECTION, PI, ZSTART)
DO 90 I = 1, 24
DUM7 = FLOAT(I-1)
DUM8 = (DUM7*PI*15)/180.0
X(I+216) = R*COS(DUM8)
Y(I+216) = R*SIN(DUM8)
Z(I+216) = ZVAL
90 CONTINUE
ZVAL = 400.0
CALL FINDRADIUS(ZVAL, NOZSET, L, R, SECTION, PI, ZSTART)
DO 100 I = 1, 24
DUM9 = FLOAT(I-1)
DUM10 = (DUM9*PI*15)/180.0
X(I+240) = R*COS(DUM10)
Y(I+240) = R*SIN(DUM10)
Z(I+240) = ZVAL
100 CONTINUE
ZVAL = 366.0
CALL FINDRADIUS(ZVAL, NOZSET, L, R, SECTION, PI, ZSTART)
DO 110 I = 1, 24
DUM11 = FLOAT(I-1)
DUM12 = (DUM11*PI*15)/180.0
X(I+264) = R*COS(DUM12)
Y(I+264) = R*SIN(DUM12)
Z(I+264) = ZVAL
110 CONTINUE
ZVAL = 358.428571428
DUM225 = 26.428571428
DO 120 J = 1, 7
ZVAL = ZVAL - DUM225
SECTION = 5.0
CALL FINDRADIUS(ZVAL, NOZSET, L, R, SECTION, PI, ZSTART)
DO 130 I = 1, 24
DUM23 = FLOAT(I-1)
DUM24 = (DUM23*PI*15)/180.0
IDUM2 = J*24 + 264 + I

Appendix A  ANSIRS Code Listing  95
X(IDUM2) = R*COS(DUM24)
Y(IDUM2) = R*SIN(DUM24)
Z(IDUM2) = ZVAL

130 CONTINUE
120 CONTINUE
ZVAL = 147.0
SECTION = 6.0
CALL FINDRADIUS(ZVAL, NOZSET, L, R, SECTION, PI, ZSTART)
DO 140 I = 1, 24
DUM25 = FLOAT(I-1)
DUM26 = (DUM25*PI*15)/180.0
X(I+456) = R*COS(DUM26)
Y(I+456) = R*SIN(DUM26)
Z(I+456) = ZVAL

140 CONTINUE
ZVAL = 121.0
SECTION = 7.0
DO 150 I = 1, 2
ZVAL = ZVAL - (30.0)*FLOAT(I-1)
CALL FINDRADIUS(ZVAL, NOZSET, L, R, SECTION, PI, ZSTART)
DO 160 J = 1, 24
DUM27 = FLOAT(J-1)
DUM28 = (DUM27*PI*15)/180.0
IDUM3 = I*24 + 456 + J
X(IDUM3) = R*COS(DUM28)
Y(IDUM3) = R*SIN(DUM28)
Z(IDUM3) = ZVAL

160 CONTINUE
150 CONTINUE
ZVAL = 61.0
SECTION = 7.0
CALL FINDRADIUS(ZVAL, NOZSET, L, R, SECTION, PI, ZSTART)
DO 170 I = 1, 24
DUM29 = FLOAT(I-1)
DUM30 = (DUM29*PI*15)/180.0
X(I+528) = R*COS(DUM30)
Y(I+528) = R*SIN(DUM30)
Z(I+528) = ZVAL

170 CONTINUE
ZVAL = 10.0
SECTION = 7.0
CALL FINDRADIUS(ZVAL, NOZSET, L, R, SECTION, PI, ZSTART)
DO 180 I = 1, 24
DUM31 = FLOAT(I-1)
DUM32 = (DUM31*PI*15)/180.0
X(I+552) = R*COS(DUM32)
Y(I+552) = R*SIN(DUM32)
Z(I+552) = ZVAL

180 CONTINUE
ZVAL = 10.0
R = 90.0
DO 190 I = 1, 24
DUM33 = FLOAT(I-1)
DUM34 = (DUM33*PI*15)/180.0
X(I+576) = R*COS(DUM34)
Y(I+576) = R*SIN(DUM34)
Z(I+576) = ZVAL

190 CONTINUE
ZVAL = 10.0
R = 39.0
DO 200 I = 1, 24
DUM35 = FLOAT(I-1)
DUM36 = (DUM35*PI*15)/180.0
X(I+600) = R*COS(DUM36)
Y(I+600) = R*SIN(DUM36)

Appendix A  ANSIRS Code Listing  96
Z(I+600) = ZVAL

200 CONTINUE
ZVAL = 43.0
SECTION = 9.0
CALL FINDRADIUS(ZVAL, NOZSET, L, R, SECTION, PI, ZSTART)
DO 210 I = 1, 124
   DUM37 = FLOAT(I-1)
   DUM38 = (DUM37*PI*15)/180.0
   X(I+624) = R*COS(DUM38)
   Y(I+624) = R*SIN(DUM38)
   Z(I+624) = ZVAL
210 CONTINUE
ZVAL = 76.0
SECTION = 9.0
CALL FINDRADIUS(ZVAL, NOZSET, L, R, SECTION, PI, ZSTART)
DO 220 I = 1, 124
   DUM39 = FLOAT(I-1)
   DUM40 = (DUM39*PI*15)/180.0
   X(I+648) = R*COS(DUM40)
   Y(I+648) = R*SIN(DUM40)
   Z(I+648) = ZVAL
220 CONTINUE
ZVAL = 108.0
SECTION = 10.0
CALL FINDRADIUS(ZVAL, NOZSET, L, R, SECTION, PI, ZSTART)
DO 230 I = 1, 124
   DUM41 = FLOAT(I-1)
   DUM42 = (DUM41*PI*15)/180.0
   X(I+672) = R*COS(DUM42)
   Y(I+672) = R*SIN(DUM42)
   Z(I+672) = ZVAL
230 CONTINUE
DO 240 I = 1, 124
   X(I+696) = X(I+648)
   Y(I+696) = Y(I+648)
   Z(I+696) = Z(I+648)
   X(I+720) = X(I+624)
   Y(I+720) = Y(I+624)
   Z(I+720) = Z(I+624)
   X(I+744) = X(I+600)
   Y(I+744) = Y(I+600)
   Z(I+744) = Z(I+600)
240 CONTINUE
ZVAL = 29.5
SECTION = 8.0
CALL FINDRADIUS(ZVAL, NOZSET, L, R, SECTION, PI, ZSTART)
DO 250 I = 1, 124
   DUM43 = FLOAT(I-1)
   DUM44 = (DUM43*PI*15)/180.0
   X(I+768) = R*COS(DUM44)
   Y(I+768) = R*SIN(DUM44)
   Z(I+768) = ZVAL
250 CONTINUE
DO 260 J = 1, 2
   DUM45 = FLOAT(J-1)
   ZVAL = 27.0 - (17.0*DUM445)
   SECTION = 12.0 + DUM445
   R = 90.0
   DO 270 I = 1, 124
      DUM45 = FLOAT(I-1)
      DUM46 = (DUM45*PI*15)/180.0
      IDUM4 = J*24 + 768 + I
      X(IDUM4) = R*COS(DUM46)
      Y(IDUM4) = R*SIN(DUM46)
      Z(IDUM4) = ZVAL
270 CONTINUE
270 CONTINUE
260 CONTINUE
ZVAL = 10.0
R = 39.0
DO 280 I = 1,2
   DUM47 = FLOAT(I-1)
   ZVAL = ZVAL + 30.0*DUM47
   SECTION = 14.0 + DUM47
   DO 290 J = 1,24
      IDUM5 = 24*I + 816 + J
      DUM48 = FLOAT(J-1)
      DUM49 = (DUM48*PI*15.0)/180.0
      X(IDUM5) = R*COS(DUM49)
      Y(IDUM5) = R*SIN(DUM49)
      Z(IDUM5) = ZVAL
290 CONTINUE
280 CONTINUE
R = 51.0
ZVAL = 50.0
DO 300 I = 1,2
   DUM50 = FLOAT(I-1)
   R = 51.0 + 29.0*DUM50
   ZVAL = 50.0 + 12.0*DUM50
   SECTION = 16.0 + DUM50
   DO 310 J = 1,24
      IDUM6 = 24*I + 864 + J
      DUM51 = FLOAT(J-1)
      DUM52 = (DUM51*PI*15.0)/180.0
      X(IDUM6) = R*COS(DUM52)
      Y(IDUM6) = R*SIN(DUM52)
      Z(IDUM6) = ZVAL
310 CONTINUE
300 CONTINUE
DO 320 I = 1,2
   DUM53 = FLOAT(I-1)
   ZVAL = 110.0 + 9.0*DUM53
   R = 16.0 - 2.5*DUM53
   SECTION = 18.0 + DUM53
   DO 330 J = 1,24
      IDUM7 = 24*I + 912 + J
      DUM54 = FLOAT(J-1)
      DUM55 = (DUM54*PI*15.0)/180.0
      X(IDUM7) = R*COS(DUM55)
      Y(IDUM7) = R*SIN(DUM55)
      Z(IDUM7) = ZVAL
      IF (DUM53.EQ.0.0) THEN
         X(IDUM7 + 48) = X(IDUM7)
         Y(IDUM7 + 48) = Y(IDUM7)
         Z(IDUM7 + 48) = Z(IDUM7)
      ENDIF
      IF (DUM53.EQ.1.0) THEN
         X(IDUM7 + 48) = 38.5*COS(DUM55)
         Y(IDUM7 + 48) = 38.5*SIN(DUM55)
         Z(IDUM7 + 48) = ZVAL
      ENDIF
330 CONTINUE
320 CONTINUE
DO 340 I = 1,2
   DUM56 = FLOAT(I-1)
   ZVAL = 118.0 - 24.0*DUM56
   R = 36.0 + 41.0*DUM56
   SECTION = 22.0 + DUM56
   DO 350 J = 1,24
      DUM57 = FLOAT(J-1)
      DUM58 = (DUM57*PI*15.0)/180.0
      IF (DUM57.EQ.0.0) THEN
         X(IDUM57 + 48) = X(IDUM57)
         Y(IDUM57 + 48) = Y(IDUM57)
         Z(IDUM57 + 48) = Z(IDUM57)
      ENDIF
      IF (DUM57.EQ.1.0) THEN
         X(IDUM57 + 48) = 38.5*COS(DUM55)
         Y(IDUM57 + 48) = 38.5*SIN(DUM55)
         Z(IDUM57 + 48) = ZVAL
      ENDIF
350 CONTINUE
Appendix A  ANSIRS Code Listing
IDUM8 = 24*I + J + 1008
X(IDUM8) = R*COS(DUM58)
Y(IDUM8) = R*SIN(DUM58)
Z(IDUM8) = ZVAL

CONTINUE

DO 360 I = 1,2
DUM59 = FLOAT(I-1)
ZVAL = 94.0 + 9.0*DUM59
R = 77.0 - 2.5*DUM59
SECTION = 24.0 + DUM59
DO 370 J = 1,24
IDUM9 = 24*I + 1056 + J
DUM60 = FLOAT(J-1)
DUM61 = (DUM60*PI*15)/180.0
X(IDUM9) = R*COS(DUM61)
Y(IDUM9) = R*SIN(DUM61)
Z(IDUM9) = ZVAL
IF (DUM59.EQ.0.0) THEN
  X(IDUM9 + 48) = X(IDUM9)
  Y(IDUM9 + 48) = Y(IDUM9)
  Z(IDUM9 + 48) = Z(IDUM9)
ENDIF
IF (DUM59.EQ.1.0) THEN
  X(IDUM9 + 48) = 79.5*COS(DUM61)
  Y(IDUM9 + 48) = 79.5*SIN(DUM61)
  Z(IDUM9 + 48) = ZVAL
ENDIF

CONTINUE

DO 380 I = 1,2
DUM62 = FLOAT(I-1)
ZVAL = 94.0 + 5.0*DUM62
R = 77.0 + 37.0*DUM62
SECTION = 28.0 + DUM62
DO 390 J = 1,24
DUM63 = FLOAT(J-1)
DUM64 = (DUM63*PI*15.0)/180.0
IDUM10 = 24*I + J + 1152
X(IDUM10) = R*COS(DUM64)
Y(IDUM10) = R*SIN(DUM64)
Z(IDUM10) = ZVAL

CONTINUE

DO 400 I = 1,2
DUM65 = FLOAT(I-1)
ZVAL = 99.0 + 9.0*DUM65
R = 114.0 - 2.5*DUM65
SECTION = 30.0 + DUM65
DO 410 J = 1,24
IDUM11 = 24*I + 1200 + J
DUM66 = FLOAT(J-1)
DUM67 = (DUM66*PI*15)/180.0
X(IDUM11) = R*COS(DUM67)
Y(IDUM11) = R*SIN(DUM67)
Z(IDUM11) = ZVAL
IF (DUM65.EQ.0.0) THEN
  X(IDUM11 + 48) = X(IDUM11)
  Y(IDUM11 + 48) = Y(IDUM11)
  Z(IDUM11 + 48) = Z(IDUM11)
ENDIF
IF (DUM65.EQ.1.0) THEN
  X(IDUM11 + 48) = 116.5*COS(DUM67)
  Y(IDUM11 + 48) = 116.5*SIN(DUM67)
  Z(IDUM11 + 48) = ZVAL

Appendix A  ANSIRS Code Listing
ENDIF

410 CONTINUE

420 CONTINUE

DO 430 I = 1,2
  DUM68 = FLOAT(I-1)
  ZVAL = 27.0 + 17.32*DUM68
  R = 90.0 - 10.0*DUM68
  SECTION = 34.0 + DUM68
  DO 430 J = 1,2
    IDUM12 = 24*I + 1296 + J
    DUM69 = FLOAT(J-1)
    DUM70 = (15.0*PI*DUM69/180.0)
    DUM71 = ((15.0*DUM69+7.5)*PI/180.0)
    X(IDUM12) = R*COS(DUM70)
    Y(IDUM12) = R*SIN(DUM70)
    Z(IDUM12) = ZVAL
    IF (DUM68.EQ.0.0) THEN
      X(IDUM12+144) = R*COS(DUM71)
      Y(IDUM12+144) = R*SIN(DUM71)
      Z(IDUM12+144) = ZVAL
    ENDIF
    IF (DUM68.EQ.1.0) THEN
      X(IDUM12+144) = 100.0*COS(DUM71)
      Y(IDUM12+144) = 100.0*SIN(DUM71)
      Z(IDUM12+144) = ZVAL
    ENDIF
  END DO
  CONTINUE

440 CONTINUE

DO 450 J = 1,2
  R = 100.0
  ZVAL = 44.32
  IDUM13 = 1368 + J
  DUM72 = FLOAT(J)
  DUM73 = (DUM72*PI*15.0)/180.0
  DUM74 = ((DUM72*15.0-7.5)*PI)/180.0
  X(IDUM13) = R*COS(DUM74)
  Y(IDUM13) = R*SIN(DUM74)
  Z(IDUM13) = ZVAL
  X(IDUM13+48) = R*COS(DUM73)
  Y(IDUM13+48) = R*SIN(DUM73)
  Z(IDUM13+48) = ZVAL
  CONTINUE

450 CONTINUE

ZVAL = ZSTART + 131.0
CALL FINDRADIUS(ZVAL,NZSET,L,R,SECTION,PI,ZSTART)
EXITAREA = PI*R*R
DUM89 = EXITAREA/23.0
DUM90 = DUM89/PI
R1 = 0.0
DO 490 J = 1,23
  R = R1 + SQRT(FLOAT(J)*DUM90)
  DO 500 I = 1,24

Appendix A  ANSIRS Code Listing
DUM91 = FLOAT(I-1)
DUM92 = (DUM91*PI*15)/180.0
IDUM50 = J*24 + I + 1488
X(IDUM50) = R*COS(DUM92)
Y(IDUM50) = R*SIN(DUM92)
Z(IDUM50) = ZVAL
500 CONTINUE
490 CONTINUE

RETURN
END

SUBROUTINE FINDLENGTH SIMPLY COMPUTES THE DISTANCE (S)
   BETWEEN TWO POINTS (A1,B1,C1) AND (A2,B2,C2)

SUBROUTINE FINDLENGTH(A1,B1,C1,A2,B2,C2,S)
   S = SQRT((A1-A2)^2+(B1-B2)^2+(C1-C2)^2)
RETURN
END

SUBROUTINE FINDAREA IS USED TO FIND THE AREA OF SURFACE
   ELEMENTS TO BE USED TO FIND THE NUMBER OF SHOTS FIRED

SUBROUTINE FINDAREA(X,Y,Z,PI,AREA)
DIMENSION X(2064), Y(2064), Z(2064), AREA(2064)
REAL PI

DO 10 I = 1,7
   K = 1 + (I-1)*24
   J = 1 + I*24
   CALL FINDLENGTH(X(K),Y(K),Z(K),X(K+1),Y(K+1),Z(K+1),S1)
   CALL FINDLENGTH(X(J),Y(J),Z(J),X(J+1),Y(J+1),Z(J+1),S2)
   BASE = (S1+S2)*0.5
   HEIGHT = S3
   AREA(K) = BASE*HEIGHT
   DO 20 L = 1,23
      AREA(L+K) = AREA(K)
   20 CONTINUE
   10 CONTINUE

THE ABOVE AREAS ARE FOR THE PLAT-SURFACED VARIABLE
   NOZZLE.
   NOW I MUST INCORPORATE THE FACT THAT THE SURFACES ARE
   CURVED WHEN CALCULATING THE AREA FOR THE REST OF THE
   SHELL ELEMENTS

DO 30 I = 7,22
   J = 1 + I*24
   ARCLENGTH = (PI/12.0)*X(J)
   CALL FINDLENGTH(X(J),
   &       Y(J),Z(J),X(J+24),Y(J+24),Z(J+24),S)
   IF (I.GT.9.AND.I.LE.11) THEN
      S = 1.0
      DHDR = 68.0/(120.5 - 106.133)
      CONEK1 = SQRT((1 + DHDR*DHDR)
      ARCLENGTH = PI*CONEK1*(X(J+24)*X(J+24)-X(J)*X(J))/24.0
   ENDIF
   AREA(J) = ARCLENGTH*S
   DO 40 K = 1,23
      L = J + K
50 DUM91 = 0
51 DUM92 = 0
52 IDUM50 = 0
53 X(IDUM50) = 0
54 Y(IDUM50) = 0
55 Z(IDUM50) = 0
56 500 CONTINUE
57 490 CONTINUE
60 RETURN
END

Appendix A  ANSIRS Code Listing   101
AREA(L) = AREA(J)
40 CONTINUE
30 CONTINUE

NOW I MUST CORRECT THE SURFACE AREA OF SINUSOIDAL PARTS
OF THE SHELL WHICH ARE ABOUT 10% LARGER THAN CALCULATED

DO 50 I = 1, 24
  AREA(I+192) = AREA(I+192) * 1.1094866
  AREA(I+240) = AREA(I+240) * 1.1094866
  AREA(I+264) = AREA(I+264) * 1.1094866
  AREA(I+456) = AREA(I+456) * 1.1094866
50 CONTINUE

BLADEAREA1 = PI * (X(553) * X(553) - PI * X(577) * X(577))
BLADEAREA2 = PI * (X(577) * X(577) - PI * 39.0 * 39.0)
AREA(553) = BLADEAREA1 / 24.0
AREA(577) = BLADEAREA2 / 24.0
DO 60 I = 1, 23
  J = I + 553
  K = I + 577
  AREA(J) = AREA(553)
  AREA(K) = AREA(577)
60 CONTINUE

DO 70 I = 1, 3
  ID6 = 501 + FLOAT(I-1) * 24
  DHDR = 98.0 / 10.0
  CONEK2 = SQRT(1 + DHDR * DHDR)
  CONEAREA = PI * CONEK2 * (X(ID6) * X(ID6) - X(ID6+24) * X(ID6+24))
  AREA(ID6) = CONEAREA / 24.0
DO 80 J = 1, 23
  AREA(ID6+J) = AREA(ID6)
80 CONTINUE

70 CONTINUE

DO 90 I = 673, 696
  AREA(I) = AREA(672)
  AREA(I+24) = AREA(648)
  AREA(I+48) = AREA(624)
90 CONTINUE

COVERAREA = 2 * PI * 19 * 39
DO 100 I = 745, 792
  AREA(I) = COVERAREA / 48.0
100 CONTINUE

OUTERCANAREA = PI * 90.0 * 2.0 * 17.0
DO 110 I = 793, 840
  AREA(I) = OUTERCANAREA / 24.0
110 CONTINUE

THE LOOP BELOW CALCULATES AREA OF THE TRAPEZOIDAL
TURBINE FRAME MEMBERS

DO 120 I = 841, 888
  AREA(I) = (((17.0 + 30.0) / 2.0) * (90.0 - 36.0))
120 CONTINUE

DO 130 I = 889, 936
  AREA(I) = 348.0
130 CONTINUE

THE FOLLOWING STEPS CALCULATE THE AREAS FOR THE
THREE FLAMEHOLDER ASSEMBLIES.

DHDR = 9.0 / 2.5
CONEK3 = SQRT(1 + DHDR * DHDR)
DIM10 = PI * CONEK3
FHAREA1 = DIM10 * (36.0 * 36.0 - 33.5 * 33.5)
FHAREA2 = DIM10 * (38.5 * 38.5 - 36.0 * 36.0)
FHAREA3 = DUM10*(77.0*77.0 - 74.5*74.5)
FHAREA4 = DUM10*(79.5*79.5 - 77.0*77.0)
FHAREA5 = DUM10*(114.0*114.0 - 111.5*111.5)
FHAREA6 = DUM10*(116.5*116.5 - 114.0*114.0)
DHDR2 = 24.0/41.0
CONEK4 = SQRT(1 + DHDR2*DHDR2)
DHDR3 = 5.0/37.0
CONEK5 = SQRT(1 + DHDR3*DHDR3)
DUM11 = PI*CONEK4
SUPPAREA1 = DUM11*(77.0*77.0 - 36.0*36.0)
DUM12 = PI*CONEK5
SUPPAREA2 = DUM12*(114.0*114.0 - 77.0*77.0)
DHDR4 = 1.732
CONEK6 = SQRT(1 + DHDR4*DHDR4)
DUM13 = PI*CONEK6
MIX1AREA = DUM13*(90.0*90.0 - 80.0*80.0)
MIX2AREA = DUM13*(100.0*100.0 - 90.0*90.0)
DO 140 I = 937,984
   AREA(I) = FHAREA1/24.0
   AREA(I+48) = FHAREA2/24.0
   AREA(I+96) = SUPPAREA1/60.0
   AREA(I+144) = FHAREA3/24.0
   AREA(I+192) = F(THREA4/24.0
   AREA(I+240) = SUPPAREA2/60.0
   AREA(I+288) = FHAREA5/24.0
   AREA(I+336) = FHAREA6/24.0
   AREA(I+384) = MIX1AREA/48.0
   AREA(I+432) = 173.2000000
   AREA(I+480) = 173.2000000
   AREA(I+528) = MIX2AREA/48.0
140 CONTINUE
DO 150 I = 1,22
   IDUM7 = 1512 +24*FLOAT(I-1)
   RAD1 = X(IDUM7+1)
   RAD2 = X(IDUM7+25)
   EXITAREA = PI*(RAD2*RAD2 - RAD1*RAD1)
   DO 160 J = 1,24
      AREA(IDUM7+J) = EXITAREA/24.0
150 CONTINUE
EXITAREA = PI*(X(I)*X(I) - X(2017)*X(2017))
DO 170 I = 2041,2064
      AREA(I) = EXITAREA/24.0
170 CONTINUE
RETURN
END

SUBROUTINE NUMSHOTS CALCULATES THE NUMBER OF SHOTS
THAT WILL BE EMMITED FROM A GIVEN SURFACE ELEMENT

SUBROUTINE NUMSHOTS (AREA, NSHOTS)
DIMENSION AREA(2064), NSHOTS(2064)

DO 10 I = 1,2064
   DUM1 = AREA(I)
   DUM2 = DUM1*40.0
   IDUM = INT(DUM2)
   NSHOTS(I) = IDUM
10 CONTINUE
RETURN
END

Appendix A ANSIRS Code Listing
SUBROUTINE NORMAL IS A SUBPROGRAM THAT CALCULATES THE
SURFACE NORMAL AT A POINT WITHIN THE GEOMETRY GIVEN X,Y,
AND Z COORDINATES AND THE ELEMENT NUMBER. IT THEN
RETURNS THE X,Y, AND Z COMPONENTS OF THE NORMAL VECTOR
A,B,C = X,Y,Z COORDINATES AT THE POINT OF INTEREST
IELEMENT = ELEMENT NUMBER (1-2064). THE SUBROUTINE ALSO
CALCULATES THE SURFACE TANGENT AT THE POINT. TOGETHER,
THE NORMAL, TANGENTIAL, AND CROSS PRODUCT OF THESE TWO
DEFINES THE LOCAL COORDINATE SYSTEM.

*NOTE THAT THE Z-COMPONENT OF THE SURFACE TANGENT IS
ALWAYS ZERO.

SUBROUTINE NORMAL(A,B,C,X,Y,Z,SECTION,XNORMAL,YNORMAL,
& ZNORMAL,PI,IELEMENT,XTANGENT,YTANGENT,ZTANGENT,
& XCROSS, YCROSS, ZCROSS)
DIMENSION X(2064), Y(2064), Z(2064)

REAL MAGNITUDE
REAL MAG
REAL PI
ZNORM = 0.0
ZTANGENT = 0.0
NDUM = 0
ELEMENT = FLOAT(IELEMENT)
IF (C.GT.465.0) THEN
  DDX21 = (X(25)-X(49))/(Z(25)-Z(49))
  DDX22 = (X(121)-X(145))/(Z(121)-Z(145))
  IF (ELEMENT.LE.96.0) ZNORM=DXD21
  IF (ELEMENT.GT.96.0) ZNORM=DXD22
  RING1 = ELEMENT/24.0
  IRING = IFIX(RING1)
  RING2 = FLOAT(IRING)
  DUMMY = ELEMENT - (24.0*RING2)
  NDUM = IELEMENT + 1
  IF (DUMMY.EQ.0.0) NDUM = IELEMENT - 23
  XYDST=SQR((X(NDUM)-X(IELEMENT))*X(NDUM)-
  & X(IELEMENT))+Y(NDUM)-
  & Y(IELEMENT))*X(NDUM)-Y(IELEMENT))/XYDST
  XTANGENT = (X(IELEMENT)-X(NDUM))/XYDST
  YTANGENT = (Y(IELEMENT)-Y(NDUM))/XYDST
  ANGLE = ((PI/180.0)*((DUMMY -1.0)*15)+7.5)
  XNORM = (-1.0)*COS(ANGLE)
  YNORM = (-1.0)*SIN(ANGLE)
  MAGNITUDE=SQR((XNORM*XNORM+YNORM*YNORM+ZNORM*ZNORM)
  XNORMAL = XNORM/MAGNITUDE
  YNORMAL = YNORM/MAGNITUDE
  ZNORMAL = ZNORM/MAGNITUDE
ENDIF
IF (C.LE.465.0.AND.ELEMENT.LE.792.0) THEN
  IF (ABS(A).LT.0.0001) A = 0.0001*(A/(ABS(A)))
  TANGLE = ATAN(B/A)
  IF (A.LT.0.0) TANGLE = TANGLE + PI
  XTANGENT = SIN(TANGLE)
  YTANGENT = (-1.0)*COS(TANGLE)
ENDIF
IF (C.LE.465.0.AND.C.GE.399.0) THEN
  ZNORM = 0.0
IF (C.GE.417.0.AND.C.LE.444.0) THEN
  ZNORM = -0.57243376*COS((PI/4.5)*(C-417.0))
ENDIF
DUM1 = SQR((A*A)+(B*B))
XNORM = (-1.0)*A/DUM1
YNORM = (-1.0)*B/DUM1
MAGNITUDE = SQRT(XNORM*XNORM + YNORM*YNORM + ZNORM*ZNORM)
XNORMAL = XNORM/MAGNITUDE
YNORMAL = YNORM/MAGNITUDE
ZNORMAL = ZNORM/MAGNITUDE
ENDIF
IF (C.LT.399.0 .AND. C.GE.332.0) THEN
  ZNORM = -0.214432835 - 0.57243276 * COS((PI/4.5)*(C-332.0))
  DUM2 = SQRT((A*A) + (B*B))
  XNORM = ((-1.0)*A)/DUM2
  YNORM = ((-1.0)*B)/DUM2
ENDIF
MAGNITUDE = SQRT(XNORM*XNORM + YNORM*YNORM + ZNORM*ZNORM)
XNORMAL = XNORM/MAGNITUDE
YNORMAL = YNORM/MAGNITUDE
ZNORMAL = ZNORM/MAGNITUDE
ENDIF
IF (C.GE.10.0 .AND. C.LE.7.0) THEN
  ZNORM = 0.0
ENDIF
IF (C.LE.147.0 .AND. C.GE.121.0) THEN
  ZNORM = -0.57243376 * COS((PI/4.5)*(C - 121.0))
ENDIF
DUM3 = SQRT((A*A) + (B*B))
XNORM = ((-1.0)*A)/DUM3
YNORM = ((-1.0)*B)/DUM3
MAGNITUDE = SQRT(XNORM*XNORM + YNORM*YNORM + ZNORM*ZNORM)
XNORMAL = XNORM/MAGNITUDE
YNORMAL = YNORM/MAGNITUDE
ZNORMAL = ZNORM/MAGNITUDE
ENDIF
IF (C.EQ.10.0 .AND. SECTION.EQ.11.0) THEN
  XNORMAL = 0.0
  YNORMAL = 0.0
  ZNORMAL = 1.0
ENDIF
IF (SECTION.EQ.8.0 .AND. C.LE.49) THEN
  ZNORM = ((C - 10.0)*(C - 10.0))/1521
  DUM4 = SQRT((A*A) + (B*B))
  XNORM = A/DUM4
  YNORM = B/DUM4
  MAG = SQRT((XNORM*XNORM) + (YNORM*YNORM) + (ZNORM*ZNORM))
  XNORMAL = XNORM/MAG
  YNORMAL = YNORM/MAG
  ZNORMAL = ZNORM/MAG
ENDIF
DXDZ3 = ( (X(625) - X(649)) / (Z(625) - Z(649)) )
IF (SECTION.EQ.9.0) THEN
  ZNORM = (-1.0)*DXDZ3
  DUM5 = SQRT((A*A) + (B*B))
  XNORM = ((-1.0)*A)/DUM5
  YNORM = ((-1.0)*B)/DUM5
  MAG = SQRT((XNORM*XNORM) + (YNORM*YNORM) + (ZNORM*ZNORM))
  XNORMAL = XNORM/MAG
  YNORMAL = YNORM/MAG
  ZNORMAL = ZNORM/MAG
ENDIF
IF (SECTION.EQ.10.0) THEN
  ZNORM = DXDZ3
  DUM6 = SQRT((A*A) + (B*B))
  XNORM = A/DUM6
  YNORM = B/DUM6
  MAG = SQRT((XNORM*XNORM) + (YNORM*YNORM) + (ZNORM*ZNORM))
  XNORMAL = XNORM/MAG
  YNORMAL = YNORM/MAG
  ZNORMAL = ZNORM/MAG
ENDIF
IF (ELEMENT.CE.793.0 .AND. ELEMENT.LE.840.0) THEN

Appendix A  ANSIRS Code Listing

105
ZNORMAL = 0.0
ZTANGENT = 0.0
IF (ABS(A).LT.0.0001) A = 0.0001*(A/(ABS(A)))
TANGLE = ATAN(B/A)
IF (A.LT.0.0) TANGLE = TANGLE + PI
XTANGENT = SIN(TANGLE)
YTANGENT = -1.0*COS(TANGLE)
DUM7 = SQRT(A*A + B*B)
IF (ELEMENT.LE.816.0) THEN
  XNORMAL = (-1.0)*A/DUM7
  YNORMAL = (-1.0)*B/DUM7
ENDIF
IF (ELEMENT.GT.816.0) THEN
  XNORMAL = A/DUM7
  YNORMAL = B/DUM7
ENDIF
THE NORMAL VECTORS FOR THE REHEATERS ARE IDENTICAL TO
THE THOSE OF THE TURBINE FRAME, THUS
IF (ELEMENT.GE.889.0.AND.ELEMENT.LE.936.0) THEN
  ELEMENT=ELEMENT-48.0
ENDIF
IF (ELEMENT.GE.841.0.AND.ELEMENT.LE.888.0) THEN
  ZNORMAL = 0.0
  DUM8 = SQRT(A*A + B*B)
  XTANGENT = A/DUM8
  YTANGENT = B/DUM8
  A = 0.0001*(A/(ABS(A)))
  TANGLE = ATAN(B/A)
  IF (A.LT.0.0) TANGLE = TANGLE + PI
  IF (ELEMENT.LE.864.0) THEN
    XNORMAL = -1.0*SIN(TANGLE)
    YNORMAL = COS(TANGLE)
  ENDIF
ENDIF
IF (ELEMENT.GT.864.0) THEN
  XNORMAL = SIN(TANGLE)
  YNORMAL = (-1.0)*COS(TANGLE)
ENDIF
ENDIF
IF (ELEMENT.GE.937.0.AND.ELEMENT.LE.1512.0) THEN
  ZNORMAL = -0.529919264
  XNORMAL = -1.0*A/DUM9
  YNORMAL = -1.0*B/DUM9
ENDIF
IF (ELEMENT.GE.961.0.AND.ELEMENT.LE.984.0) THEN
  XTANGENT = -1.0*SIN(TANGLE)
  YTANGENT = COS(TANGLE)
  DUM9 = SQRT(A*A + B*B)
ENDIF
IF (ELEMENT.GE.985.0.AND.ELEMENT.LE.1008.0) THEN
  ZNORMAL = 0.529919264
  XNORMAL = -1.0*A/DUM9
ENDIF
Appendix A  ANSIRS Code Listing  106
YNORM = -1.0*B/DUM9
ENDIF
IF (ELEMENT.GE.1009.0.AND.ELEMENT.LE.1032.0) THEN
  ZNORM=-0.529919264
  XNORM = A/DUM9
  YNORM = B/DUM9
ENDIF
IF (ELEMENT.GE.1033.0.AND.ELEMENT.LE.1056.0) THEN
  ZNORM=-0.863014469
  XNORM = -1.0*A/DUM9
  YNORM = -1.0*B/DUM9
ENDIF
IF (ELEMENT.GE.1057.0.AND.ELEMENT.LE.1080.0) THEN
  ZNORM=0.863014469
  XNORM = A/DUM9
  YNORM = B/DUM9
ENDIF
IF (ELEMENT.GE.1081.0.AND.ELEMENT.LE.1104.0) THEN
  ZNORM=-0.529919264
  XNORM = -1.0*A/DUM9
  YNORM = -1.0*B/DUM9
ENDIF
IF (ELEMENT.GE.1105.0.AND.ELEMENT.LE.1128.0) THEN
  ZNORM=0.529919264
  XNORM = A/DUM9
  YNORM = B/DUM9
ENDIF
IF (ELEMENT.GE.1129.0.AND.ELEMENT.LE.1152.0) THEN
  ZNORM = 0.529919264
  XNORM = -1.0*A/DUM9
  YNORM = -1.0*B/DUM9
ENDIF
IF (ELEMENT.GE.1153.0.AND.ELEMENT.LE.1176.0) THEN
  ZNORM=-0.529919264
  XNORM = A/DUM9
  YNORM = B/DUM9
ENDIF
IF (ELEMENT.GE.1177.0.AND.ELEMENT.LE.1200.0) THEN
  ZNORM=-0.99099243
  XNORM = A/DUM9
  YNORM = B/DUM9
ENDIF
IF (ELEMENT.GE.1201.0.AND.ELEMENT.LE.1224.0) THEN
  ZNORM = 0.99099243
  XNORM = -1.0*A/DUM9
  YNORM = -1.0*B/DUM9
ENDIF
IF (ELEMENT.GE.1225.0.AND.ELEMENT.LE.1248.0) THEN
  ZNORM=-0.529919264
  XNORM = -1.0*A/DUM9
  YNORM = -1.0*B/DUM9
ENDIF
IF (ELEMENT.GE.1249.0.AND.ELEMENT.LE.1272.0) THEN
  ZNORM=0.529919264
  XNORM = A/DUM9
  YNORM = B/DUM9
ENDIF
IF (ELEMENT.GE.1273.0.AND.ELEMENT.LE.1296.0) THEN
  ZNORM = 0.529919264
  XNORM = -1.0*A/DUM9
  YNORM = -1.0*B/DUM9
ENDIF
IF (ELEMENT.GE.1297.0.AND.ELEMENT.LE.1320.0) THEN
  ZNORM=-0.529919264
  XNORM = A/DUM9

Appendix A  ANSIRS Code Listing
YNORM = B/DUM9
ENDIF
IF (ELEMENT.GE.1321.0.AND.ELEMENT.LE.1344.0) THEN
ZNORM = -0.5
XNORM = -1.0*A/DUM9
YNORM = -1.0*B/DUM9
ENDIF
IF (ELEMENT.GE.1345.0.AND.ELEMENT.LE.1368.0) THEN
ZNORM = 0.5
XNORM = A/DUM9
YNORM = B/DUM9
ENDIF
IF (ELEMENT.GE.1369.0.AND.ELEMENT.LE.1464.0) THEN
XTANGENT = A/DUM9
YTANGENT = B/DUM9
IF (ELEMENT.LE.1392.0) THEN
XNORM = SIN(TANGLE)
YNORM = -1.0*COS(TANGLE)
ENDIF
IF (ELEMENT.GE.1393.0.AND.ELEMENT.LE.1416.0) THEN
XNORM = -1.0*SIN(TANGLE)
YNORM = COS(TANGLE)
ENDIF
IF (ELEMENT.GE.1417.0.AND.ELEMENT.LE.1440.0) THEN
XNORM = SIN(TANGLE)
YNORM = -1.0*COS(TANGLE)
ENDIF
IF (ELEMENT.GE.1441.0) THEN
XNORM = -1.0*SIN(TANGLE)
YNORM = COS(TANGLE)
ENDIF
ENDIF
IF (ELEMENT.GE.1465.0.AND.ELEMENT.LE.1488.0) THEN
ZNORM = 0.5
XNORM = -1.0*A/DUM9
YNORM = -1.0*B/DUM9
ENDIF
IF (ELEMENT.GE.1489.0.AND.ELEMENT.LE.1512.0) THEN
ZNORM = -0.5
XNORM = A/DUM9
YNORM = B/DUM9
ENDIF
MAG = SQRT(XNORM*XNORM + YNORM*YNORM + ZNORM*ZNORM)
XNORMAL = XNORM/MAG
YNORMAL = YNORM/MAG
ZNORMAL = ZNORM/MAG
ENDIF
IF (SECTION.EQ.42.0.AND.ELEMENT.LE.1512.0) THEN
ZNORMAL = -1.0
XNORMAL = 0.0
YNORMAL = 0.0
ENDIF

NOW I HAVE THE NORMAL AND TANGENTIAL VECTORS OF A LOCAL
COORDINATE SYSTEM. NEXT I MUST TAKE THE CROSS PRODUCT OF
THESE TWO TO DETERMINE A THIRD AXIS OF THE LOCAL SYSTEM.

A1 = XNORMAL
A2 = YNORMAL
A3 = ZNORMAL
B1 = XTANGENT
B2 = YTANGENT
B3 = ZTANGENT
XCROSS = (A2*B3) - (B2*A3)
YCROSS = (-1.0)*((A1*B3) - (B1*A3))

Appendix A  ANSIRS Code Listing 108
ZCROSS = (A1*B2) - (B1*A2)
RETURN
END

SUBROUTINE CHAN(CE) GENERATES A RANDOM NUMBER BETWEEN 0 AND 1 EACH SUCCESSIVE TIME IT IS CALLED (SEED MUST BE PASSED)

SUBROUTINE CHAN(SEED, RND)

INTEGER SEED

RND = 0.0
SEED = 2045*SEED + 1
SEED = SEED - (SEED/1048576)*1048576
RND = REAL(SEED + 1)/1048577.0
RETURN
END

SUBROUTINE SOURCEPT GENERATES RANDOM POINTS ON A GIVEN SURFACE ELEMENT. THESE POINTS ARE TO BE THE POINTS OF RANDOM EMISSIONS AND THE ORIGIN OF RAYS TO BE TRACED.

SUBROUTINE SOURCEPT(X,Y,Z,
& SEED, NSHOTS, PI, SECTION, NUM, N, AREA,
& IELEMENT, XI, Yi, ZI, NOZSET, L, ZSTART, ZMAX, RND2)

INTEGER SEED
REAL NOZSET
REAL PI
DIMENSION X(2064), Y(2064), Z(2064)
DIMENSION NSHOTS(2064), AREA(2064)

ELEMENT = FLOAT(IELEMENT)
FLAG = 0.0
IF (ELEMENT.LE.96.0) SECTION = 1.0
IF (ELEMENT.GT.96.0.AND.ELEMENT.LE.168.0) SECTION = 2.0
IF (ELEMENT.GT.168.0.AND.ELEMENT.LE.240.0) SECTION = 3.0
IF (ELEMENT.GT.240.0.AND.ELEMENT.LE.288.0) SECTION = 4.0
IF (ELEMENT.GT.288.0.AND.ELEMENT.LE.456.0) SECTION = 5.0
IF (ELEMENT.GT.456.0.AND.ELEMENT.LE.480.0) SECTION = 6.0
IF (ELEMENT.GT.480.0.AND.ELEMENT.LE.552.0) SECTION = 7.0
IF (ELEMENT.GT.552.0.AND.ELEMENT.LE.600.0) SECTION = 11.0
IF (ELEMENT.GT.600.0.AND.ELEMENT.LE.672.0) SECTION = 9.0
IF (ELEMENT.GT.672.0.AND.ELEMENT.LE.744.0) SECTION = 10.0
IF (ELEMENT.GT.744.0.AND.ELEMENT.LE.792.0) SECTION = 8.0
IF (ELEMENT.GT.792.0.AND.ELEMENT.LE.816.0) SECTION = 12.0
IF (ELEMENT.GT.816.0.AND.ELEMENT.LE.840.0) SECTION = 13.0
IF (ELEMENT.GT.840.0.AND.ELEMENT.LE.864.0) SECTION = 14.0
IF (ELEMENT.GT.864.0.AND.ELEMENT.LE.888.0) SECTION = 15.0
IF (ELEMENT.GT.888.0.AND.ELEMENT.LE.912.0) SECTION = 16.0
IF (ELEMENT.GT.912.0.AND.ELEMENT.LE.936.0) SECTION = 17.0
IF (ELEMENT.GT.936.0.AND.ELEMENT.LE.960.0) SECTION = 18.0
IF (ELEMENT.GT.960.0.AND.ELEMENT.LE.984.0) SECTION = 19.0
IF (ELEMENT.GT.984.0.AND.ELEMENT.LE.1008.0) SECTION = 20.0
IF (ELEMENT.GT.1008.0.AND.ELEMENT.LE.1032.0) SECTION = 21.0
IF (ELEMENT.GT.1032.0.AND.ELEMENT.LE.1056.0) SECTION = 22.0
IF (ELEMENT.GT.1056.0.AND.ELEMENT.LE.1080.0) SECTION = 23.0
IF (ELEMENT.GT.1080.0.AND.ELEMENT.LE.1104.0) SECTION = 24.0
IF (ELEMENT.GT.1104.0.AND.ELEMENT.LE.1128.0) SECTION = 25.0
IF (ELEMENT.GT.1128.0.AND.ELEMENT.LE.1152.0) SECTION = 26.0
IF (ELEMENT.GT.1152.0.AND.ELEMENT.LE.1176.0) SECTION = 27.0
IF (ELEMENT.GT.1176.0.AND.ELEMENT.LE.1200.0) SECTION = 28.0
IF (ELEMENT.GT.1200.0.AND.ELEMENT.LE.1224.0) SECTION= 29.0
IF (ELEMENT.GT.1224.0.AND.ELEMENT.LE.1248.0) SECTION= 30.0
IF (ELEMENT.GT.1248.0.AND.ELEMENT.LE.1272.0) SECTION= 31.0
IF (ELEMENT.GT.1272.0.AND.ELEMENT.LE.1296.0) SECTION= 32.0
IF (ELEMENT.GT.1296.0.AND.ELEMENT.LE.1320.0) SECTION= 33.0
IF (ELEMENT.GT.1320.0.AND.ELEMENT.LE.1344.0) SECTION= 34.0
IF (ELEMENT.GT.1344.0.AND.ELEMENT.LE.1368.0) SECTION= 35.0
IF (ELEMENT.GT.1368.0.AND.ELEMENT.LE.1392.0) SECTION= 36.0
IF (ELEMENT.GT.1392.0.AND.ELEMENT.LE.1416.0) SECTION= 37.0
IF (ELEMENT.GT.1416.0.AND.ELEMENT.LE.1440.0) SECTION= 38.0
IF (ELEMENT.GT.1440.0.AND.ELEMENT.LE.1464.0) SECTION= 39.0
IF (ELEMENT.GT.1464.0.AND.ELEMENT.LE.1488.0) SECTION= 40.0
IF (ELEMENT.GT.1488.0.AND.ELEMENT.LE.1512.0) SECTION= 41.0
I = I+1
J = 1
R = 0
C INITIALIZING R AND J AS PRECAUTIONS
CALL CHAN(SEED,RND)
RND1 = RND
CALL CHAN(SEED,RND)
RND2 = RND
IF (SECTION.LE.2.0) THEN
DUM01 = ELEMENT/24.0
DUM01 = IFIX(DUM01)
DUM02 = FLOAT(DUM01)
DUM03 = 24.0*DUM02
DUM04 = ELEMENT - DUM03
DUM05 = AREA(I)
IF (DUM05.GE.0.0) J = J - 1
IF (DUM05.LE.0.0) J = J + 1
CALL FINDLENGTH(X(I),Y(I),Z(I),X(J),Y(J),Z(J),BASE1)
CALL FINDLENGTH(X(I+24),Y(I+24),Z(I+24),
& X(J+24),Y(J+24),Z(J+24),BASE2)
CALL FINDLENGTH(X(I),Y(I),Z(I),
& X(I+24),Y(I+24),Z(I+24),DELTZ)
IF (BASE1.GE.BASE) THEN
DUM06 = BASE1 - BASE2 + 0.000001
DUM07 = DELTZ/(2.0*DUM06*DUM05)
DUM08 = ((DUM07*BASE1*BASE1) - RND2)/DUM07
BASE3 = SQRT(DUM08)
DUM09 = (BASE1-BASE3)/DUM06
ZI = Z(I) - DUM09*(Z(I) - Z(I+24))
ENDIF
IF (BASE2.GT.BASE1) THEN
DUM06 = BASE2 - BASE1
DUM07 = DELTZ/(2.0*DUM06*DUM05)
DUM08 = ((DUM07*BASE2*BASE2) - RND2)/DUM07
BASE3 = SQRT(DUM08)
DUM09 = (BASE2-BASE3)/DUM06
ZI = Z(I+24) - DUM09*(Z(I) - Z(I+24))
ENDIF
SCALE =ZI/(Z(I)-Z(I+24))
XITEMP = X(I) - SCALExI - X(I+24))
XJTEMP = X(J) - SCALexJ - X(J+24))
YITEMP = Y(I) - SCALeyI - Y(I+24))
YJTEMP = Y(J) - SCALeyJ - Y(J+24))
XTMP = RND1*XYJTEMP + XITEMP
YTMP = RND1*YJTEMP + YITEMP
CALL FINDLENGTH(XTMP,HTEMP,1,XTEMP,YITEMP,BASE4)
XI = X'TMP
YI = Y'TMP
ENDIF
IF (SECTION.EQ.4.0.OR.SECTION.EQ.10.0.
& OR.SECTION.EQ.9.0) THEN
R2 = 0.0

Appendix A  ANSIRS Code Listing  110
FLAG = 1.0
DUM1 = ELEMENT/24.0
DUM2 = IFIX(DUM1)
DUM3 = 24.0*DUM2
DUM4 = ELEMENT - DUM3
IF (DUM4.EQ.0.0) J = I - 23
IF (DUM4.NE.0.0) J = I + 1
R1 = SQRT(X(I)*X(I) + Y(I)*Y(I))
R01 = R1
R3 = SQRT(X(I+24)*X(I+24) + Y(I+24)*Y(I+24))
DELTA2 = Z(I) - Z(I+24)/(R1 - R3)
IF (R3.GT.R1) THEN
  R2 = R1
  F1 = R3
ENDIF
IF (R1.GT.R3) THEN
  R2 = R3
ENDIF
R4 = SQRT(RND2*(R1*R1 - R2*R2) + R2*R2)
DUM5 = (R01 - R4)
ZI = Z(I) - DUM5*DELTA2
ENDIF
IF (SECTION.EQ.3.0.OR.SECTION.LE.7.0.AND.SECTION.GE.5.0) THEN
  ZI = RND2*((Z(I+24) - Z(I))) + Z(I)
ENDIF
IF (SECTION.EQ.8.0) THEN
  IF (ELEMENT.LE.768.0) ZI = 29.5 - ((RND2)*19.5)
  IF (ELEMENT.GT.768.0) ZI = 49.0 - ((RND2)*19.5)
ENDIF
IF (SECTION.GT.2.0.AND.SECTION.LE.11.0) THEN
  IF (SECTION.GT.2.0.AND.SECTION.LE.10.0) THEN
    ZVAL = ZI
    CALL FINDRADIUS(ZVAL,NOZSET,L,R,SECTION,PI,ZSTART)
  ENDIF
ENDIF
IF (SECTION.EQ.11.0) THEN
  ZI = 10.0
  D01 = X(I+24)
  D02 = Y(I+24)
  D08 = (X(I)*X(I) + Y(I)*Y(I)) - (X(I+24)*X(I+24) + Y(I+24)*Y(I+24))
  R = SQRT((D01*D01 + D02*D02) + (RND2)*(D08))
ENDIF
DUM8 = ELEMENT/24.0
DUM9 = IFIX(DUM8)
DUM10 = FLOAT(DUM9)
DUM11 = ELEMENT - (24.0*DUM10)
DUM12 = DUM11 - 1
XI = R*COS(((DUM12*15*PI)/180) + ((RND1*15*PI)/180))
YI = R*SIN(((DUM12*15*PI)/180) + ((RND1*15*PI)/180))
ENDIF
IF (SECTION.EQ.12.0.OR.SECTION.EQ.13.0) THEN
  ZI = 10.0 + RND2*17.0
  R = 90.0
  DUM13 = ELEMENT/24.0
  IDUM10 = IFIX(DUM13)
  DUM14 = FLOAT(IDUM10)
  DUM15 = ELEMENT - (24.0*DUM14)
  DUM16 = DUM15 - 1.0
  XI = R*COS(((DUM16*15.0*PI)/180.0) + ((RND1*15*PI)/180.0))
  YI = R*COS(((DUM16*15.0*PI)/180.0) + ((RND1*15*PI)/180.0))
ENDIF

Appendix A ANSIRS Code Listing
DUM17 = ELEMENT/24.0
IDUM11 = IFIX(DUM17)
DUM18 = FLOAT(IDUM11)
DUM19 = ELEMENT - (24.0*DUM18)
DUM20 = DUM19 - 1.0
IF (SECTION.EQ.14.0.OR.SECTION.EQ.15.0) THEN
  R = 39.0+117.6921077-
& 0.5*SQRRT(55405.91717-(37614.46154*RND1))
  ZI = 10.0 + RND2*(17.0 + 13.0*((90.0 - R)/51.0))
  XI = R*COS(DUM20*15.0*PI/180.0)
  YI = R*SIN(DUM20*15.0*PI/180.0)
ENDIF

IF (SECTION.EQ.16.0.OR.SECTION.EQ.17.0) THEN
  ZI = 50.0 + RND2*12.0
  R = 51.0 + 29.0*RND1
  XI = R*COS(DUM20*15.0*PI/180.0)
  YI = R*SIN(DUM20*15.0*PI/180.0)
ENDIF

IF (SECTION.EQ.18.0.OR.SECTION.EQ.19.0) THEN
  R = SQRRT(RND2*(36.0*36.0 - 33.5*33.5) + 33.5*33.5)
  ZI = 118.0 + ((36.0-R)/2.5)*9.0
  XI = R*
& (COS(((DUM20*15*PI)/180.0)+((RND1*15*PI)/180.0)))
  YI = R*
& (SIN(((DUM20*15*PI)/180.0)+((RND1*15*PI)/180.0)))
ENDIF

IF (SECTION.EQ.20.0.OR.SECTION.EQ.21.0) THEN
  R = SQRRT(RND2*(38.5*38.5 - 36.0*36.0) +36.0*36.0)
  ZI = 118.0 + ((R-36.0)/2.5)*9.0
  XI = R*
& (COS(((DUM20*15*PI)/180.0)+((RND1*15*PI)/180.0)))
  YI = R*
& (SIN(((DUM20*15*PI)/180.0)+((RND1*15*PI)/180.0)))
ENDIF

IF (SECTION.EQ.22.0.OR.SECTION.EQ.23.0) THEN
  R = SQRRT(RND2*(77.0*77.0 -36.0*36.0) +36.0*36.0)
  ZI = 94.0 + ((77.0 - R)/41.0)*24.0
  XI = R*
& (COS(((DUM20*15*PI)/180.0)+((RND1*6*PI)/180.0)))
  YI = R*
& (SIN(((DUM20*15*PI)/180.0)+((RND1*6*PI)/180.0)))
ENDIF

IF (SECTION.EQ.24.0.OR.SECTION.EQ.25.0) THEN
  R = SQRRT(RND2*(77.0*77.0 - 74.5*74.5) + 74.5*74.5)
  ZI = 94.0 + ((77.0 - R)/2.5)*9.0
  XI = R*
& (COS(((DUM20*15*PI)/180.0)+((RND1*15*PI)/180.0)))
  YI = R*
& (SIN(((DUM20*15*PI)/180.0)+((RND1*15*PI)/180.0)))
ENDIF

IF (SECTION.EQ.26.0.OR.SECTION.EQ.27.0) THEN
  R = SQRRT(RND2*(79.5*79.5 - 77.0*77.0) + 77.0*77.0)
  ZI = 94.0 + ((R - 77.0)/2.5)*9.0
  XI = R*
& (COS(((DUM20*15*PI)/180.0)+((RND1*15*PI)/180.0)))
  YI = R*
& (SIN(((DUM20*15*PI)/180.0)+((RND1*15*PI)/180.0)))
ENDIF

IF (SECTION.EQ.28.0.OR.SECTION.EQ.29.0) THEN
  R = SQRRT(RND2*(114.0*114.0 - 77.0*77.0) + 77.0*77.0)
  ZI = 94.0 + ((R - 77.0)/37.0)*5.0
  XI = R*
& (COS(((DUM20*15*PI)/180.0)+((RND1*6*PI)/180.0)))
  YI = R*
& (SIN(((DUM20*15*PI)/180.0)+((RND1*6*PI)/180.0)))
ENDIF
IF (SECTION.EQ.30.0.OR.SECTION.EQ.31.0) THEN
  R = SQRT(RND2* (114.0*114.0 - 111.5*111.5) + 111.5*111.5)
  ZI = 99.0 + ((114.0 - R)/2.5)*9.0
  XI = R*
  & (COS(((DUM20*15*PI)/180.0)+((RND1*15*PI)/180.0))
  YI = R*
  & (SIN(((DUM20*15*PI)/180.0)+((RND1*15*PI)/180.0))
ENDIF
IF (SECTION.EQ.32.0.OR.SECTION.EQ.33.0) THEN
  R = SQRT(RND2* (116.5*116.5 - 114.0*114.0) + 114.0*114.0)
  ZI = 99.0 + ((R - 114.0)/2.5)*9.0
  XI = R*
  & (COS(((DUM20*15*PI)/180.0)+((RND1*15*PI)/180.0))
  YI = R*
  & (SIN(((DUM20*15*PI)/180.0)+((RND1*15*PI)/180.0))
ENDIF
IF (SECTION.EQ.34.0.OR.SECTION.EQ.35.0) THEN
  R = SQRT(RND2*(90.0*90.0 - 80.0*80.0) + 80.0*80.0)
  ZI = 44.32 - ((R - 80.0)/10.0)*17.32
  XI = R*
  & (COS(((DUM20*15.0*PI)/180.0)+((RND1*7.5*PI)/180.0))
  YI = R*
  & (SIN(((DUM20*15.0*PI)/180.0)+((RND1*7.5*PI)/180.0))
ENDIF
IF (SECTION.EQ.40.0.OR.SECTION.EQ.41.0) THEN
  R = SQRT(RND2*(100.0*100.0 - 90.0*90.0) + 90.0*90.0)
  ZI = 27.0 + 17.32* ((R-90.0)/10.0)
  XI = R*
  & COS(((DUM20*15+7.5)*PI)/180.0+((RND1*7.5*PI)/180.0))
  YI = R*
  & SIN(((DUM20*15+7.5)*PI)/180.0+((RND1*7.5*PI)/180.0))
ENDIF
IF (SECTION.EQ.36.0.OR.SECTION.EQ.37.0) THEN
  ZI = 27.0 + 17.32*SQRT(RND2)
  R=80.0+
  & 10* ((44.32-ZI)/17.32)+RND1*20.0* ((ZI-27.0)/17.32)
  XI = R*COS(((DUM20*15.0+7.5)*PI)/180.0)
  YI = R*SIN(((DUM20*15.0+7.5)*PI)/180.0)
ENDIF
IF (SECTION.EQ.38.0.OR.SECTION.EQ.39.0) THEN
  ZI = 27.0 + 17.32*SQRT(RND2)
  R=80.0+
  & 10* ((44.32-ZI)/17.32)+RND1*20.0* ((ZI-27.0)/17.32)
  XI = R*COS(((DUM20+1.0)*15.0)*PI/180.0)
  YI = R*SIN(((DUM20+1.0)*15.0)*PI/180.0)
ENDIF
RETURN
END

SUBROUTINE DIRECTION(SEQU,THET,PHI,PI)
INTEGER SEED
CALL CHAN(SEQU,RND)
THET = ASIN(SQRT(RND))
CALL CHAN(SEQU,RND)
PH = 2.0*PI*RND
RETURN
END

SUBROUTINE QUADRATIC SOLVES THE QUADRATIC EQUATION WITH

Appendix A  ANSIRS Code Listing
COEFFICIENTS A, B AND C AND RETURNS TWO ROOTS T1 AND T2

SUBROUTINE QUADRATIC(A, B, C, T1, T2)

IF (A.EQ.0.0) A = 0.0000001
D1 = (B*B - (4.0*A*C))
IF (D1.LT.0.0) THEN

IF THE NUMBER UNDER THE RADICAL IS <ZERO THE ANSWER IS
IMAGINARY AND DOES US NO GOOD

    T1 = 0.0
    T2 = 0.0
    GOTO 10
ENDIF
DUM = SQRT(D1)
T1 = -1.0*(B/(2.0*A)) + DUM/(2.0*A)
T2 = -1.0*(B/(2.0*A)) - DUM/(2.0*A)
10 RETURN
END

THIS SUBROUTINE MAY BE USED TO FIND THE INTERSECTION
POINT OF ANY LINE AND A CONE OF KNOWN PROPERTIES

SUBROUTINE CONESEARCH(XIN, YIN, ZIN, XRAY, YRAY, ZRAY, Q, &
                        Z0, T1, T2)

    ADUM = ZRAY*ZRAY - Q*Q*XRAY*XRAY - Q*Q*YRAY*YRAY
    BDUM=2.0*(ZIN*ZRAY-ZRAY*Z0-Q*Q*XRAY-Q*Q*YIN*YRAY)
    CDUM=ZIN*ZIN-2.0*ZIN*Z0+Z0*Z0-Q*Q*XIN*XIN-Q*Q*YIN*YIN
    CALL QUADRATIC(ADUM, BDUM, CDUM ,T1, T2)
RETURN
END

SUBROUTINE INTERSECT(X, Y, Z, XI, YI, ZI, XNORMAL, YNORMAL, &
                      ZNORMAL, XTANGENT, YTANGENT, ZTANGENT, SECTION, &
                      PI, N, NUM, THETA, PHI, IELEMENT, NOZSET, &
                      ISURF, XJ, YJ, ZJ, ZMAX, ZSTART, XCROSS, &
                      ZCROSS, ZCROSS, AREA, M, ALP, XAP, SEED, &
                      XPLANE, YPLANE, ZPLANE, U, V, W, ZRAY, COUNT, &
                      MSURF)

THIS SUBROUTINE TAKES THE POINT OF ORIGIN AND THE RANDOM
DIRECTIONS OF EMISSION AND RETURNS THE INTERSECTION POINT
AND SURFACE ELEMENT NUMBER. XI, YI, AND ZI IS THE POINT
OF EMISSION (FROM SURFACE M). XJ, YJ, ZJ ARE THE COORDINATES
OF INTERSECTION WITH THE SURFACE. ISURF IS THE ELEMENT
NUMBER OF INTERSECTION.

DIMENSION X(2064), Y(2064), Z(2064)
DIMENSION XPLANE(1500), YPLANE(1500), ZPLANE(1500)
DIMENSION DF(1512,2064), ARRA(2064), ALP(2064)
DIMENSION XAP(2064), U(1500), V(1500), W(1500)
REAL NOZSET
REAL PI
INTEGER SEED
COMMON /BLOCK1/DF
M = MSURF
03 TACTUAL = 0.0
ZJNN = 0.0
CALL NORMAL(XI, YI, ZI, X, Y, Z, SECTION, XNORMAL, YNORMAL, &
            ZNORMAL, PI, M, XTANGENT, YTANGENT, ZTANGENT, &
            XCROSS, YCROSS, ZCROSS)
XRAY, YRAY, AND ZRAY ARE THE X, Y, AND Z COMPONENTS OF THE
EMITTED RAY, WHICH I Normalize TO A MAGNITUDE OF ONE.
XLOCAL, YLOCAL, AND ZLOCAL ARE THE COMPONENTS OF THE
EMITTED RAY EXPRESSED IN THE LOCAL COORDINATE SYSTEM.
XLOCAL CORRESPONDS TO THE SURFACE NORMAL, YLOCAL TO THE
SURFACE TANGENT AND ZLOCAL TO THE RAY'S COMPONENT IN
THE 'CROSS' DIRECTION (NORMAL X TANGENT = CROSS).

XLOCAL = COS(THETA)
YLOCAL = (SIN(THETA))
ZLOCAL = (SIN(PHI)*SIN(THETA))
XRAY = XLOCAL*XNORMAL + YLOCAL*XTANGENT + ZLOCAL*XCROSS
YRAY = XLOCAL*YNORMAL + YLOCAL*YTANGENT + ZLOCAL*YCROSS
ZRAY = XLOCAL*ZNORMAL + YLOCAL*ZTANGENT + ZLOCAL*ZCROSS
DUM35 = SQRT(XRAY*XRAY + YRAY*YRAY + ZRAY*ZRAY)
XRAY = XRAY/DUM35
YRAY = YRAY/DUM35
ZRAY = ZRAY/DUM35

NOW I BEGIN THE SEARCH TO FIND WHERE THE RAY INTERSECTS
PORTIONS OF THE ENGINE. MY SEARCHING HIERARCHY IS
DESIGNED SO THAT THE SECTIONS WHICH ARE EASY SEARCHED
AND THE FIRST TIME INTENSIVE ARE CONDUCTED FIRST IF
POSSIBLE. THE FIRST STEP IS TO CHECK WHETHER OR NOT THE
RAY HAS LANDED IN THE 'OPEN' PORTION OF THE ENGINE
CAVITY (ONLY ONE POSSIBLE RADIUS OR SECTION FOR EACH Z-
LOCATION). THIS IS WHAT THE NEXT SECTION ACCOMPLISHES.
THE 'FLAG' TERMS INITIALIZED BELOW HELP DETERMINE WHICH
SIDE OF A TWO SIDED GEOMETRY ARE INTERSECTED.

CONTINUE
TACTUAL = 0.0
ZJNN = 0.0
FLAG2 = 0.0
FLAG21 = 0.0
FLAG22 = 0.0
FLAG23 = 0.0
FLAG24 = 0.0
FLAG25 = 0.0
FLAG26 = 0.0
FLAG27 = 0.0
FLAG32 = 0.0
FLAG42 = 0.0
FLAG91 = 0.0
FLAG92 = 0.0
ZFLAG93 = 0.0
XIN = XI
YIN = YI
ZIN = ZI
FLAG55 = 0.0

THE FIRST STEP IN THE SEARCH IS SEARCHING THE 'CLEAN'
SECTIONS FIRST SO THAT I MAY DETERMINE WHETHER OR NOT I
MUST LOOK AT THE 'MESSY' SECTIONS AT ALL. TO THAT END,
I FIND THE RAY'S INTERSECTION WITH THE LARGEST 'CLEAN'
SHELL PORTION (SECTION 5) TO DETERMINE IF THE RAY WENT
INTO THE THAT PORTION. FIRST I SOLVE THE INTERSECTION OF
THE LINE WITH THE CYLINDER TO DETERMINE TACTUAL. TACTUAL
IS A PARAMETER THAT SCALES THE POINT OF EMISSION TO THE
INTERSECTION POINT. IT COMES FROM THE PARAMETRIC LINE
EQUATIONS:

XJ = XI + TACTUAL*(XRAY)
YJ = YI + TACTUAL*(YRAY)
ZJ = ZI + TACTUAL*(ZRAY)
A = XRAY*XRAY + YRAY*YRAY
B = 2.0*(XIN*XRAY + YIN*YRAY)
C = (XIN*XIN) + (YIN*YIN) - 14520.25
CALL QUADRATIC(A, B, C, T1, T2)
TTEMP = T1
IF (T1.LT.0.0.AND.T2.GT.0.0) TTEMP = T2
IF (T2.LT.0.0.AND.T1.GT.0.0) TTEMP = T1
IF (T1.GT.0.0.AND.T2.GT.0.0.AND.T1.LT.T2) TTEMP = T2
IF (T1.GT.0.0.AND.T2.GT.0.0.AND.T1.GT.T2) TTEMP = T1
ZJNT = ZIN + TTEMP*ZRAY
IF (TTEMP.GT.0.01) THEN
  IF (ZJNT.GE.10.00.AND.ZJNT.LE.332.0) THEN
    TACTUAL = TTEMP
  ENDIF
ENDIF

IF WE GET TO THIS POINT, WE KNOW THAT THE POINT OF
INTERSECTION WITH THE 'CLEAN' PART OF THE ENGINE CAVITY
LIES PAST A Z-VALUE OF 332 SO WE MAY BEGIN TO SEARCH THE
OTHER SECTIONS. THE NEXT SECTION TO SEARCH IS THE
EXHAUST NOZZLE EXIT.

CALL FINDRADIUS(ZMAX, NOZSET, L, R, 1.0, PI, ZSTART)
ZJNNTMP = ZMAX
IF (ZRAY.EQ.0.0) ZRAY = 0.000001
TDUM = (ZMAX-ZIN)/ZRAY
XJNT = XIN + TDUM*XRAY
YJNT = YIN + TDUM*YRAY
RDUM = SQRT(XJNT*XJNT + YJNT*YJNT)
IF (RDUM.LE.R) THEN
  IF (TDUM.GT.0.01) THEN
    IF (TDUM.LT.TACTUAL.OR.TACTUAL.EQ.0.0) THEN
      A = XRAY*XRAY + YRAY*YRAY + ZRAY*ZRAY
      B = 2.0*(XIN*XRAY+YIN*YRAY+ZIN*ZRAY-ZMAX*ZRAY)
      C =-XIN*XIN-
      & YIN*YIN-ZIN-ZIN-2.0*ZMAX*ZIN+((ZMAX*ZMAX)-140625.0)
      CALL QUADRATIC(A, B, C, T1, T2)
      TTEMP = 0.0
      IF (T1.GT.0.0.AND.T2.LT.0.0) TTEMP = T1
      IF (T2.LT.0.0.AND.T2.GT.0.0) TTEMP = T2
      ZT1 = ZIN+T1*ZRAY
      ZT2 = ZIN+T2*ZRAY
      IF (ZT1.GE.ZMAX.AND.T1.GT.0.0) TTEMP = T1
      IF (ZT2.GE.ZMAX.AND.T2.GT.0.0) TTEMP = T2
      TACTUAL = TTEMP
    ENDIF
  ENDIF
ENDIF

IF RDUM IS GREATER THAN R, THEN WE KNOW THE INTERSECTION
OF THE RAY WITH THE EXIT IS NOT THE POINT WE
ARE INTERESTED IN BECAUSE IT IS NOT ON THE EXIT PLANE.
NOW USING THE QUADRATIC EQUATION AGAIN TO SOLVE FOR THE
INTERSECTION OF THE LINE WITH THE CONIC SECTION FROM Z=
332.0 TO 400.0

Q = -4.663464885
Z0 = 893.9475186
CALL CONESEARCH(XIN, YIN, ZIN, XRAY, YRAY, ZRAY, Q, Z0, T1, T2)
TTEMP = T1
IF (T1.LT.0.050) TTEMP = T2
IF (T2.LT.0.050) TTEMP = T1
ZJNT = ZIN + TTEMP*ZRAY
YJNT = YIN + TTEMP*YRAY
XJNT = XIN + TTEMP*XRAY

Appendix A  ANSIRS Code Listing
IF (ZJNT.GT.332.0.AND.ZJNT.LE.400.0) THEN
IF (TTEMP.GT.0.005) THEN
  IF (TTEMP.LT.TACTUAL.OR.TACTUAL.EQ.0.0) THEN
    TACTUAL = TTEMP
  ENDIF
ENDIF
ENDIF

IF ZJNN IS NOT IN THE ABOVE INTERVAL, I MUST SEARCH THE
NEXT SECTION, WHICH IS CYLINDRICAL IN SHAPE

A = XRAY*XRAY + YRAY*YRAY
B = 2.0*(XIN*XRAY + YIN*YRAY)
C = XIN*XIN + YIN*YIN - 106.133*106.133
CALL QUADRATIC(A, B, C, T1, T2)
IF (T1.LE.0.001) TTEMP = T2
IF (T2.LE.0.001) TTEMP = T1
XJNT = XIN + TTEMP*XRAY
YJNT = YIN + TTEMP*YRAY
ZJNT = ZIN + TTEMP*ZRAY
IF (ZJNT.GT.400.0.AND.ZJNT.LE.485.0) THEN
IF (TTEMP.GT.0.005) THEN
  IF (TTEMP.LT.TACTUAL.OR.TACTUAL.EQ.0.0) THEN
    TACTUAL = TTEMP
  ENDIF
ENDIF
ENDIF
ENDIF

IF ZJNN IS INSIDE THE ABOVE RANGE I HAVE THE POINT I AM
LOOKING FOR. IF NOT, I SEARCH THE NEXT SECTION
WHICH IS THE SECOND SECTION OF THE VARIABLE NOZZLE.
SINCE THE FIRST TWO SECTIONS ARE PLANE SURFACES, I NEED
THE SURFACE NORMAL AND A SURFACE POINT TO DEFINE THE
PLANE. XPLANE, YPLANE, AND ZPLANE ARE THE NORMAL VECTORS
AND U, V, AND W DEFINE A POINT ON THE SURFACE.

DO 20 K = 1,168
  IF (XRAY.EQ.0.0) XRAY = 0.000001
  XPUM = XPLANE(K)
  XNUM = XPUM*(XIN-U(K))*
  & YPLANE(K)*(YIN-V(K)) + ZPLANE(K)*(ZIN-W(K))
  DEN = XPLANE(K)*XRAY + YPLANE(K)*YRAY + ZPLANE(K)*ZRAY
  IF (DEN.EQ.0.0) DEN = 0.000001
  TTEMP = (XNUM/DEN)*(-1.0)
  XJNT = XIN + TTEMP*XRAY
  YJNT = YIN + TTEMP*YRAY
  ZJNT = ZIN + TTEMP*ZRAY
  IF (K.LE.96) SECTION = 1.0
  IF (K.GE.97) SECTION = 2.0
  CALL FINDRADIUS(ZJNT,NOZSET,L,R,SECTION,PI,ZSTART)
  RDOM = SQRT(XJNT*XJNT + YJNT*YJNT)
  IF (TTEMP.GT.0.05) THEN
    IF (RDOM.LE.R.AND.ZJNT.LE.Z(K).AND.ZJNT.GE.Z(K+24))
      THEN
        IF (K.NE.M) THEN
          IF (TTEMP.GT.0.05) THEN
            IF (TTEMP.LT.TACTUAL.OR.TACTUAL.EQ.0.0) THEN
              TACTUAL = TTEMP
            ENDIF
          ENDIF
        ENDIF
      ENDIF
    ENDIF
  ENDIF
20 CONTINUE

Appendix A  ANSIRS Code Listing  117
IF I HAVE A SOLUTION IN THE ABOVE SURFACES AND THE POINT
OF ORIGIN IS IN THE CLEAN PART, THEN I KNOW THAT I HAVE
THE TRUE INTERSECTION POINT AND NEED NOT SEARCH THE NEXT
PART OF THE GEOMETRY. THE TEST BELOW DETERMINES IF THIS
IS THE CASE.

IF (TACTUAL < GT.0.0) THEN
  IF (ZIN < GT.127.0 AND ZRAY < GT.0.0) THEN
    GOTO 200
  ENDIF
ENDIF

IF WE GET TO THIS POINT IN THE PROGRAM, WE KNOW THAT THE
ENERGY BUNDLE DID NOT LAND ANYWHERE IN THE CLEAN PART
OF THE ENGINE CAVITY. SO WE MUST NOW BEGIN THE SEARCH IN
THE MESSY PART OF THE CAVITY, AT A Z-VALUE OF LESS THAN
127 AND SEARCH EACH SECTION. AFTER THE COMPUTER HAS
SEARCHED EACH SECTION AND FOUND THE SHORTEST DISTANCE
TO INTERSECTION, WE HAVE OUR POINT. STARTING WITH
SECTION 11 (THE PLANE OF THE TURBINE BLADES).

ASSIGNING TACTUAL A VALUE IN ANY OF THE SEARCHES
INDICATES THAT AN INTERSECTION POINT WAS FOUND AND AT A
SHORTER DISTANCE THAN A PREVIOUS INTERSECTION.

ZJNT = 10.0
IF (ZRAY EQ.0.0) ZRAY = 0.000001
TTEMP = (ZJNT - ZIN)/ZRAY
XJNT = XIN + TTEMP*XRAY
YJNT = YIN + TTEMP*YRAY
DUMRAD = SQRT(XJNT*XJNT + YJNT*YJNT)
IF (DUMRAD GT.39.0 AND DUMRAD LT.120.5) THEN
  IF (TTEMP LT TACTUAL OR TACTUAL EQ.0.0) THEN
    TACTUAL = TTEMP
  ENDIF
ENDIF

THE NEXT SECTION I WILL SEARCH IS THE STRUCTURE
SURROUNDING THE TURBINE FRAME SUPPORTS.

A = XRAY*XRAY + YRAY*YRAY
B = 2.0*(XIN*XRAY + YIN*YRAY)
C = XIN*XIN + YIN*YIN - 8100.0
CALL QUADRATIC(A, B, C, T1, T2)
TTEMP = T1
IF (T1 LT 0.0 AND T2 GT 0.0) TTEMP = T2
IF (T2 LT 0.0 AND T1 GT 0.0) TTEMP = T1
IF (T1 GT 0.0 AND T2 GT 0.0 AND T1 LT T2) TTEMP = T1
IF (T1 GT 0.0 AND T2 GT 0.0 AND T2 LT T1) TTEMP = T2
ZJNT = ZIN + TTEMP*ZRAY
IF (ZJNT GT 10.0 AND ZJNT LT 27.0) THEN
  IF (TTEMP GT 0.0) THEN
    IF (TTEMP LT TACTUAL OR TACTUAL EQ.0.0) THEN
      TACTUAL = TTEMP
    ENDIF
    XJPREV = XIN + (TACTUAL - 0.5)*XRAY
    YJPREV = YIN + (TACTUAL - 0.5)*YRAY
    ZJPREV = ZIN + (TACTUAL - 0.5)*ZRAY
    RADIUS = SQRT(XJPREV*XJPREV + YJPREV*YJPREV)
    RACTUAL = 90.0
    IF (RACTUAL LT RADIUS) FLAG2 = 7.0
    IF (RACTUAL LT RADIUS) FLAG2 = 6.0
  ENDIF
ENDIF

Appendix A ANSIRS Code Listing 118
THE NEXT SECTION TO SEARCH IS THE CONICAL STRUCTURE WHICH I CALL THE PILOT CAN.

Q = -9.8
20 = 392.2
CALL CONESEARCH(XIN, YIN, ZIN, XRAY, YRAY, ZRAY, Q, Z0, T1, T2)
TTEMP = 0.0
IF (T1.GT.0.1.AND.T2.GT.0.1) THEN
ZT1 = ZIN + T1*ZRAY
ZT2 = ZIN + T2*ZRAY
IF (ZT1.GT.10.AND.ZT1.LT.108. 
   AND.ZT2.GT.10.AND.ZT2.LT.108) THEN
IF (T1.LT.T2) TTEMP = T1
IF (T2.LT.T1) TTEMP = T2
ENDIF
IF (ZT1.GT.10.AND. 
   & ZT1.LT.108.AND.(ZT2.LT.10.OR.ZT2.GT.108)) TTEMP=T1
IF (ZT2.GT.10.AND. 
   & ZT2.LT.108.AND.(ZT1.LT.10.OR.ZT1.GT.108)) TTEMP=T2
ENDIF
IF (T1.GT.0.1.AND.T2.LT.0.1) TTEMP = T1
IF (T2.GT.0.1.AND.T1.LT.0.1) TTEMP = T2
ZJNT = ZIN + TTEMP*ZRAY
YJNT = YIN + TTEMP*YRAY
XJNT = XIN + TTEMP*XRAY
D01 = SQRT(XJNT*XJNT + YJNT*YJNT)
IF (ZJNT.LT.108.0.AND.ZJNT.GT.10.0) THEN
IF (TTEMP.GT.0.0) THEN
IF (TTEMP.LT.TACTUAL.OR.TACTUAL.EQ.0.0) THEN
TACTUAL = TTEMP
XJPREV = XIN + (TACTUAL-0.5)*XRAY
YJPREV = YIN + (TACTUAL-0.5)*YRAY
ZJPREV = ZIN + (TACTUAL-0.5)*ZRAY
RJPREV = SQRT(XJPREV*XJPREV + YJPREV*YJPREV)
RACTUAL = (ZJNT-Z0)/Q
IF (RACTUAL.LT.RJPREV) FLAG2 = 3.0
IF (RJPREV.LE.RACTUAL) FLAG2 = 2.0
ENDIF
ENDIF
ENDIF

THE NEXT SECTION TO SEARCH IS THE CONICAL STRUCTURE WHICH IS THE INNER FACE OF THE INNERMOST V-GUTTER.

Q = -3.6
20 = 247.6
CALL CONESEARCH(XIN, YIN, ZIN, XRAY, YRAY, ZRAY, Q, Z0, T1, T2)
TTEMP = 0.0
IF (T1.GT.0.1.AND.T2.GT.0.1) THEN
Z1 = ZIN + T1*ZRAY
Z2 = ZIN + T2*ZRAY
IF (Z1.GT.118.AND.Z1.LT.127.AND.Z2.GT.118.AND.Z2.LT.127) 
   THEN
IF (T1.LT.T2) TTEMP = T1
IF (T2.LT.T1) TTEMP = T2
ENDIF
IF (Z1.GT.118.AND. 
   & Z1.LT.127.AND.(Z2.LT.118.OR.Z2.GT.127)) TTEMP=T1
IF (Z2.GT.118.AND. 
   & Z2.LT.127.AND.(Z1.LT.118.OR.Z1.GT.127)) TTEMP=T2
ENDIF
IF (T1.GT.0.1.AND.T2.LT.0.1) TTEMP = T1
IF (T2.GT.0.1.AND.T1.LT.0.1) TTEMP = T2

Appendix A  ANSIRS Code Listing 119
ZJNT = ZIN + TTEMP*ZRAY
YJNT = YIN + TTEMP*YRAY
XJNT = XIN + TTEMP*XRAY
D01 = SQRT(XJNT*XJNT + YJNT*YJNT)
IF (ZJNT.LT.127.0.AND.ZJNT.GT.118.0) THEN
  IF (TTEMP.GT.0.0) THEN
    TACTUAL = TTEMP
    XJPREV = XIN + (TACTUAL-0.5)*XRAY
    YJPREV = YIN + (TACTUAL-0.5)*YRAY
    ZJPREV = ZIN + (TACTUAL-0.5)*ZRAY
    RADPREV = SQRT(XJPREV*XJPREV + YJPREV*YJPREV)
    RACTUAL = (ZJNT-Z0)/Q
    IF (RACTUAL.LT.RADPREV) FLAG21 = 11.0
    IF (RADPREV.LE.RACTUAL) FLAG21 = 10.6
  ENDIF
ENDIF
ENDIF

THE NEXT SECTION TO SEARCH IS THE CONICAL STRUCTURE THAT
COMPRISSES THE OUTER FACE OF THE INNERMOST V-GUTTER.

Q = 3.6
Z0 = -11.6
CALL CONESEARCH(XIN,YIN,ZIN,XRAY,YRAY,ZRAY,Q,Z0,T1,T2)
TTEMP = 0.0
IF (T1.GT.0.1.AND.T2.GT.0.1) THEN
  Z1 = ZIN + T1*XRAY
  Z2 = ZIN + T2*XRAY
  IF (Z1.GT.118.AND.Z1.LT.127.AND.Z2.GT.118.AND.Z2.LT.127)
     THEN
    IF (T1.LT.T2) TTEMP = T1
    IF (T2.LT.T1) TTEMP = T2
  ENDIF
ENDIF
IF (Z1.GT.118.AND.
    & Z1.LT.127.AND.Z2.LT.118.AND.Z2.LT.127) TTEMP=T1
IF (Z2.GT.118.AND.
  & Z2.LT.127.AND.Z1.LT.118.AND.Z1.LT.127) TTEMP=T2
ENDIF
IF (T1.GT.0.1.AND.T2.LT.0.1) TTEMP = T1
IF (T2.GT.0.1.AND.T1.LT.0.1) TTEMP = T2
ZJNT = ZIN + TTEMP*XRAY
YJNT = YIN + TTEMP*YRAY
XJNT = XIN + TTEMP*XRAY
D01 = SQRT(XJNT*XJNT + YJNT*YJNT)
IF (ZJNT.LT.127.0.AND.ZJNT.GT.118.0) THEN
  IF (TTEMP.GT.0.0) THEN
    TACTUAL = TTEMP
    XJPREV = XIN + (TACTUAL-0.5)*XRAY
    YJPREV = YIN + (TACTUAL-0.5)*YRAY
    ZJPREV = ZIN + (TACTUAL-0.5)*ZRAY
    RADPREV = SQRT(XJPREV*XJPREV + YJPREV*YJPREV)
    RACTUAL = (ZJNT-Z0)/Q
    IF (RACTUAL.LT.RADPREV) FLAG21 = 10.0
    IF (RADPREV.LE.RACTUAL) FLAG21 = 11.0
  ENDIF
ENDIF
ENDIF

THE NEXT SECTION TO SEARCH IS THE CONICAL STRUCTURE
THAT CONNECTS THE INNER AND MIDDLE FLAMEHOLDER RINGS.

Q = -24.0/41.0
Z0 = 139.0731707
CALL CONESEARCH(XIN,YIN,ZIN,XRAY,YRAY,ZRAY,Q,Z0,T1,T2)
TTEMP = 0.0
IF (T1.GT.0.0.AND.T2.GT.0.0) THEN
  Z1= ZIN + T1*ZRAY
  Z2= ZIN + T2*ZRAY
  IF (Z1.GT.94.AND.
    & Z1.LT.118.AND.Z2.GT.94.AND.Z2.LT.118) THEN
    IF (T1.LT.T2) TTEMP = T1
    IF (T2.LT.T1) TTEMP = T2
  ENDIF
ENDIF
IF (Z1.GT.94.AND.
  & Z1.LT.118.AND.(Z2.LT.94.OR.Z2.GT.118)) TTEMP = T1
IF (Z2.GT.94.AND.
  & Z2.LT.118.AND.(Z1.LT.94.OR.Z1.GT.118)) TTEMP = T2
ENDIF
IF (T1.GT.0.0.AND.T2.LT.0.0) TTEMP = T1
IF (T2.GT.0.0.AND.T1.LT.0.0) TTEMP = T2
ZJNT = ZIN + TTEMP*ZRAY
YJNT = YIN + TTEMP*YRAY
XJNT = XIN + TTEMP*XRAY
D01 = SQRT(XJNT*XJNT + YJNT*YJNT)
IF (ZJNT.LT.118.0.AND.ZJNT.GT.94.0) THEN
  IF (XJNT.GE.0.0.AND.XJNT.LT.0.0001) XJNT = 0.0001
  IF (XJNT.LT.0.0.AND.XJNT.GT.-0.0001) XJNT = -0.0001
  ANGLE = ATAN(YJNT/XJNT)
  IF (XJNT.LT.0.0) ANGLE = ANGLE + PI
  IF (YJNT.LT.0.0.AND.XJNT.GT.0.0) ANGLE = ANGLE + 2.0*PI
  DUMANG1 = 15.0*PT/180.0
  DUMANG2 = (ANGLE/DUMANG1)
  IDUMANG1 = IFIX(DUMANG2)
  DUMANG3 = DUMANG2 - FLOAT(IDUMANG1)
  DUMANG4 = 6.0*PI/180.0
  IF (DUMANG3.LT.0.00001) THEN
    IF (TTEMP.GT.0.0) THEN
      TACTUAL = TTEMP
      XJPREV = XIN + (TACTUAL-0.5)*XRAY
      YJPREV = YIN + (TACTUAL-0.5)*YRAY
      ZJPREV = ZIN + (TACTUAL-0.5)*ZRAY
      RPREDV = SQRT(XJPREV*XJPREV + YJPREV*YJPREV)
      RACTUAL = (ZJNT-Z0)/Q
      IF (RACTUAL.LT.RPREDV) FLAG2 = 15.0
      IF (RPREDV.LE.RACTUAL) FLAG2 = 14.0
    ENDIF
  ENDIF
ENDIF
ENDIF
ENDIF

THE NEXT SECTION TO SEARCH IS THE CONICAL STRUCTURE
THAT FORMS THE INNER FACE OF THE MIDDLE FRAMEHOLDER
(V-GUTTER) RING.

Q = -3.6
Z0 = 371.2
CALL CONESEARCH(XIN,YIN,ZIN,XRAY,YRAY,ZRAY,Q,Z0,T1,T2)
TTEMP = 0.0
IF (T1.GT.0.1.AND.T2.GT.6.1) THEN
  Z1 = ZIN + T1*ZRAY
  Z2 = ZIN + T2*ZRAY
  IF (Z1.GT.94.AND.
    & Z1.LT.103.AND.Z2.GT.94.AND.Z2.LT.103) THEN
    IF (T1.LT.T2) TTEMP = T1
    IF (T2.LT.T1) TTEMP = T2
  ENDIF
ENDIF
IF (Z1.GT.94.AND.

Appendix A  ANSIRS Code Listing  121
& Z1.LT.103.AND.(Z2.LT.94.OR.Z2.GT.103)) TTEMP=T1
& IF(Z2.GT.94.AND.
& Z2.LT.103.AND.(Z1.LT.94.OR.Z1.GT.103)) TTEMP=T2
& ENDF
IF(T1.GT.0.1.AND.T2.LT.0.1) TTEMP = T1
IF(T2.GT.0.1.AND.T1.LT.0.1) TTEMP = T2
ZJNT = ZIN + TTEMP*XRAY
YJNT = YIN + TTEMP*YRAY
XJNT = XIN + TTEMP*XRAY
D01 = SQRT(XJNT*XJNT + YJNT*YJNT)
IF(ZJNT.LT.103.0.AND.ZJNT.GT.94.0) THEN
IF(TTEMP.GT.0.0) THEN
IF(TTEMP.LT.TACTUAL.OR.TACTUAL.EQ.0.0) THEN
TACTUAL = TTEMP
XJPREV = XIN + (TACTUAL-0.5)*XRAY
YPREV = YIN + (TACTUAL-0.5)*YRAY
ZJPREV = ZIN + (TACTUAL-0.5)*ZRAY
RDPREV = SQRT(XJPREV*XJPREV + YPREV*YPREV)
RACTUAL = (ZJNT-Z0)/Q
IF(RACTUAL.LT.RDPREV) FLG92 = 17.0
IF(RPREV.LE.RACTUAL) FLG92 = 16.0
ENDIF
ENDIF
ENDIF
ENDIF

THE NEXT SECTION TO SEARCH IS A CONICAL STRUCTURE WHICH
FORMS THE OUTER FACE OF THE MIDDLE V-GUTTER.

Q = 3.6
Z0 = -183.2
CALL CONESearch(XIN,YIN,ZIN,XRAY,YRAY,ZRAY,Q,Z0,T1,T2)
TTEMP = 0.0
IF(T1.GT.0.1.AND.T2.GT.0.1) THEN
Z1 = ZIN + T1*XRAY
Z2 = ZIN + T2*XRAY
IF(Z1.GT.94.AND.Z1.LT.103.AND.Z2.GT.94.AND.Z2.LT.103) THEN
IF(T1.LT.T2) TTEMP = T1
IF(T2.LT.T1) TTEMP = T2
ENDIF
ENDIF
IF(Z1.GT.94.AND.
& Z1.LT.103.AND.(Z2.LT.94.OR.Z2.GT.103)) TTEMP=T1
& IF(Z2.GT.94.AND.
& Z2.LT.103.AND.(Z1.LT.94.OR.Z1.GT.103)) TTEMP=T2
ENDIF
IF(T1.GT.6.1.AND.T2.LT.0.1) TTEMP = T1
IF(T2.GT.0.1.AND.T1.LT.0.1) TTEMP = T2
ZJNT = ZIN + TTEMP*XRAY
YJNT = YIN + TTEMP*YRAY
XJNT = XIN + TTEMP*XRAY
D01 = SQRT(XJNT*XJNT + YJNT*YJNT)
IF(ZJNT.LT.103.0.AND.ZJNT.GT.94.0) THEN
IF(TTEMP.GT.0.0) THEN
IF(TTEMP.LT.TACTUAL.OR.TACTUAL.EQ.0.0) THEN
TACTUAL = TTEMP
XJPREV = XIN + (TACTUAL-0.5)*XRAY
YPREV = YIN + (TACTUAL-0.5)*YRAY
ZJPREV = ZIN + (TACTUAL-0.5)*ZRAY
RDPREV = SQRT(XJPREV*XJPREV + YPREV*YPREV)
RACTUAL = (ZJNT-Z0)/Q
IF(RACTUAL.LT.RDPREV) FLG92 = 16.0
IF(RPREV.LE.RACTUAL) FLG92 = 17.0
ENDIF
ENDIF
ENDIF

Appendix A  ANSIRS Code Listing
THE NEXT SECTION TO SEARCH IS CONICAL IN SHAPE AND IS THE SUPPORT FROM THE MIDDLE TO OUTER FLAMEHOLDERS.

Q = 5.0/37.0
Z0 = 83.5945946
CALL CONESearch(XIN,YIN,ZIN,XRAY,YRAY,ZRAY,Q,Z0,T1,T2)
TTEMP = 0.0
IF (T1.GT.0.0.AND.T2.GT.0.0) THEN
  Z1 = ZIN + T1*XRAY
  Z2 = ZIN + T2*XRAY
ENDIF
IF (Z1.GT.99.AND.
& Z1.LT.99.AND.Z2.GT.94.AND.Z2.LT.99) THEN
  IF (T1.LT.T2) TTEMP = T1
  IF (T2.LT.T1) TTEMP = T2
ENDIF
IF (Z1.GT.94.AND.
& Z1.LT.99.AND.(Z2.LT.94.OR.Z2.GT.99)) TTEMP=T1
IF (Z2.GT.94.AND.
& Z2.LT.99.AND.(Z1.LT.94.OR.Z1.GT.99)) TTEMP=T2
DO1 = SQRT(XJNT*XJNT + YJNT*YJNT)
IF (ZJNT.LT.99.00.AND.ZJNT.GT.94.0) THEN
  IF (XJNT.GE.0.0.AND.XJNT.LT.0.0001) XJNT = 0.0001
  IF (XJNT.LT.0.0001.AND.XJNT.GT.-0.0001) XJNT = -0.0001
  ANGLE = ATAN(YJNT/XJNT)
  IF (XJNT.LT.0.0) ANGLE = ANGLE + PI
  IF (YJNT.LT.0.0.AND.XJNT.GT.0.0) ANGLE = ANGLE + 2.0*PI
  DUMANG5 = 15.0*PI/180.0
  DUMANG6 = ANGLE/DUMANG5
  IDUMANG2 = IP[X(DUMANG6)]
  DUMANG7 = DUMANG6 - FLOAT(IDUMANG2)
  DUMANG8 = 6.0*PI/180.0
  IF (DUMANG8.GT.0.40000) THEN
    IF (TTEMP.GT.0.0) THEN
      TACTUAL = TTEMP
      XJPREV = XIN + (TACTUAL-0.5)*XRAY
      YJPREV = YIN + (TACTUAL-0.5)*YRAY
      ZJPREV = ZIN + (TACTUAL-0.5)*ZRAY
      RADPRev = SQRT(XJPREV*XJPREV + YJPREV*YJPREV)
      RACTUAL = (ZJNT-Z0)/Q
      IF (RACTUAL.LT.RADPRev) FLAG2 = 20.0
      IF (RADPRev.LE.RACTUAL) FLAG2 = 21.0
      ENDIF
      ENDIF
      ENDIF
      ENDIF
      ENDIF
      ENDIF
THE NEXT SECTION TO SEARCH IS THE CONICAL STRUCTURE THAT FORMS THE INNER SIDE OF THE OUTER FLAMEHOLDER.

Q = -3.6
Z0 = 509.4
CALL CONESearch(XIN,YIN,ZIN,XRAY,YRAY,ZRAY,Q,Z0,T1,T2)
TTEMP = 0.0
IF (T1.GT.0.1.AND.T2.GT.0.1) THEN
  Z1 = ZIN + T1*XRAY
  Z2 = ZIN + T2*XRAY
ENDIF
IF (Z1.GT.99.AND.Z1.LT.108.AND.Z2.GT.99.AND.Z2.LT.108)
&

THEN

IF (T1.LT.T2) TTEMP = T1
IF (T2.LT.T1) TTEMP = T2
ENDIF

IF (Z1.GT.99.AND.
&
Z1.LT.108.AND.(Z2.LT.99.OR.Z2.GT.108)) TTEMP=T1

IF (Z2.GT.99.AND.
&
Z2.LT.108.AND.(Z1.LT.99.OR.Z1.GT.108)) TTEMP=T2
ENDIF
IF (T1.GT.0.1.AND.T2.LT.0.1) TTEMP = T1
IF (T2.GT.0.1.AND.T1.LT.0.1) TTEMP = T2
ZJNT = ZIN + TTEMP*ZRAY
YJNT = YIN + TTEMP*YRAY
XJNT = Xin + TTEMP*XRAY
D01 = SQRT(XJNT*XJNT + YJNT*YJNT);

IF (ZJNT.LT.108.0.AND.ZJNT.GT.99.0) THEN
IF (TTEMP.GT.0.0) THEN
IF (TTEMP.LT.TACTUAL.OR.TACTUAL.EQ.0.0) THEN
TACTUAL = TTEMP
XJPREV = Xin + (TACTUAL-0.5)*XRAY
YJPREV = YIN + (TACTUAL-0.5)*YRAY
ZJPREV = ZIN + (TACTUAL-0.5)*ZRAY
RAPREV = SQRT(XJPREV*XJPREV + YJPREV*YJPREV)
RACTUAL = (ZJNT-20)/Q
IF (FACTUAL.LT.RAPREV) FLAG23 = 23.0
IF (RAPREV.LE.RACTUAL) FLAG23 = 22.0
ENDIF
ENDIF
ENDIF

Q = 3.6
Z0 = -311.4
CALL CONESearch(XIN,YIN,ZIN,XRAY,YRAY,ZRAY,Q,Z0,T1,T2)
TTEMP = 0.0
IF (T1.GT.0.1.AND.T2.GT.0.1) THEN
Z1 = ZIN + T1*ZRAY
Z2 = ZIN + T2*ZRAY
IF (Z1.GT.99.AND.Z1.LT.108.AND.Z2.GT.99.AND.Z2.LT.108) THEN
IF (T1.LT.T2) TTEMP = T1
IF (T2.LT.T1) TTEMP = T2
ENDIF
IF (Z1.GT.99.AND.
&
Z1.LT.108.AND.(Z2.LT.99.OR.Z2.GT.108)) TTEMP=T1

IF (Z2.GT.99.AND.
&
Z2.LT.108.AND.(Z1.LT.99.OR.Z1.GT.108)) TTEMP=T2
ENDIF
IF (T1.GT.0.1.AND.T2.LT.0.1) TTEMP = T1
IF (T2.GT.0.1.AND.T1.LT.0.1) TTEMP = T2
ZJNT = ZIN + TTEMP*ZRAY
YJNT = YIN + TTEMP*YRAY
XJNT = Xin + TTEMP*XRAY
D01 = SQRT(XJNT*XJNT + YJNT*YJNT);

IF (ZJNT.LT.108.0.AND.ZJNT.GT.99.0) THEN
IF (TTEMP.GT.0.0) THEN
IF (TTEMP.LT.TACTUAL.OR.TACTUAL.EQ.0.0) THEN
TACTUAL = TTEMP
XJPREV = Xin + (TACTUAL-0.5)*XRAY
YJPREV = YIN + (TACTUAL-0.5)*YRAY
ZJPREV = ZIN + (TACTUAL-0.5)*ZRAY
RAPREV = SQRT(XJPREV*XJPREV + YJPREV*YJPREV)
RACTUAL = (ZJNT-20)/Q

Appendix A  ANSIRS Code Listing

124
IF (RACTUAL.LT.RADPREV) FLAG93 = 22.0
IF (RADPREV.LE.RACTUAL) FLAG93 = 23.0
ENDIF
ENDIF
ENDIF

THE NEXT SECTION TO SEARCH IS THE CONICAL STRUCTURE
I CALL THE TURBULATOR OR FLOW MIXING DUCT (CORE/BYPASS).

Q = -1.0*(3.0)**0.5
Z0 = 182.8845727
CALL CONESRCH2(XIN,YIN,ZIN,XRAY,YRAY,ZRAY,Q,Z0,T1,T2)
TTEMP = 0.0
IF (T1.GT.0.0.AND.T2.GT.0.0) THEN
  Z1 = ZIN + T1*ZRAY
  Z2 = ZIN + T2*ZRAY
IF(T1.GT.27.AND.
  & Z1.LT.44.32.AND.Z2.GT.27.AND.Z2.LT.44.32) THEN
  IF (T1.LT.T2) TTEMP = T1
  IF (T2.LT.T1) TTEMP = T2
ENDIF
Z3 = 44.32
IF (Z1.GT.27.AND.
  & Z1.LT.Z3.AND.(Z2.LT.27.OR.Z2.GT.Z3)) TTEMP=T1
IF (Z2.GT.27.AND.
  & Z2.LT.Z3.AND.(Z1.LT.27.OR.Z1.GT.Z3)) TTEMP=T2
ENDIF
IF (T1.GT.0.0.AND.T2.LT.0.0) TTEMP = T1
IF (T2.GT.0.0.AND.T1.LT.0.0) TTEMP = T2
ZJNT = ZIN + TTEMP*ZRAY
YJNT = YIN + TTEMP*YRAY
XJNT = XIN + TTEMP*XRAY
D01 = SQRT(XJNT**XJNT + YJNT**YJNT)
IF (ZJNT.LT.44.32.AND.ZJNT.GT.27.0) THEN
  IF (XJNT.GE.0.0.AND.XJNT.LT.0.0001) XJNT = 0.0001
  IF (XJNT.LT.0.0.AND.XJNT.GT.-0.0001) XJNT =-0.0001
  ANGLE = ATAN(XJNT/XJNT)
  IF (XJNT.LT.0.0) ANGLE = ANGLE + PI
  IF (XJNT.LT.0.0.AND.XJNT.GT.0.0) ANGLE=ANGLE+2.0*PI
DUMANG9 = 15.0*PI/180.0
DUMANG10 = ANGLE/DUMANG9
IDUMANG3 = IFIX(DUMANG10)
DUMANG11 = DUMANG10 - FLOAT(IDUMANG3)
DUMANG12 = 7.5*PI/180.0
IF (DUMANG11.LT.0.50) THEN
  IF (TTEMP.GT.0.0) THEN
    IF (TTEMP.LT.TACTUAL.OR.TACTUAL.EQ.0.0) THEN
      TACTUAL = TTEMP
XJPREV = XIN + (TACTUAL-0.5)*XRAY
YJPREV = YIN + (TACTUAL-0.5)*YRAY
ZJPREV = ZIN + (TACTUAL-0.5)*ZRAY
RADPREV = SQRT(XJPREV**XJPREV + YJPREV**YJPREV)
RACTUAL = (ZJNT-Z0)/Q
IF (RACTUAL.LT.RADPREV) FLAG25 = 2.0
IF (RADPREV.LE.RACTUAL) FLAG25 = 1.0
ENDIF
ENDIF
ENDIF
ENDIF

THE NEXT SECTION TO SEARCH IS THE CONICAL STRUCTURE
WHICH IS THE OTHER PART OF THE FLOW MIXING DUCT.

Q = (3.0)**0.5
Z0 = -128.8845727

Appendix A  ANSIRS Code Listing
CALL CONESEARCH(XIN,YIN,ZIN,XRAY,YRAY,ZRAY,Q,Z0,T1,T2)
TTTEMP = 0.0
IF (T1.GT.0.0.AND.T2.GT.0.0) THEN
    Z1 = ZIN + T1*ZRAY
    Z2 = ZIN + T2*ZRAY
ENDIF

& Z1.LT.44.32.AND.Z2.GT.27.AND.Z2.LT.44.32) THEN
    IF (T1.LT.T2) TTEMP = T1
    IF (T2.LT.T1) TTEMP = T2
ENDIF

Z3 = 44.32
IF (Z1.GT.27.AND.
    & Z1.LT.Z3.AND.(Z2.LT.27.OR.Z2.GT.Z3)) TTEMP=T1
IF (Z2.GT.27.AND.
    & Z2.LT.Z3.AND.(Z1.LT.27.OR.Z1.GT.Z3)) TTEMP=T2

ENDIF

IF (T1.GT.0.0.AND.T2.LT.0.0) TTEMP = T1
IF (T2.GT.0.0.AND.T1.LT.0.0) TTEMP = T2
ZJNT = ZIN + TTEMP*ZRAY
YJNT = YIN + TTEMP*YRAY
XJNT = XIN + TTEMP*XRAY
D01 = SQRT(XJNT*XJNT + YJNT*YJNT)

& (ZJNT.LT.44.32.AND.ZJNT.GT.27.0) THEN
    IF (ZJNT.GE.0.0.AND.ZJNT.LT.0.0001) XJNT = 0.0001
    IF (ZJNT.LT.0.0.AND.ZJNT.GT.-0.0001) XJNT = -0.0001
    ANGLE = ATAN(YJNT/XJNT)
    IF (XJNT.LT.0.0) ANGLE = ANGLE + PI
    IF (YJNT.LT.0.0.AND.XJNT.GT.0.0) ANGLE=ANGLE+2.0*PI

DUMANG13 = 15.0*PI/180.0
DUMANG14 = ANGLE/DUMANG13
DUMANG4 = IFIX(DUMANG14)
DUMANG15 = DUMANG14 - FLOAT(DUMANG4)
DUMANG16 = 7.5*PI/180.0

IF (DUMANG15.GT.0.50) THEN
    IF (TTEMP.GT.0.0) THEN
        IF (TTEMP.LT.TACTUAL.CR.TACTUAL.EQ.0.0) THEN
            TACTUAL = TTEMP
        XJPREV = XIN + (TACTUAL-0.5)*XRAY
        YJPREV = YIN + (TACTUAL-0.5)*YRAY
        ZJPREV = ZIN + (TACTUAL-0.5)*ZRAY
        RAPREV = SQRT(XJPREV**XRAY + YJPREV**YRAY + ZJPREV**ZRAY)
        RACTUAL = (ZJNT-20)/Q
        IF ((RACTUAL.LT.RAPREV) .AND. (FLAG27 .EQ. 2.0)
            .AND. (RACTUAL.LE.RACTUAL) .AND. (FLAG27 .EQ. 1.0)
            ENDIF
        ENDIF
        ENDIF
        ENDIF
        ENDIF
        THE NEXT SECTION THAT WILL BE SEARCHED IS THE BEARING COVER, WHICH IS SPHERICAL IN SHAPE.

TTEMP1 = 0.0
TTEMP2 = 0.0
TTEMP = 0.0
ZJNT = 0.0
IF (ABS(XRAY).LT.0.000001) XRAY = 0.000001
A = XRAY*XRAY + YRAY*YRAY + ZRAY*ZRAY
B = 2.0*(XIN*XRAY + YIN*YRAY + ZIN*ZRAY - 10.0*ZRAY)
C = XIN*XIN + YIN*YIN + ZIN*ZIN -20.0*ZIN +1421.00
CALL QUADRATIC(A, B, C, T1, T2)
TTEMP = 0.0
IF (T1.GT.0.0.AND.T2.GT.0.0.AND.T1.LT.T2) TTEMP = T1
IF (T2.GT.0.0.AND.T1.GT.0.0.AND.T2.LT.T1) TTEMP = T2
IF (T1.GT.0.0.AND.T2.LE.0.0) TTEMP = T1

Appendix A  ANSIRS Code Listing
126
IF (T2.GT.0.0 .AND. T1.LE.0.0) TTEMP = T2
IF (TTEMP.NE.0.0) THEN
ZJNT = ZIN + TTEMP*ZRAY
XJNT = XIN + TTEMP*XRAY
YJNT = YIN + TTEMP*YRAY
IF (ZJNT.GT.10.0 .AND. ZJNT.LE.49.000) THEN
  IF (TTEMP.GT.0.0) THEN
    IF (TTEMP.LT.TACTUAL.OR.TACTUAL.EQ.0.0) THEN
      IF (ZJNT.GT.25.0) THEN
        TACTUAL = TTEMP
      ENDIF
    ENDIF
  ENDIF
ENDIF
IF (ZJNT.LE.25.0 .AND. M.GE.673 .AND. M.LE.792) THEN
  TACTUAL = TTEMP
ENDIF
ENDIF
ENDIF
ENDIF
ENDIF

THE NEXT SECTIONS TO SEARCH ARE THE TURBINE FRAME AND FUEL C
INJECTORS. THEY ARE GROUPED TOGETHER BECAUSE THEY ARE
PLANAR SURFACES AND THEIR ELEMENT NUMBERS ARE SEQUENTIAL C

DO 80 K = 841,864
  XPDM = XPLANE(K)
  YPDM = YPLANE(K)
  ZPDM = ZPLANE(K)
  XNUM = XPDM*(XIN-U(K))
  &
  YPDM*(YIN-V(K)) + ZPDM*(ZIN-W(K))
  DEN = XPLANE(K)*XRAY + YPLANE(K)*YRAY + ZPLANE(K)*ZRAY
  IF (DEN.EQ.0.0) DEN = 0.000001
  TTEMP = (XNUM/DEN)*(-1.0)
  XJNT = XIN + TTEMP*XRAY
  YJNT = YIN + TTEMP*YRAY
  ZJNT = ZIN + TTEMP*ZRAY
  DUM1 = SQRT(XJNT*XJNT + YJNT*YJNT)
  F0 = 27.0 + ((90.0-DUM1)/51.0)*13.0
  IF (DUM1.GT.39 .AND. DUM1.LT.90 .AND. ZJNT.GT.10 .AND. ZJNT.LT.0) THEN
    IF (TTEMP.GT.0.1) THEN
      IF (TTEMP.LT.TACTUAL.OR.TACTUAL.EQ.0.0) THEN
        XJP = XIN + (TTEMP-0.5)*XRAY
        YJP = YIN + (TTEMP-0.5)*YRAY
        ZJP = ZIN + (TTEMP-0.5)*ZRAY
        CALL FINDLENGTH(XJP,YJP,ZJP,U(K),V(K),W(K),DIST)
        IF (DIST.LT.65.0) THEN
          TACTUAL = TTEMP
          TS1 = XPDM*(XJP-U(K))
          &
          YPDM*(YJP-V(K)) + ZPDM*(ZJP-W(K))
          IF (TS1.GT.0.0) FLAG32 = 4.0
          IF (TS1.LT.0.0) FLAG32 = 5.0
        ENDIF
      ENDIF
    ENDIF
  ENDIF
  CONTINUE
  DO 85 K = 889,912
  XPDM = XPLANE(K)
  YPDM = YPLANE(K)
  ZPDM = ZPLANE(K)
  XNUM = XPDM*(XIN-U(K))
  &
  YPDM*(YIN-V(K)) + ZPDM*(ZIN-W(K))
  DEN = XPDM*XRAY + YPDM*YRAY + ZPDM*ZRAY

Appendix A  ANSIRS Code Listing 127
IF (DEN.EQ.0.0) DEN = 0.0000001
TTTEMP = (XNUM/DEN)*(-1.0)
XJNT = XIN + TTEMP*XRAY
YJNT = YIN + TTEMP*YRAY
ZJNT = ZIN + TTEMP*ZRAY
DUM = SQRT(XJNT*XJNT + YJNT*YJNT)
& DUM1.LT.51.AND.
& DUM1.LT.80.AND.ZJNT.GT.50.AND.ZJNT.LT.62 THEN
IF (TTTEMP.GT.0.1) THEN
IF (TTTEMP.LT.TACTUAL.OR.TACTUAL.EQ.0.0) THEN
XJP = XIN + (TTTEMP-0.5)*XRAY
YJP = YIN + (TTTEMP-0.5)*YRAY
ZJP = ZIN + (TTTEMP-0.5)*ZRAY
CALL FINDLENGTH(XJP,YJP,ZJP,U(K),V(K),W(K),DIST)
ENDIF
ENDIF
ENDIF
ENDIF
85 CONTINUE

THE FINAL SECTIONS THAT NEED TO BE SEARCHED ARE THE 96 SURFACES THAT COMPRIS THE DIVIDING WALLS FOR THE CORE

/BYPASS MIXING APPARATUS

DO 90 K = 1369,1392
XPDUM = XPLANE(K)
YPDUM = YPLANE(K)
ZPDUM = ZPLANE(K)
XNUM= XPDUM*(XIN-U(K))+
& YPDUM*(YIN-V(K)) + ZPDUM*(ZIN-W(K))
DEN = XPLANE(K)*XRAY + YPLANE(K)*YRAY + ZPLANE(K)*ZRAY
IF (DEN.EQ.0.0) DEN = 0.0000001
TTTEMP = (XNUM/DEN)*(-1.0)
XJNT = XIN + TTEMP*XRAY
YJNT = YIN + TTEMP*YRAY
ZJNT = ZIN + TTEMP*ZRAY
D1 = 80.0 + ((44.32-ZJNT)/17.32)*10.0
D2 = 90.0 + ((ZJNT - 27.0)/17.32)*10.0
IF (D1.GT.144.32.AND.D1.LT.144.32) THEN
& ZJNT.GT.27.AND.ZJNT.LT.44.32 THEN
IF (TTTEMP.GT.0.1) THEN
IF (TTTEMP.LT.TACTUAL.OR.TACTUAL.EQ.0.0) THEN
XJP = XIN + (TTTEMP-0.5)*XRAY
YJP = YIN + (TTTEMP-0.5)*YRAY
ZJP = ZIN + (TTTEMP-0.5)*ZRAY
CALL FINDLENGTH(XJP,YJP,ZJP,U(K),V(K),W(K),DIST)
ENDIF
ENDIF
ENDIF
TACTUAL = TTEMP
TST = XPDUM*(XJP-U(K)) +
& YPDUM*(YJP-V(K)) + ZPDUM*(ZJP-W(K))
IF (TST.GT.0.0) FLAG24 = 1.0
IF (TST.LT.0.0) FLAG24 = 2.0
ENDIF
ENDIF
ENDIF
90 CONTINUE

DO 95 K = 1427, 1440

Appendix A  ANSIRS Code Listing  128
XPDUM = XPLANE(K)
YPDUM = YPLANE(K)
ZPDUM = ZPLANE(K)
XNUM = XPDUM*(XIN-U(K)) +
& YPDUM*(YIN-V(K)) + ZPDUM*(ZIN-W(K))
DEN = XPDUM*XRAY + YPDUM*YRAY + ZPDUM*ZRAY
IF (DEN.EQ.0.0) DEN = 0.0000001
TTEMP = (XNUM/DEN)*(-1.0)
XJNT = XIN + TTEMP*XRAY
YJNT = YIN + TTEMP*YRAY
ZJNT = ZIN + TTEMP*ZRAY
DM = SQRT(XJNT**2 + YJNT**2 + ZJNT**2)
D1 = 80.0 + ((44.32-ZJNT)/17.32)*10.0
D2 = 90.0 + ((ZJNT - 27.0)/17.32)*10.0
IF (DM.GT.D1.AND.DM.LT.D2.AND.
& ZJNT.GT.27.AND.ZJNT.LT.44.32) THEN
IF (TTEMP.GT.0.1) THEN
IF (TTEMP.LT.TACTUAL.OR.TACTUAL.EQ.0.0) THEN
XJP = XIN + (TTEMP-0.5)*XRAY
YJP = YIN + (TTEMP-0.5)*YRAY
ZJP = ZIN + (TTEMP-0.5)*ZRAY
CALL FINDLENGTH(XJP,YJP,ZJP,U(K),V(K),W(K),DIST)
IF (DIST.LT.80.0) THEN
TACTUAL = TTEMP
TST = XPDUM*(XJP-U(K)) +
& YPDUM*(YJP-V(K)) + ZPDUM*(ZJP-W(K))
IF (TST.GT.0.0) FLAG24 = 3.0
IF (TST.LT.0.0) FLAG24 = 4.0
ENDIF
ENDIF
ENDIF
ENDIF
95 CONTINUE

WHEN WE GET TO THIS POINT IN THE CODE WE SHOULD HAVE ONE
VALUE FOR TACTUAL, OR THE SCALING FACTOR. KNOWING
THIS FACTOR, WE MAY FIND THE FINAL POINT OF INTERSECTION
WITHIN THE ENGINE CAVITY. THEN WE SIMPLY CALL SUBROUTINE
SURFACE TO DETERMINE THE ELEMENT NUMBER OF INTERSECTION
THEN WE MAY ACCESS DATA SUCH AS EMISSIVITY, ETC. FOR
THAT PARTICULAR ELEMENT.

200 CONTINUE
XJNN = XIN + TACTUAL*XRAY
YJNN = YIN + TACTUAL*YRAY
ZJNN = ZIN + TACTUAL*ZRAY
IF (XJNN.EQ.0.0) XJNN = 0.0001
ZMXDUM = ZMAX - 0.001
IF (ZJNN.GE.ZMXDUM) ZJNN = ZJNN+0.001
CALL SURFACE(X,Y,Z,XNN,YNN,ZNN,ISURF,ZMAX,ZSTART,
& SECTION,P1,FLAG2,FLAG21,FLAG22,FLAG23,FLAG24,FLAG25,
& FLAG26,FLAG27,FLAG32,FLAG42,FLAG91,FLAG92,FLAG93,M)
CASE = 0.0
CALL AB(ISURF,ALP,XAP,SEED,CASE)
IF (CASE.EQ.1.0) THEN
ISURF = ISURF
ENDIF
IF (CASE.EQ.2.0) THEN
XI = XJNN
YI = YJNN
ZI = ZJNN
ELEMENT = FLOAT(ISURF) -
& M = ISURF
CALL DIRECTION(SEED,THETA,PHI,PI)
GOTO 3
SUBROUTINE SURFACE(X,Y,Z,D,E,F,ISURF,ZMAX,ZSTART,
& SECTION,PI,FLAG2,FLAG21,FLAG22,FLAG23,
& FLAG24,FLAG25,FLAG26,FLAG27,FLAG32,
& FLAG42,FLAG91,FLAG92,FLAG93,M)
C
C THIS SUBROUTINE IDENTIFIES A SURFACE ELEMENT NUMBER FOR
A GIVEN SET OF POINTS. THE INPUT POINTS ARE D, E, F,
REPRESENTING THE X, Y, AND Z COORDINATES OF THE POINT OF
INTEREST. THE SUBROUTINE DETERMINES THE SURFACE ELEMENT
NUMBER AND PASS IT BACK AS ISURF.
C
DIMENSION X(2064), Y(2064), Z(2064)
REAL PI

SECTION = 0.0
IF (F.GE.ZMAX) SECTION = 42.0
IF (F.GE.ZSTART.AND.F.LT.ZMAX) SECTION = 1.0
IF (F.GE.465.0.AND.F.LT.ZSTART) SECTION = 2.0
IF (F.GE.400.0.AND.F.LT.465.0) SECTION = 3.0
IF (F.GE.332.0.AND.F.LT.400.0) SECTION = 4.0
IF (F.GT.147.0.AND.F.LE.332.0) SECTION = 5.0
DM1 = SQRT(D*D + E*E)
DM2 = SQRT(D*D + F*F)
IF (F.GT.121.0.AND.
& F.LT.147.0.AND.DUM1.GT.118.0) SECTION = 6.0
IF (F.GE.10.0.AND.F.LT.121.0.AND.DUM1.GT.120.0) THEN
SECTION = 7.0
ENDIF
IF (F.LT.10.01.AND.DUM1.GT.39.0) SECTION = 11.0
IF (F.GE.10.01.AND.F.LT.110.0.AND.DUM1.GT.40.0) THEN
IF (FLAG2.EQ.2.0) SECTION = 9.0
IF (FLAG2.EQ.3.0) SECTION = 10.0
ENDIF
ENDIF
IF (F.LE.49.0) THEN
DUM2 = F - 10.0
DUM3 = SQRT((DUM1*DUM1) + (DUM2*DUM2))
IF ((ABS(DUM3 - 39.0)).LT.0.001) SECTION = 8.0
F0 = 27.0 - ((90.0 - DUM1)/51.0)*13.0
IF (DUM1.GT.39.AND.
  DUM1.LT.90.AND.F.LE.F0.AND.FLAG32.EQ.4) THEN
  SECTION = 14.0
ENDIF
IF (DUM1.GT.39.AND.
  DUM1.LT.90.AND.F.LE.F0.AND.FLAG32.EQ.5) THEN
  SECTION = 15.0
ENDIF
IF (ABS(DUM1-90.0).LT.0.01.AND.FLAG2.EQ.6.0) SECTION = 12.0
IF (ABS(DUM1-90.0).LT.0.01.AND.FLAG2.EQ.7.0) SECTION = 13.0
ENDIF
IF (F.GE.50.0.AND.
  F.LE.62.0.AND.DUM1.GE.51.AND.DUM1.LE.80) THEN
  IF (FLAG2.EQ.8.0) SECTION = 18.0
  IF (FLAG2.EQ.9.0) SECTION = 19.0
ENDIF
IF (F.GE.118.0.AND.F.LE.127.0) THEN
  IF (DUM1.LT.36.0) THEN
    IF (FLAG21.EQ.10.0) SECTION = 18.0
    IF (FLAG21.EQ.11.0) SECTION = 19.0
  ENDIF
  IF (DUM1.GT.36.0.AND.DUM1.LT.38.5) THEN
    IF (FLAG91.EQ.11.0) SECTION = 20.0
    IF (FLAG91.EQ.10.0) SECTION = 21.0
  ENDIF
ENDIF
IF (F.GE.94.0.AND.F.LT.118.0) THEN
  DU15 = (118.0-F)/24.0*41.0 + 36.1
  DM16 = DU15 - 0.2
  IF (DUM1.GE.DM16.AND.DUM1.LE.DU15) THEN
    IF (FLAG2.EQ.14.0) SECTION = 22.0
    IF (FLAG2.EQ.15.0) SECTION = 23.0
  ENDIF
ENDIF
IF (F.GE.94.0.AND.F.LT.103.0) THEN
  DU17 = (F-94.0)/9.0*2.50 + 77.01
  DU18 = 76.99 - (F-94.0)/9.0*2.50
  IF (DUM1.LT.77.0.AND.DUM1.GT.DU18) THEN
    IF (FLAG22.EQ.16.0) SECTION = 24.0
    IF (FLAG22.EQ.17.0) SECTION = 25.0
  ENDIF
  IF (DUM1.GT.77.0.AND.DUM1.LT.DU17) THEN
    IF (FLAG92.EQ.17.0) SECTION = 26.0
    IF (FLAG92.EQ.16.0) SECTION = 27.0
  ENDIF
ENDIF
IF (F.GE.94.0.AND.F.LT.99.0) THEN
  DU19 = (F-94.0)/5.0*37.0 + 76.99
  DU20 = DU19 + 0.02
  IF (DUM1.GT.DU19.AND.DUM1.LT.DU20) THEN
    IF (FLAG2.EQ.20.0) SECTION = 28.0
    IF (FLAG2.EQ.21.0) SECTION = 29.0
  ENDIF
ENDIF
IF (F.GE.99.0.AND.F.LT.108.0) THEN
  IF (DUM1.LT.114.0.AND.DUM1.GT.111.5) THEN
    IF (FLAG23.EQ.22.0) SECTION = 30.0
    IF (FLAG23.EQ.23.0) SECTION = 31.0
  ENDIF
  IF (DUM1.GT.114.0.AND.DUM1.LT.116.5) THEN
    IF (FLAG93.EQ.23.0) SECTION = 32.0
    IF (FLAG93.EQ.22.0) SECTION = 33.0
  ENDIF

Appendix A  ANSIRS Code Listing  131
ENDIF
ENDIF
IF (F.GE.27.0.AND.F.LE.44.32.AND.DUM1.GE.80.0) THEN
IF (DUM1.LE.90.0) THEN
RDU1 = 90.0 - (F-27.0)*(10.0/17.32)
ENDIF
IF (ABS(DUM1 - RDU1).GT.0.005) THEN
IF (FLAG24.EQ.1.0) SECTION = 36.0
IF (FLAG24.EQ.2.0) SECTION = 37.0
IF (FLAG24.EQ.3.0) SECTION = 38.0
IF (FLAG24.EQ.4.0) SECTION = 39.0
ENDIF
IF (ABS(DUM1 - RDU1).LT.0.005) THEN
IF (FLAG25.EQ.1.0) SECTION = 34.0
IF (FLAG25.EQ.2.0) SECTION = 35.0
ENDIF
ENDIF
IF (DUM1.GE.90.0.AND.DUM1.LT.101.0) THEN
RDU1 = 90.0 + (F-27.0)*(10.0/17.32)
IF (ABS(DUM1 - RDU1).GT.0.005) THEN
IF (FLAG24.EQ.3.0) SECTION = 36.0
IF (FLAG24.EQ.4.0) SECTION = 39.0
IF (FLAG24.EQ.1.0) SECTION = 36.0
IF (FLAG24.EQ.2.0) SECTION = 37.0
ENDIF
IF (ABS(DUM1 - RDU1).LT.0.005) THEN
IF (FLAG27.EQ.1.0) SECTION = 40.0
IF (FLAG27.EQ.2.0) SECTION = 41.0
ENDIF
ENDIF
IF (SECTION.EQ.1.0) THEN
DUM4 = (ZMAX-ZSTART)/4.0
DUM5 = (ZMAX-F)/DUM4
IRING = 1 + IFIX(DUM5)
ENDIF
IF (SECTION.EQ.2.0) THEN
DUM5 = (538.00-465.0)/3.0
DUM7 = (538.00-F)/DUM6
IRING = 5 + IFIX(DUM7)
ENDIF
IF (SECTION.EQ.3.0) THEN
IF (F.GT.444.0.AND.F.LE.465.0) IRING = 8
IF (F.GT.416.0.AND.F.LE.465.0) IRING = 9
IF (F.GE.400.0.AND.F.LE.416.0) IRING = 10
ENDIF
IF (SECTION.EQ.4.0) THEN
DUM10 = 34.0
DUM11 = (400.0-F)/DUM10
IRING = 11 + IFIX(DUM11)
ENDIF
IF (SECTION.EQ.5.0) THEN
DUM12 = (332.0-147.0)/7.0
DUM13 = (332.0-F)/DUM12
IRING = 13 + IFIX(DUM13)
ENDIF
IF (SECTION.EQ.6.0) THEN
IRING = 20
ENDIF
IF (SECTION.EQ.7.0) THEN
IF (F.GE.91.0.AND.F.LT.121.0) IRING = 21
IF (F.GE.61.0.AND.F.LT.91.0) IRING = 22
IF (F.GE.10.0.AND.F.LT.61.0) IRING = 23
ENDIF
IF (SECTION.EQ.11.0) THEN
DUM18 = SQRT(D*D + E*E)

Appendix A ANSIRS Code Listing

132
IF (DUM18.GT.90.0) IRING = 24
IF (DUM18.LE.90.0) IRING = 25
ENDIF
IF (SECTION.EQ.9.0) THEN
  DUM19 = 33.0
  DUM20 = (F - 10.0)/DUM19
  IRING = 26 + IFIX(DUM20)
ENDIF
IF (SECTION.EQ.10.0) THEN
  DUM21 = (109.0 - 10.0)/3.0
  DUM22 = (109.0 - F)/DUM21
  IRING = 29 + IFIX(DUM22)
ENDIF
IF (SECTION.EQ.8.0) THEN
  IF (F.LT.29.5) IRING = 32
  IF (F.GE.29.5 .AND. F.LE.49.0) IRING = 33
ENDIF
IF (SECTION.EQ.12.0) IRING = 34
IF (SECTION.EQ.13.0) IRING = 35
IF (SECTION.EQ.14.0) IRING = 36
IF (SECTION.EQ.15.0) IRING = 37
IF (SECTION.EQ.16.0) IRING = 38
IF (SECTION.GT.16.0 .AND. SECTION.LT.42.0) THEN
  IRING = 22 + IFIX(SECTION)
ENDIF
IF (SECTION.EQ.42.0) THEN
  DUM23 = (F - 669.0)/(375.0/23.0)
  IDUM24 = IFIX(DUM23)
  IRING = 64 + IDUM24
ENDIF

NOW THAT I HAVE THE RING NUMBER OF THE POINT, I MUST
FIND THE ANGULAR POSITION OF THE POINT D, E, F
IN ORDER TO FIND THE CORRECT SURFACE ELEMENT

IF (D.GT.0.0 .AND. D.LT.0.00001) D = 0.00001
IF (D.LE.0.0 .AND. D.GT.-0.00001) D = -0.00001
IF (E.EQ.0.0 .AND.
  SECTION.GE.38 .AND. SECTION.LE.39.0) E = -0.0001
TANGLE = ATAN(E/D)
IF (D.LT.0.0) TANGLE = TANGLE + PI
IF (E.LT.0.0 .AND. D.GT.0.0) TANGLE = TANGLE + 2*PI
IF (SECTION.GE.36.0 .AND. SECTION.LE.39.0) TANGLE = TANGLE - 0.1
DUM28 = TANGLE/(15.0*(PI/180.0))
TANGLE = IFIX(DUM28) + 1
IF (SECTION.GE.14.0 .AND. SECTION.LE.17.0) THEN
  TANGLE = NINT(DUM28) + 1
ENDIF
IDUM28 = (IRING - 1)*24
ISURF = TANGLE + IDUM28
RETURN
END

THIS SUBROUTINE DETERMINES WHETHER OR NOT THE RAY
WAS ABSORBED OR REFLECTED. THE INPUT TO THE SUBROUTINE
INCLUDES THE SURFACE ELEMENT WHERE THE RAY STRUCK,
AND THE OUTPUT IS CASE, EITHER 1.0, 2.0, OR 3.0 (WHERE
1.0 IS AN ABSORPTION, 2.0 IS A DIFFUSE REFLECTION, 3.0
IS A SPHERICAL REFLECTION

SUBROUTINE AB(ISURF, ALP, XAP, SEED, CASE)
DIMENSION ALP(2064), XAP(2064)
INTEGER SEED
CASE = 0
IF (ISURF.LE.0.OR.ISURF.GT.2064) THEN
    ISURF = 1
ENDIF
CALL CHAN(SEED,RND)
RND3 = RND
CALL CHAN(SEED,RND)
RND4 = RND
IF (RND3.LE.ALP(ISURF)) CASE = 1.0
IF (RND3.GT.ALP(ISURF)) THEN

DECIDING WHETHER REFLECTION IS SPECULAR OR DIFFUSE

    IF (RND4.LT.XAP(ISURF)) THEN
        DIFFUSE REFLECTION
            CASE = 2.0
        ENDIF
    ENDIF

SPECULAR REFLECTION
    CASE = 3.0
ENDIF
20 CONTINUE
RETURN
END
Vita

David D. Chapman was born in Washington, D.C., and lived in Montclair, VA, until 1987 when he graduated from Potomac High School. He received an appointment to the U.S. Air Force Academy, Colorado Springs, CO, and entered the Academy in June, 1987. In May, 1991, David was commissioned a second lieutenant in the U.S. Air Force and also earned the degree of Bachelor of Science in Engineering Sciences. While attending the Academy he held various positions, including flight commander, operations officer, and honor officer. In the summer of 1990, he worked for the Strategic Defense Initiative Organization (SDIO) Headquarters at the Pentagon.

In August, 1991, David began pursuing a Master of Science degree in Mechanical Engineering at Virginia Polytechnic Institute and State University, Blacksburg, VA, as part of the U.S. Air Force Academy’s Graduate Scholarship Program. He worked there under the guidance of Dr. J.R. Mahan. David began the U.S. Air Force’s Undergraduate Pilot Training in October, 1992, at Columbus AFB, MS.

David D. Chapman