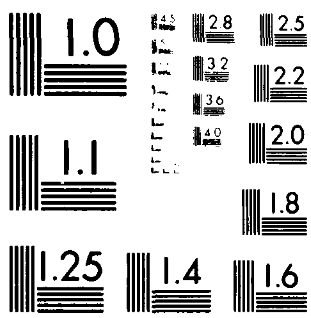
AD-A117 651 FAILURE ANALYSIS ASSOCIATES PALO ALTO CA F/G 21/8.2
PROBABILISTIC ENVIRONMENTAL MODEL FOR SOLID ROCKET MOTOR LIFE P--ETC(U)
MAR 82 G DERBALIAN, J M THOMAS, P JOHNSTON N60530-78-C-0127
UNCLASSIFIED NWC-TP-6305 NL

1 OF 3
AD
A117651



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

NWC TP 6305

AD A117 ...

Probabilistic Environmental Model for Solid Rocket Motor Life Prediction

by
George Derbalian
Jerrell M. Thomas
Paul Johnston
Gregg Brooks
Failure Analysis Associates
for the
Ordnance Systems Department

MARCH 1982

NAVAL WEAPONS CENTER
CHINA LAKE, CALIFORNIA 93555

Approved for public release; distribution unlimited.

Naval Weapons Center

AN ACTIVITY OF THE NAVAL MATERIAL COMMAND

FOREWORD

The purpose of the project described in this report was to develop a probabilistic environmental model for solid rocket motor life prediction. Methodologies used for constructing an environmental model are described. Movement of rockets from one location to another is simulated by Markov chain theory. Damage caused by thermal stresses is calculated, and a cumulative damage failure criterion is adopted. Specific results are shown for Sidewinder rockets.

This task was performed by Failure Analysis Associates (FAA) for the Naval Weapons Center (NWC), China Lake, under Contract N60530-78-C-0127. This program was supported by the Naval Air Systems Command under the Missile Propulsion Technology Block Program (AirTask A03W-3300/008B/F31300000). Mr. Lee N. Gilbert is the NWC technology administrator for this program. Mr. Ronald Vetter was the program manager.

This report was reviewed for technical accuracy by Mr. Vetter. It is released for information at the working level and does not necessarily reflect the views of NWC.

Approved by
C. L. SCHANIEL, *Head*
Ordnance Systems Department
1 March 1982

Under authority of
J. J. LAHR
Capt., U.S. Navy
Commander

Released for publication by
R. M. HILLYER
Technical Director

NWC Technical Publication 6305

Published by .... Technical Information Department
Collation.. Cover, 109 leaves
First printing................................ 250 unnumbered copies

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER NWC TP 6305	2. GOVT ACCESSION NO. AD-A117251	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) PROBABILISTIC ENVIRONMENTAL MODEL FOR SOLID ROCKET MOTOR LIFE PREDICTION		5. TYPE OF REPORT & PERIOD COVERED
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) George Derbalian, Jerrell M. Thomas, Paul Johnston, Gregg Brooks		8. CONTRACT OR GRANT NUMBER(s) Contract N60530-78-C-0127
9. PERFORMING ORGANIZATION NAME AND ADDRESS Failure Analysis Associates 750 Welch Road, Suite 116 Palo Alto, California 94304		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Weapons Center China Lake, California 93555		12. REPORT DATE MARCH 1982
		13. NUMBER OF PAGES 216
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Environment
Probability of failure
Sidewinder
Solid rocket motor
Statistics
Air-launched tactic

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

See reverse side of this form.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102 LF 014 6601

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

(U) *Probabilistic Environmental Model for Solid Rocket Motor Life Prediction*, by George Derbalian, Jerrell M. Thomas, Paul Johnston, and Gregg Brooks, Failure Analysis Associates. China Lake, Calif., Naval Weapons Center, March 1982, 216 pp. (NWC TP 6305, publication UNCLASSIFIED.)

(U) Temperature changes that solid rocket motors experience while residing in various locations with different climates cause thermal stresses in the propellant. Repeated application of such stresses can cause damage to the rocket propellant, which may result in cracks.

(U) This report describes a probabilistic environmental model for solid rocket motors. Movement of rockets from one station to another is simulated, using a Markov chain technique. A cumulative damage model is used to compile the damage resulting in each rocket location. The rocket external temperature is assumed to be a random variable and expressed in a probabilistic form for each Markov state. As an example, data were collected to represent the environmental model for Sidewinder rockets. The Markov matrix for Sidewinders was based on past motor logistics. Propellant failure is predicted when the cumulative damage exceeds a critical value.

CONTENTS

ACKNOWLEDGMENTS

The authors are grateful for the technical support and management of the Naval Weapons Center, China Lake, California, and, in particular, to Mr. Ken Bischel (deceased) and Mr. Ronald Vetter, who were program managers. The Navy Fleet Analysis Center was helpful by supplying an example of Sidewinder rocket motor history tape. Professor E. H. Lee and Professor W. Knauss contributed to the viscoelastic stress analysis formulation. The authors are thankful for the typing support of Mrs. Shirley Lawder and for the illustrations by Mrs. Jeanne Weir.

SUMMARY

In this report, a methodology for estimating the time to failure of rockets by using probabilistic analysis has been described and illustrated. In particular, a probabilistic model of the thermal environment to which rockets are exposed in real life situations was developed. As an example, the Sidewinder rocket motors were considered, and the realistic thermal environmental and logistic (i.e., movement of rockets from one location to another) data were probabilistically modeled. The methods used are general and can be applied to other rockets. Failure occurs when the rocket motor has exceeded an allowable level of cumulative propellant damage.

In this report, propellant damage was caused by thermal stresses resulting from external temperature only. There are other causes besides temperature for damage in rockets (such as shock and vibration, chemical aging of the propellant, humidity, and radiation). These are not considered in this model; however, it is recommended that these other effects be added into the damage calculations.

The example for Sidewinder rockets was integrated with the model development. The environmental temperature model was developed first. This model is quite general and describes the temperature in a probabilistic manner. The example internal temperature distribution through the motor case and grain was then determined in a sealed motor for a given external temperature variation. Following this, example stresses due to (1) temperature gradients in the rocket propellant and (2) differences in material properties between propellant and casing were determined. A Markov state model was then developed that describes the probabilistic manner in which rockets change locations using various modes of transportation. A damage accumulation model was interfaced with the Markov state model, and numerical results are shown for Sidewinder rockets using past motor logistics data and a representative environmental model.

Damage accumulation was considered for storage, transportation, and captive flight conditions for Sidewinder missiles.

Damage incurred during storage is minimal. Navy storage locations are situated in mild climates with the most climatically severe storage site located at Tokyo/Atsugi. Rockets aboard ships also experience a mild temperature. The surrounding sea acts as a temperature stabilizer and ships do not frequently travel in very cold and icy waters where damage could be higher.

Truck, train, and air transportation is more damaging than storage (by at least two orders of magnitude); and captive flight is more damaging than either storage or transport. During captive flight the rocket is directly exposed to the surroundings; the air at high altitudes is very cold and, hence, can cause a larger amount of damage because of resulting low motor temperatures.

INTRODUCTION

The propellant of solid rocket motors in weapons systems is known to degrade with time. Prediction of the time at which the solid rocket motor is no longer serviceable is extremely important to overall planning of the deployment and replenishment of the weapons system. The weapons system planning is related to the time when early failures in the solid rocket motor fleet are expected, rather than the time when an "average" motor is expected to fail, resulting in a need for probabilistic failure prediction methods. These methods of failure prediction can also be expected to lead to improved specifications for motor procurement.

An accurate description of the environment to which the motor will be subjected is a critical first step in failure prediction. Since environmental factors (such as temperature) and the life cycle of deployed motors are random in nature, a probabilistic treatment of life prediction is essential. The purpose of this project was to develop a probabilistic model of solid rocket motor environments and to demonstrate that the model can be used in life predictions.

One of the main natural causes of rocket motor degradation is thermal stresses in the propellant which can cause initiation and growth of cracks. Rocket motors are frequently transported from one location to another or they are stored in different locations around the world. Each location is characterized by its own unique climate. Rockets could be in covered storage, dump stored, inside aircraft carriers or other ships, air or ground transported, on the deck, or in captive flight. All of these activities have their own representative climates. The rocket external temperature varies due both to diurnal temperature changes and seasonal temperature changes and with position on the surface of the earth. These temperatures and the temperature changes induce thermal stresses in the propellant. Thermal stresses arise, even in a steady state condition (uniform temperature), because of differences in thermal expansion between the propellant and the rocket case and because they are below their stress-free temperature. The most highly stressed locations are generally the case bond and the bore.

In this report, the effect of temperature on rocket motor damage is emphasized, and an example external temperature model for Sidewinder rockets is developed. Although there are other causes for rocket damage such as vibration, chemical aging of the propellant, radiation, humidity, and shock, these effects are not considered here.

The propellant material is frequently characterized by a visco-elastic thermo-rheologically-simple material behavior. The most important feature of this behavior compared to purely elastic behavior is that the relationship between stresses and strains in the material is not unique but is a function of time and temperature.

Substantial test information exists that relates time-to-failure in a propellant to constant applied stress, constant applied strain, or stress applied at constant strain rate. Propellants in real weapons systems are not subjected to constant stresses or strains. Instead, both the stress and strain vary continuously with time, dependent upon the environmental (particularly temperature) fluctuations. Limited information exists to relate these time variant stresses and strains to propellant failure. One relationship that has shown considerable promise as a failure criterion is the linear cumulative damage model proposed by Bills and Wiegand.[1]

Related studies have been done by Cost and Dagen,[2] and Heller,[3] and more recently by Cost.[4,5,6] These investigators made considerable progress in calculating propellant response to probabilistic environments. The present model builds on this previous work by emphasizing the relationship of the environment to motor logistics and the effect on motor failure of cumulative damage due to the environment.

The major different features of this development compared to previous work are:

1. Combination of a previously defined cumulative damage model that is consistent with observed propellant behavior with an environmental model that produces cumulative damage. Propellant failure can be caused by the accumulation of many temperature cycles, even if the critical stress or strain to cause failure in a single cycle is never exceeded.

2. Development of a realistic probabilistic model of the environment that characterizes the deployment of Navy tactical weapons systems. Such an environmental model takes into account the fact that motors in these weapons

systems are frequently moved from one location to another using various modes of transportation.

3. Use of simulation techniques that can readily calculate the probability distributions of life for the earliest failures in the fleet. It is this information that is important in making decisions on fleet replenishments.

To develop a probabilistic model of solid rocket motor environments with these features, it is important that the eventual use of the model be clearly established. The proposed model is to be used to predict the damage incurred by (and eventual failure of) solid rocket propellants. Therefore, it is essential that the environmental model contain the major effects that result in damage to the propellant. The damage model to be used in this development depends on the amount of time at a particular stress and temperature for all discrete locations in the propellant. The major environmental factors affecting these parameters are temperature external to the motor and changes in this temperature as a function of time. Therefore, emphasis in the model was placed on temperature and temperature changes.

The interaction among elements of motor life prediction is shown in Figure 1. As shown, environmental data and motor logistics data must be combined to formulate the probabilistic model. It is also clear that the model must be in such a form that the propellant stress and environment history can be used with the damage model to make probabilistic predictions of motor life.

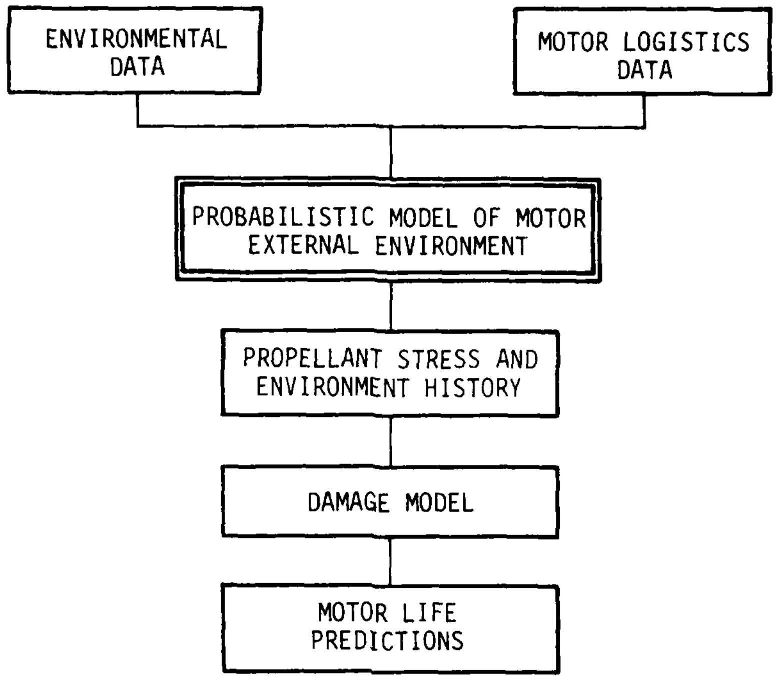


Figure 1 - Relationships in Overall Life Prediction Problem.

ROCKET MOTOR EXTERNAL TEMPERATURE MODEL

The temperature of a rocket motor skin in a given environment is modeled as a random variable which can be characterized by a sinusoidal series consisting of various harmonics. This model can be viewed as a complex Fourier series representation of the rocket skin temperature

$$T_s = T_m + \sum_{i=1}^{I} T_i e^{i\omega_i t} \tag{1}$$

where T_s is the skin temperature, T_m is the mean temperature, T_i is the amplitude of the harmonics, ω_i are the harmonic frequencies, t is time, and I is the number of harmonics considered; j designates $\sqrt{-1}$.

In the present model, the frequencies are assumed to be fixed deterministic values. The amplitudes T_m and T_i are modeled as random variables to be characterized from past recorded temperature data.

Within the propellant, a temperature solution of the form

$$T(\vec{r},t) = T_T(\vec{r},t) + T_m + \sum_{i=1}^{I} T_i R_i e^{j\omega_i t} \tag{2}$$

is expected, where $\vec{r}$ represents the location within the propellant, $T_T(\vec{r},t)$ is a transient term that decays with time, and R_i is a term dependent upon location, ω_i, propellant properties, and geometry, but not upon time.

Suitable analysis methods can be used to solve for stresses versus time under these temperature conditions. Stresses are dependent upon temperature gradients within the propellant, properties and differences in properties of the propellant and case, and the propellant and case geometries. A typical assumption is that propellants behave ideally as viscoelastic media with temperature-dependent relaxation modulus.[7] Stress solutions can generally be expressed as

$$\sigma_{ij} = \sigma_{ij}(\vec{E}_c,\vec{E}_p,\vec{r},t) \tag{3}$$

where σ_{ij} are the components of stress, $\vec{E}_c$ is a vector of case properties, $\vec{E}_p$ is a vector of propellant properties, $\vec{r}$ represents location, and t is time. For most propellants, the properties $\vec{E}_p$ must be taken as time and temperature dependent for realistic analysis (i.e., thermo-viscoelastic analysis).

A power spectrum for temperature exhibits two significant, narrow peaks centered at the frequencies of the seasonal and diurnal cycles[8] as shown in Fig. 2. Therefore, the temperature representation model can be simplified by retaining only the diurnal and seasonal frequencies in the Fourier series.

Hence, the temperature is expressed by

$$T_s = T_m + T_a \sin \omega_a t + T_d \sin \omega_d t \tag{4}$$

where T_m is the mean temperature, T_a and T_d are the amplitudes of the annual and diurnal cycles, respectively, and ω_a and ω_d are the annual and diurnal angular frequencies given by

$$\omega_a = 2\pi F_a = \frac{2\pi}{365x24} \text{ hour}^{-1} \tag{5}$$

$$\omega_d = 2\pi F_d = \frac{2\pi}{24} \text{ hour}^{-1} \tag{6}$$

with F denoting the cyclic frequency.

This analytic form of the external temperature is convenient for solving the heat equation in closed form and for determining the temperature distribution throughout the rocket cross section.

The temperature model assumed in this analysis and described by Eq.(4) would give a discrete frequency spectrum as shown in Fig. 2. Therefore, the deterministic frequency model is an idealization of the actual temperature representation. Although the frequencies of the temperature cycles are assumed to be fixed deterministic values, the amplitudes of the seasonal and diurnal cycles, T_a and T_d, are to be represented probabilistically. As an example, Sidewinder rockets are considered, and the specific temperature amplitudes T_a, T_d are determined for different Sidewinder locations.

EXAMPLE EXTERNAL TEMPERATURE MODEL FOR SIDEWINDER ROCKETS

Sidewinders are five inch diameter rockets used in air-to-air operations. These rockets are found in a variety of environments and conditions around the world. The average life of a Navy Sidewinder rocket is spent as follows (see NWC TP 4464, Part 1, p.15[9]).

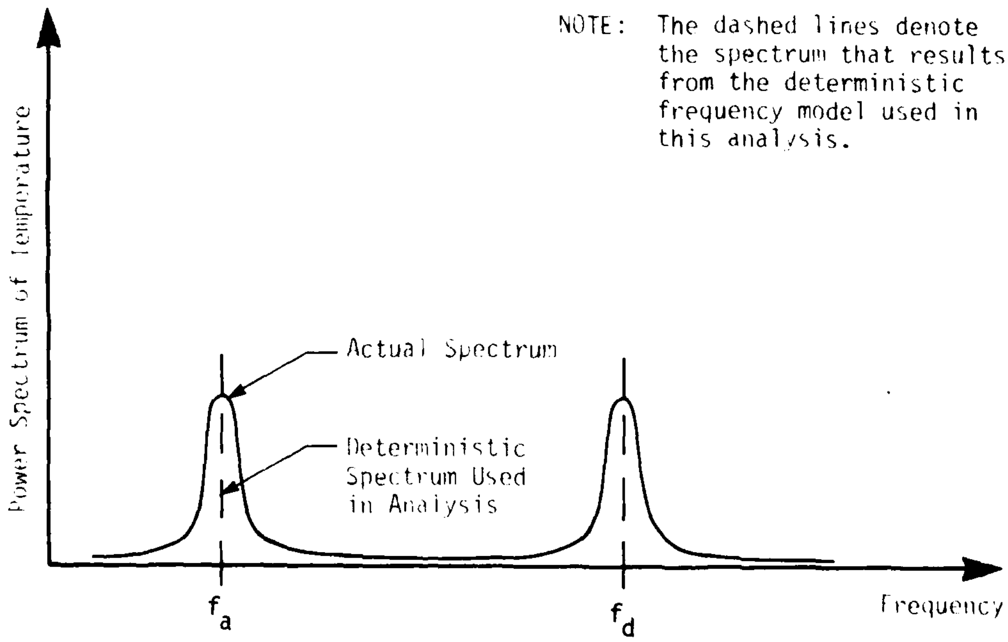


Figure 2 -- Schematic Ambient Temperature Power Spectrum

Location	Condition	Portion of Life Spent
Storage	covered	87%
	exposed/dump	3% (lower during peacetime)
Transportation	air	0.82%
	ground	4.15%
Ammunition and Combat ship		5%
Captive flight *		0.03%

There are different climates unique to each of these main categories. Table 1 shows possible locations throughout the world where Sidewinders are found. These include both continental USA and overseas bases, ships (such as aircraft carriers, ammunition ships) and others. Rockets are frequently transported from one of these locations to another; transportation can be either by sea, air or ground (train or truck). In the following sections the applicable external temperature will be developed for each of these locations. That is, the parameters describing the external temperature (Eq.(1)) will be determined.

---

* This datum was determined from the Navy Fleet Analysis Center's RM History tape.

Table 1 - Sidewinder Locations

1.	Yorktown, VA		Ships
2.	Subic Bay, RP	32.	Kitty Hawk CV-63
3.	Israel	33.	Independence CV-62
4.	Fallbrook, CA	34.	JFK CV-67
5.	Agana, Guam	35.	Midway CV-41
6.	Coral Sea, CV-43	36.	Constellation CV-64
7.	Subic Bay NAVMAG	37.	Ranger CV-61
8.	Concord, CA	38.	Enterprise CVN-65
9.	Yuma, AZ	39.	Oriskany CV-34
10.	Seal Beach, CA	40.	Shasta AE-33
11.	Miramar, CA	41.	Santa Barbara AE-28
12.	Dallas, TX	42.	Suribachi AE-21
13.	Kaneohe, HI	43.	Saratoga CV-60
14.	Atsugi, Japan	44.	Butte AE-27
15.	Oceana, VA	45.	Nimitz CVN-68
16.	El Toro, CA	46.	Forrestal CV-59
17.	Roosevelt Roads, PR	47.	Flint AE-32
18.	NAHA, Okinawa	48.	Hull
19.	Kadena	49.	Mt. Baker AE-34
20.	Point Mugu, CA	50.	Wabash AOR-5
21.	Sigonella, Italy	51.	America CV-66
22.	Beaufort, SC	52.	F.D. Roosevelt CV-42
23.	Norfolk, VA	53.	Detroit AOE-4
24.	Singapore	54.	Nitroh AE-23
25.	Da Nang, Vietnam	55.	Eisenhower CVN-69
26.	Nellis AFB, NV	56.	Canisteo AO-99
27.	Rota, Spain	57.	Camden AOE-2
28.	Cherry Point, NC	58.	Haleakala AF-25
29.	Key West, FL	59.	Kiska AF-35
30.	Iwakuni, Japan	60.	Hancock CV-19
31.	Nam Phong		

Sidewinder Storage Location External Temperature

The National Oceanic and Atmospheric Administration (NOAA) in Asheville, North Carolina, has an extensive data base of temperatures* for many locations both in the United States and overseas. Temperature data are recorded hourly or every three hours and for several years.

Typically, the data span a ten to twenty year period or longer. A temperature computer tape for each of the Sidewinder ground locations was obtained from NOAA.

A computer program** called WEATHER was developed at Failure Analysis Associates (FAA) which analyzes the NOAA weather tapes and produces probabilistic distributions of the mean annual temperature T_m, seasonal temperature amplitude T_a, and diurnal temperature amplitude T_d.

WEATHER initially computes daily mean temperatures. From these, monthly means are obtained and then the annual mean is obtained by calculating the means of the months. A deterministic value of T_m is used since the annual mean temperature has a very small standard deviation.

Cumulative density functions (CDF)[10] were found for the annual temperature, T_a and the daily temperature amplitude, T_d. The annual temperature amplitude for a given year was assumed to be half the difference between the highest and lowest monthly mean tempera-

---

* NOAA's weather tapes contain additional weather information besides ambient temperatures.

** All computer programs here are written in FORTRAN and are listed in Appendix D.

tures. The diurnal temperature amplitude for a given day was assumed to be half the difference between the highest and the lowest daily temperatures. The cumulative density functions for the amplitudes were constructed from the frequency of the size of the amplitudes. Typically, the seasonal temperature amplitude is found to have a smaller deviation than the diurnal temperature amplitude. Figures 3 and 4 describe a typical CDF for the seasonal and diurnal temperature amplitudes for Point Mugu, California. Notice the wider variation of the diurnal temperature amplitudes.

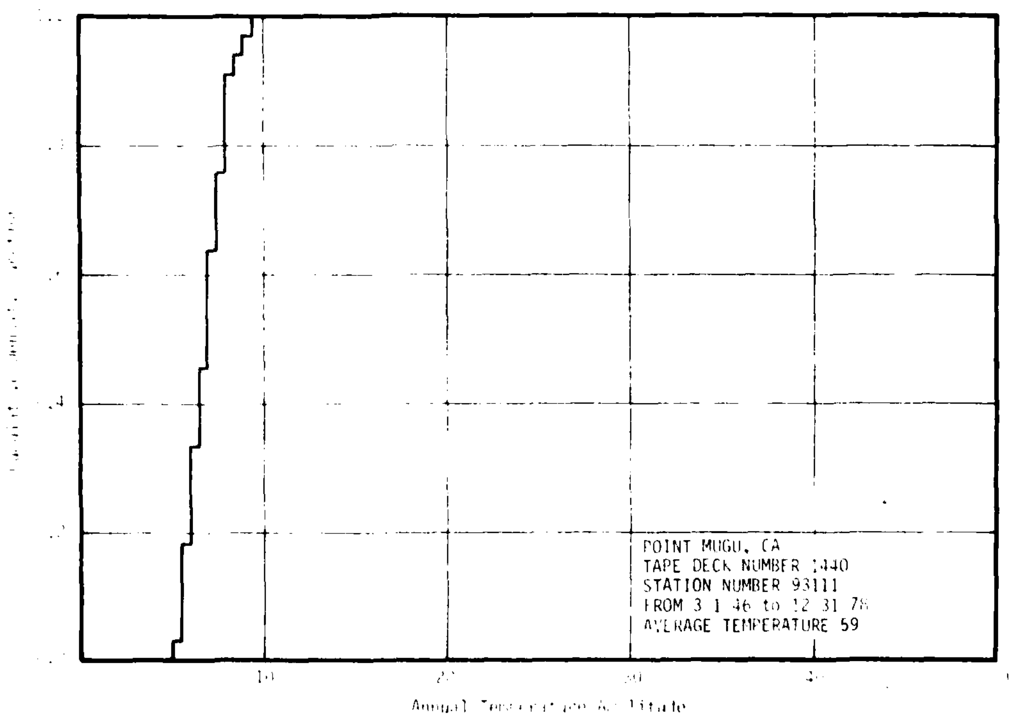


Figure 3 - Probabilistic Annual Temperature Amplitude.

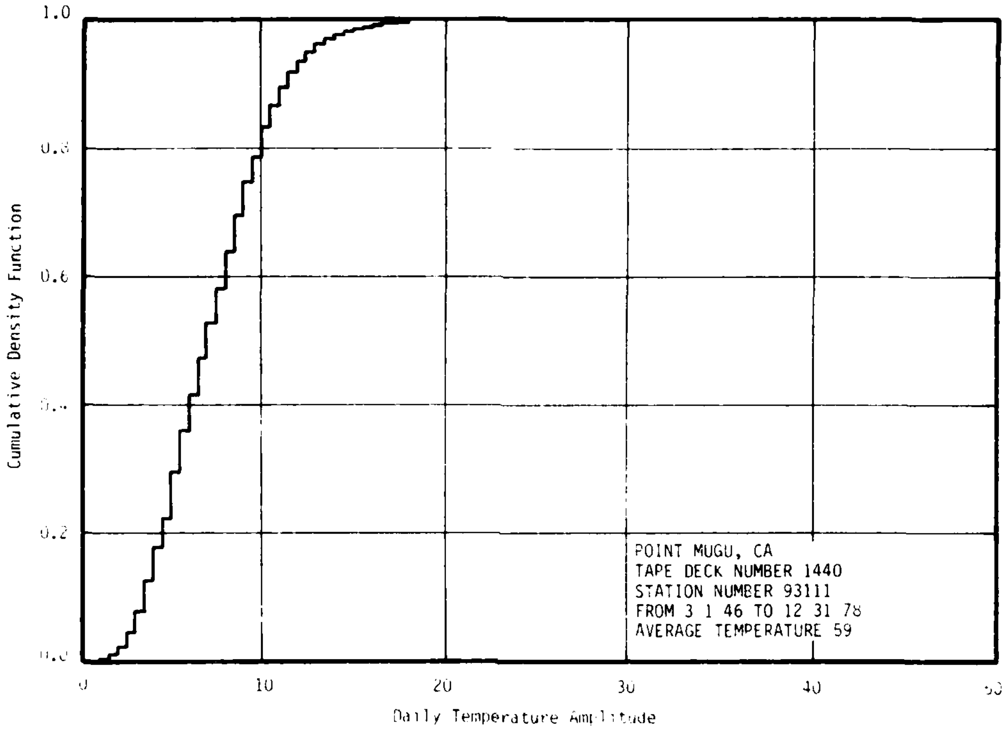


Figure 4 - Probabilistic Daily Temperature Amplitude.

During the Monte Carlo simulation for a rocket motor in storage, a random number between zero and one is generated to select the annual temperature amplitude. This determination is achieved by using the CDF appropriate for that storage location. The same process is repeated to determine the diurnal temperature amplitude. The actual data for the storage location are used directly without attempting to fit a parametric model to the temperature distribution. The random number generator program RANDK[11] is available at the Stanford University computer.

The complete seasonal and diurnal CDFs for all the storage locations are shown in Appendix A. A digital form of these data is used later in the damage calculation.

Approximately 98% of the stored rockets are in covered type storage and the rest are dump stored or not covered (NWC TP 4464, Part 1, p. 15[9]). Rockets in covered storage are well insulated. Therefore, the rocket skin will experience a small temperature variation compared to the ambient air temperature fluctuations. The effect of the insulation will be modeled by introducing seasonal and diurnal amplitude scale factors, f_a and f_d, in the following storage temperature representation formula:

$$T_{storage} = T_m + f_a T_a \sin \omega_a t + f_d T_d \sin \omega_d t \tag{7}$$

An approximate value of f_a, based on earth covered storage temperature data in Oahu/Hawaii, is 0.66 (see NOTS TP 4143, Part 2, p.5).[12] An estimate of f_d could not be inferred from that same report. It is expected that f_d will be smaller than f_a and close to zero.

External Temperature for Ship Transport

Temperature data for ordnance carried on-board ships are available but limited to only a few ships. However, data from many levels are collected. Since these ships are constantly in motion they will experience different climates not only because of seasonal changes but also because the ships move to different locations on the oceans. Therefore, temperature history for a given ship is not very meaningful unless its location and season are known. Such data will be very difficult to construct. Instead of developing an external temperature input for each ship, we can construct a single global

temperature model for all ships, assuming the available temperature data constitute a representative sample. The form of the external rocket temperature will be the same as Eq.(4), i.e.,
$T = T_m + T_a \sin \omega_a t + T_d \sin \omega_d t$, whereby the probabilistic amplitudes T_a and T_d and the mean temperature T_m are determined from the cumulative density functions (CDF) of all the ships for which temperature data are available. Howard Schafer of NWC, China Lake uses a single parameter probabilistic CDF but his method cannot account for seasonal and/or diurnal temperature cycles (see Fig. 36, NWC TP 4824, (Ref. 13) for composite temperature of all ships, all levels). A computer program AIRCARRY was written that determines the CDFs of seasonal and diurnal temperature amplitudes from the raw temperature data of rockets inside ships.

Temperatures from different levels in ships and different carriers were used. The source of the data is:

USS Franklin D. Roosevelt	CVA -42
USS Kitty Hawk	CVA -63
USS Enterprise	CVAN -65
USS Shangri-La	CVA -38

A total of 4638 days were considered in compiling the temperature CDF data. The resulting CDF for the diurnal temperature amplitude is shown in Figure 5. Note that the daily temperature variation is generally small. Since the temperature data used came from only two different years, a probabilistic representation of seasonal temperature amplitude was not available and a deterministic mean value of 4°F was assumed. When temperature data from other years becomes available, a probabilistic seasonal temperature amplitude can be constructed.

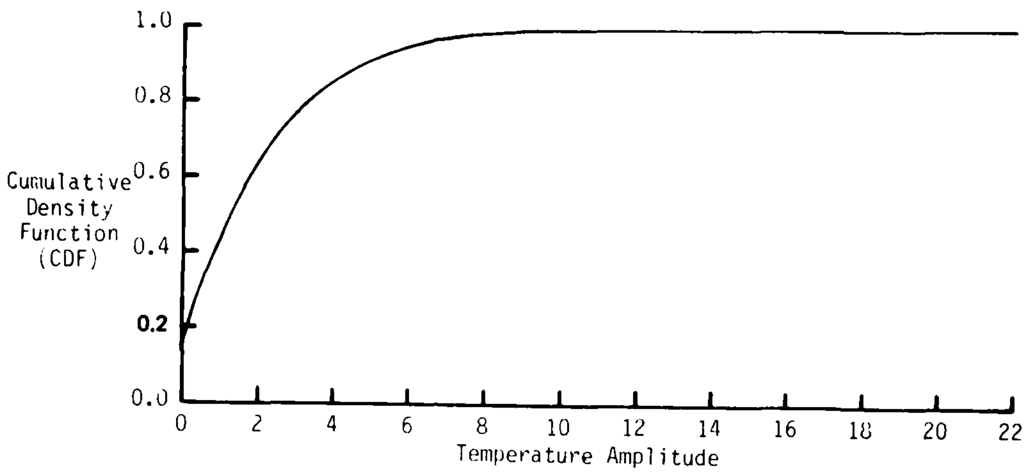


Figure 5 - Diurnal Temperature Amplitude CDF.

Captive Flight External Temperature Model

Measurements of skin temperature of rockets during captive flight[14] indicate that with the exception of take-off and landing, external temperature of ordnance is quite uniform. Figures 6 and 7 show typical temperature vs. time plots of Sidewinder rockets during captive flight.

A majority of captive flights are flown at low to medium altitude where the ambient air temperature is mild. A small percentage of flights, however, are flown at high altitudes (40,000 to 50,000 feet) where temperatures as low as -70°F can develop on the external skin of rockets. Because of the limited data during captive flight at this time, a complete probabili-tic model for external temperature of rockets cannot be established. Instead, a simplified model is used, represented by a constant temperature during each flight, with the temperature randomly selected from an assumed Gaussian distribution. A mean temperature of 30°F and a standard deviation of

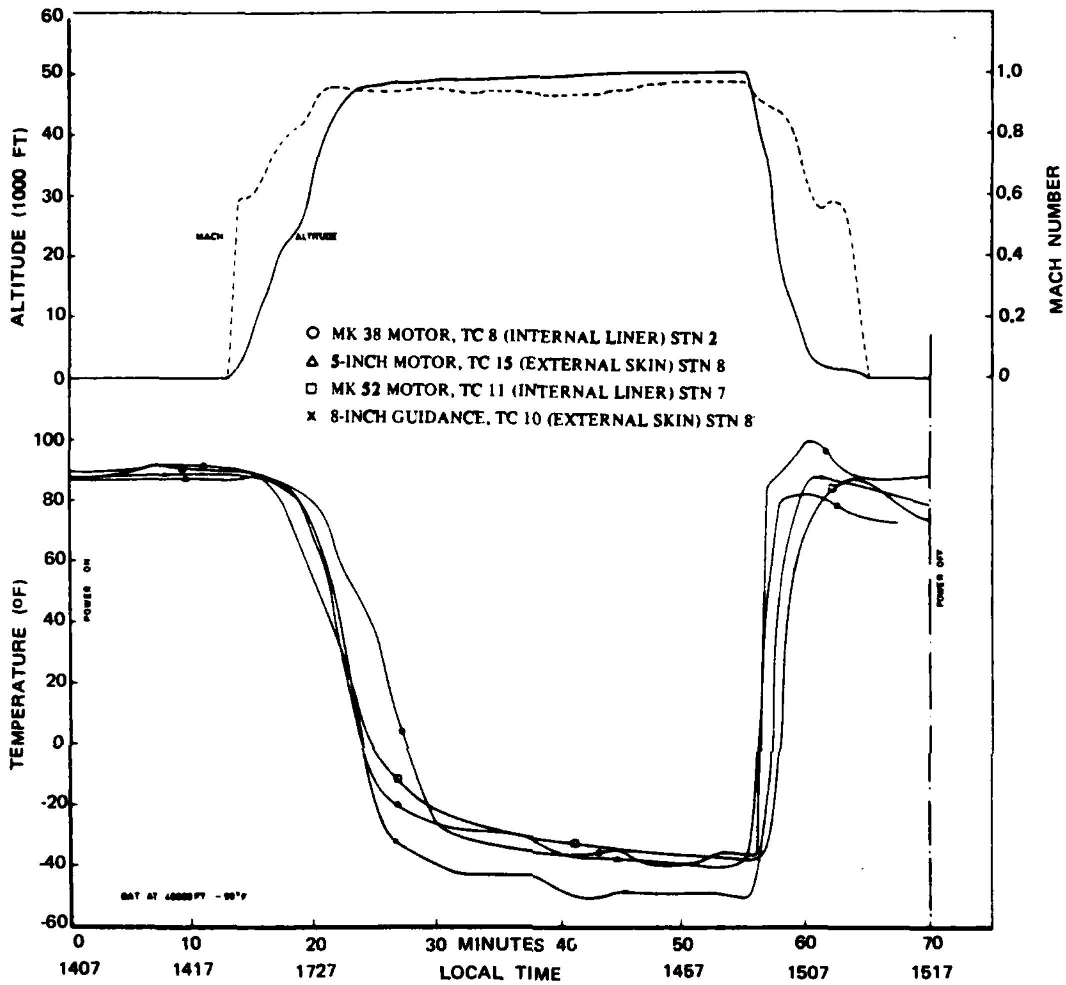


Figure 6 - Flight Profile of a Typical High-Altitude Loiter Flight (from NWC TP 5365).[14]

20°F are assumed. The average duration of a captive flight is approximately two hours.

External Temperature of Sidewinders During Transportation

Approximately 83% of Sidewinders are ground transported and 17% air transported. A similar temperature model is used for air transportation as in captive flight; that is, a uniform external temperature is assumed during the flight. However, the mean temperature and standard deviation will be different than for captive flight. Limited information exists in NWC TP 4828[15] for temperatures of air transported ordnance.

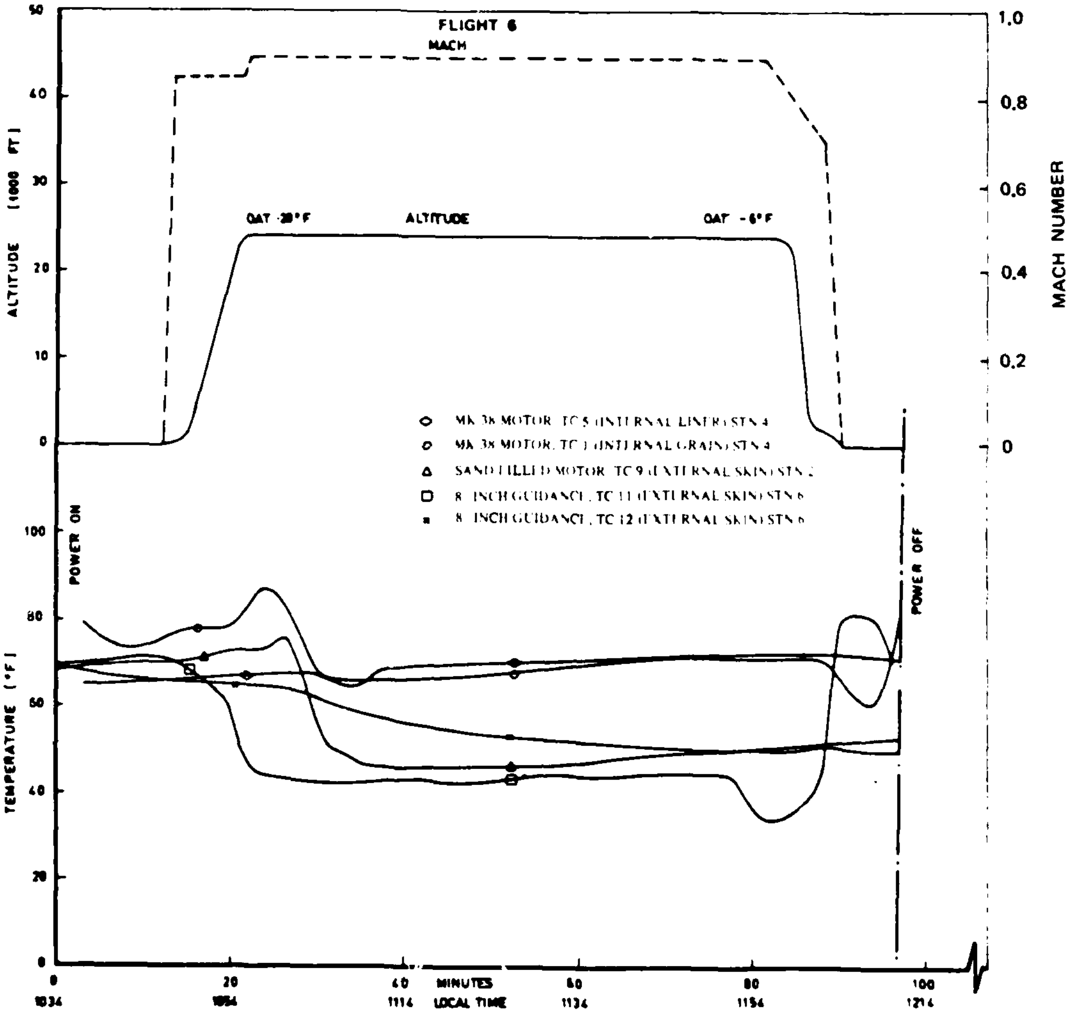


Figure 7 - Flight Temperature Profiles During Medium-Altitude Flight (from NWC TP 5365).[14]

During air transportation a Gaussian distribution with a mean temperature of 60°F and a standard deviation of 10°F was assumed (see Figs. 12-17 of NWC TP 4828[15]).

During ground transportation (truck or train), the external temperature varies due to the effect of diurnal temperature cycles and also due to changing location (i.e., changing climates). It would be a formidable task to precisely trace the complete external temperature variation of rockets during ground transportation. Therefore, a model based on the temperatures characteristic of the departure and arrival locations is used. Hence, the transportation period is divided equally between departure and arrival locations. Obviously, a more exact model would incorporate several climates that the truck or train will encounter during the entire course of travel.

During ground transportation, the appropriate rocket external temperature is the ambient air temperature of the departure and arrival locations but scaled down because of some insulation provided by the truck or train shell. This insulation is typically less effective than an earth-covered storage insulation. Also, when the rocket is removed from storage and taken to a truck or train it will be exposed to ambient temperature (i.e., it is uninsulated). This period is short and furthermore, if the temperature is very severe, the transportation may be delayed to avoid unreasonably foul weather.

It is necessary to determine appropriate scaling factors, f_d and f_a, for the amplitudes in the characterization of the rocket skin temperature during ground transportation. That is,

$$T_{\text{ground transportation}} = T_m + f_a T_a \sin \omega_a t + f_d T_d \sin \omega_d t \qquad (8)$$

where T_a and T_d are the ambient temperature amplitudes. A scaling factor for the diurnal cycle is assumed to be 0.63 (see NWC TP 4822,

Fig. 8).[16] The seasonal temperature amplitude is not scaled down (i.e., $f_a = 1$) because the duration of the transportation is generally short compared to seasonal weather changes.

EXAMPLE TEMPERATURE DISTRIBUTIONS IN A PROPELLANT

As an example, transient temperature and stress distributions were calculated based on the Fourier heat equation for a cylindrically shaped solid propellant. Figure 8 shows a cross section of a propellant and casing, which is assumed to be cylindrical for this example, but could be of any shape in the general applications.

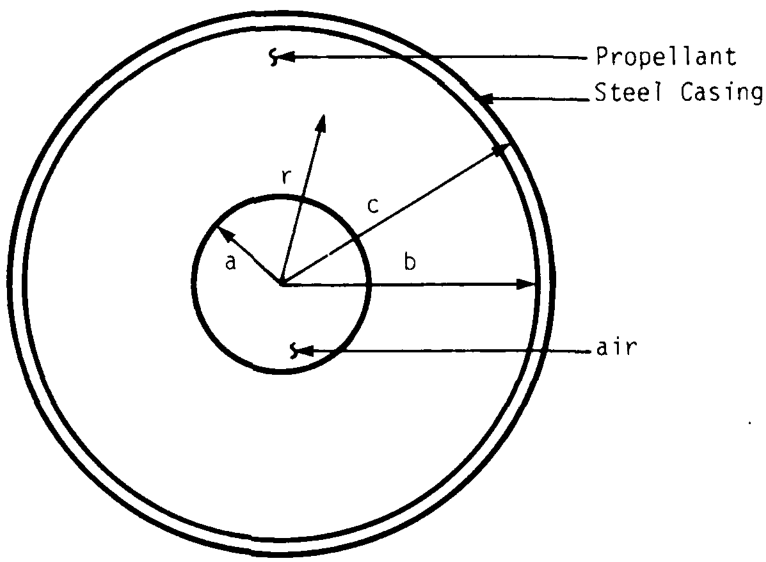


Figure 8 - Schematic of a Rocket Motor Cross-Section

The partial differential equation governing heat conduction in a solid[17] is given by

$$\text{(a)} \quad \frac{1}{k_1}\frac{\partial T_1}{\partial t} = \nabla^2 T_1 = \frac{\partial^2 T_1}{\partial r^2} + \frac{1}{r}\frac{\partial T_1}{\partial r} \qquad r<a \qquad \text{(air)}$$

$$\tag{9}$$

$$\text{(b)} \quad \frac{1}{k_2}\frac{\partial T_2}{\partial t} = \nabla^2 T_2 = \frac{\partial^2 T_2}{\partial r^2} + \frac{1}{r}\frac{\partial T_2}{\partial r} \qquad a<r<b \qquad \text{(propellant)}$$

T_1 is the temperature in the cavity (air) and T_2 is the propellant temperature. k_1 and k_2 are the thermal diffusivities of air and propellant respectively. ∇^2 is the harmonic operator which is expressed here in polar coordinates for convenience, and t denotes time.

The boundary conditions require

$$T_1\,(a,t) = T_2(a,t) \qquad \text{(continuity of temperature)} \tag{10}$$

$$C_1 \frac{\partial T_1(a,t)}{\partial r} = C_2 \frac{\partial T_2(a,t)}{\partial r} \qquad \text{(continuity of heat flux)} \tag{11}$$

C_1 and C_2 are thermal conductivities of air and propellant respectively. For the boundary condition at b, the previously described input from the environment (Eq.(4)) is used.

$$T_2\,(b,t) = T_m + T_a \sin \omega_a t + T_d \sin \omega_d t \tag{12}$$

The steady state solution of the temperature distribution in the propellant is given by

$$
\begin{aligned}
T_2(r,t) = T_m &+ T_a \left[-A_aF_a(r) + B_aE_a(r)\right] \cos \omega_a t \\
&+ T_a \left[A_aE_a(r) + B_aF_a(r)\right] \cos \omega_a t \\
&+ T_d \left[-A_dF_d(r) + B_dE_d(r)\right] \sin \omega_d t \\
&+ T_d \left[A_dE_d(r) + B_dF_d(r)\right] \sin \omega_d t
\end{aligned}
\tag{13}
$$

where A and B are constants and E and F are functions of position. All parameters used here are defined in Appendix B.

Note that to obtain this solution, the temperature of the steel casing was assumed to be approximately constant, i.e., $T(b,t) \approx T(c,t)$. This assumption that the temperatures at the inner and outer radii of the case are approximately the same can be justified because of the relatively large conductivity of the steel case compared to the conductivity of the propellant; furthermore, the thickness of the case is much smaller than the radius of the propellant. The thermal diffusivity of the steel case is about 100 times that of the propellant.

As the time increases and steady conditions prevail, the response frequency to the annual and diurnal inputs will be the same as the "forcing function" frequencies and the solution will be of the form

$$T_{\text{long time}} = T_m + T_aR_a \sin (\theta_a + \omega_a t) + T_dR_d \sin (\theta_d + \omega_d t) \tag{14}$$

where R_a and R_d are not functions of time. θ is a phase shift that depends upon radial location, geometry and properties of the propellant, (Eq.(12)). [No matter how complex the propellant geometry, the solution can be expressed in this form.] If input frequencies other than annual and diurnal become important, more similar terms can be added to Eq.(14). Note that this relationship is not applicable to the captive flight and air transportation conditions because the time is too short for steady state conditions to be realized.

STRESS ANALYSIS

Temperature differences below the stress-free temperature in the rocket (including gradients due to external temperature variations) and differences in properties between propellant and case, cause thermal stresses in the propellant.

For a case-bonded rocket motor, the thermal stresses and strains have been developed[18] for elastic media. The stresses in the propellant grain (under plane strain conditions) may be expressed by

$$\sigma_r = p' + \frac{\alpha E_e}{1-\nu}\left[\frac{1}{b^2}\int_0^b r\,T\,dr - \frac{1}{r^2}\int_0^r r\,T\,dr\right] \tag{15}$$

$$\sigma_\theta = -p' + \frac{\alpha E_e}{1-\nu}\left[\frac{1}{b^2}\int_0^b T\,r\,dr + \frac{1}{r^2}\int_0^r T\,r\,dr - T\right] \tag{16}$$

$$\sigma_z = -2\nu p' + \frac{2\nu\alpha E_e}{(1-\nu)}\left[\frac{1}{b^2}\int_0^b T\,r\,dr - \frac{T}{2}\right] \tag{17}$$

$$p' = \frac{E_e\left[2\alpha\,\frac{(1+\nu)}{b^2}\int_0^b T\,r\,dr - \alpha_c(1+\nu_c)\;T(b,t)\right]}{(1+\nu)\;(1-2\nu) + (1-\nu^2{}_c)\,\frac{bE_e}{h_cE_c}} \tag{18}$$

where α and α_c are the coefficients of thermal expansion, ν and ν_c are Poisson's ratios, E_e and E_c are the elastic modulii of the propellant and the case, respectively, and h_c is the thickness of the motor case.

The term containing E_e and E_c in the interlaminar pressure term, Eq.(18), can be eliminated because $E_c \gg E_e$. A significant portion of the thermal stress in the propellant is caused by the difference in expansion properties between the steel case and the propellant. This stress is expressed by the interlaminar pressure term. The remaining terms proportional to E_e in Eqs. (15) through (17) indicate thermal stresses due to temperature gradient in the propellant.

Solid rocket motor propellants are represented to behave (ideally) as viscoelastic media whose relaxation modulus[7] is temperature dependent. Therefore, a viscoelastic stress analysis is preferable. It has been shown in many cases that a viscoelastic response will be approximately equal to an elastic solution wherein elastic constants are replaced by time-dependent creep or relaxation functions.[19] Hence, the elastic stress solution of a long cylinder with a case should be converted into an equivalent viscoelastic solution by replacing the elastic modulus E with the viscoelastic relaxation modulus. The relaxation modulus is given in terms of the reduced time ξ[20] which is computed by integrating the time-temperature shift factor up to the current time.

A viscoelastic solution can be derived from the above solution by replacing the elastic modulus E_e with the relaxation modulus for each stress component. Hence

$$\sigma_v = E(\xi)\,\frac{\sigma_e}{E_e}$$

where $E(\xi)$ denotes the relaxation modulus; σ_e is the elastic solution for a given stress component; E_e is the elastic modulus for the rocket propellant; σ_v is the corresponding viscoelastic stress component; and ξ is the reduced time[20] given by

$$\xi = \int_0^t \frac{d\tau}{a_T(T(r,\tau))}$$

where a_T is the shift factor which is a function of temperature and consequently a function of time. Typical curves for relaxation moduli and time temperature shift factors are shown in Figures 9 and 10.

Although this viscoelastic approximation is good for monotonic loading conditions, when cyclic thermal stresses exist, this type of approximation is not as accurate. Therefore, a rigorous thermoviscoelastic solution is needed. Because of the lengthy computation required during a Monte Carlo simulation procedure, an elastic stress analysis which is faster and less costly was used. When an efficient and reliable viscoelastic program becomes available, the overall approach has been devised so that it could be substituted.

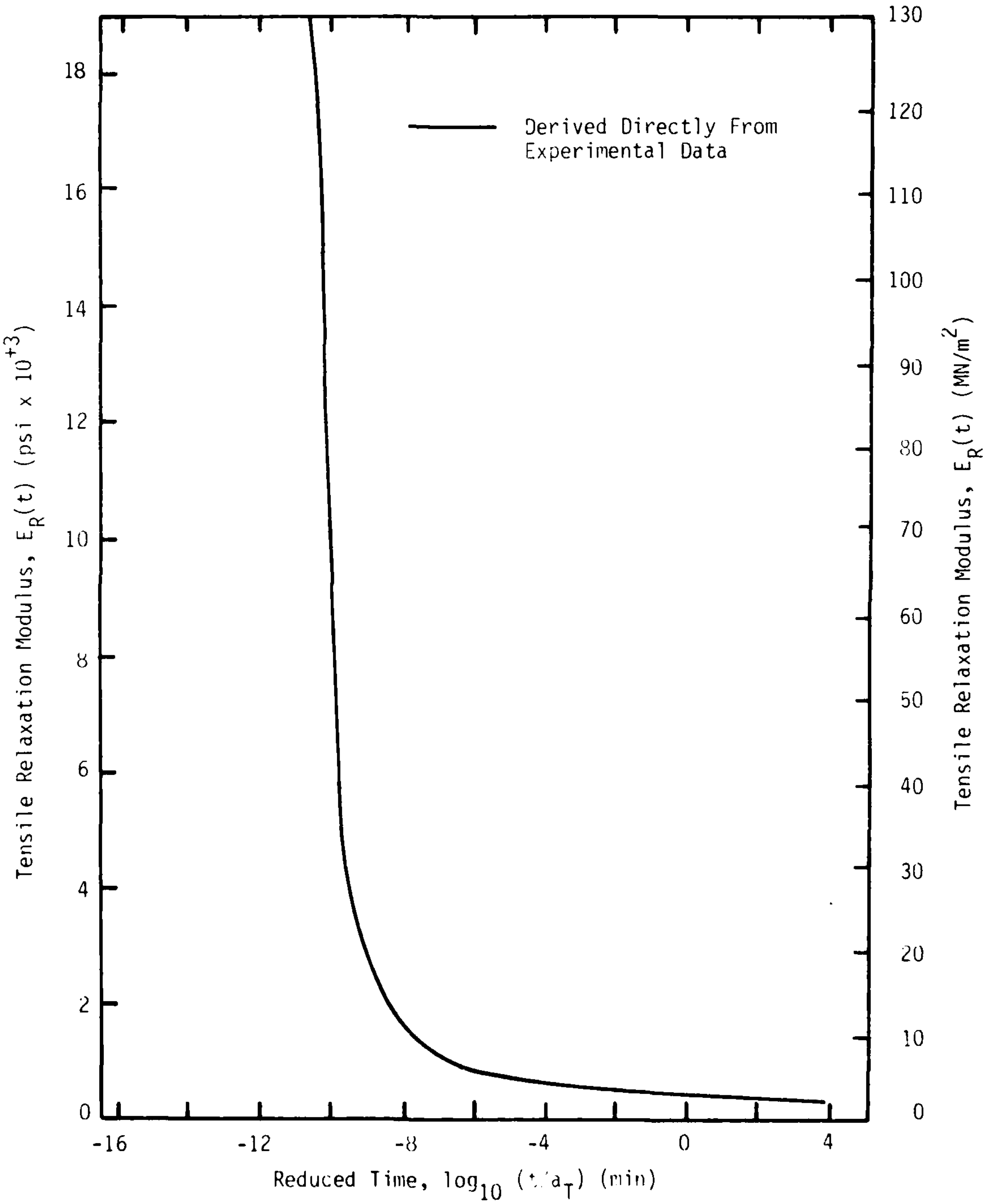


Figure 9 - Tensile Relaxation Modulus for GBP Propellant.

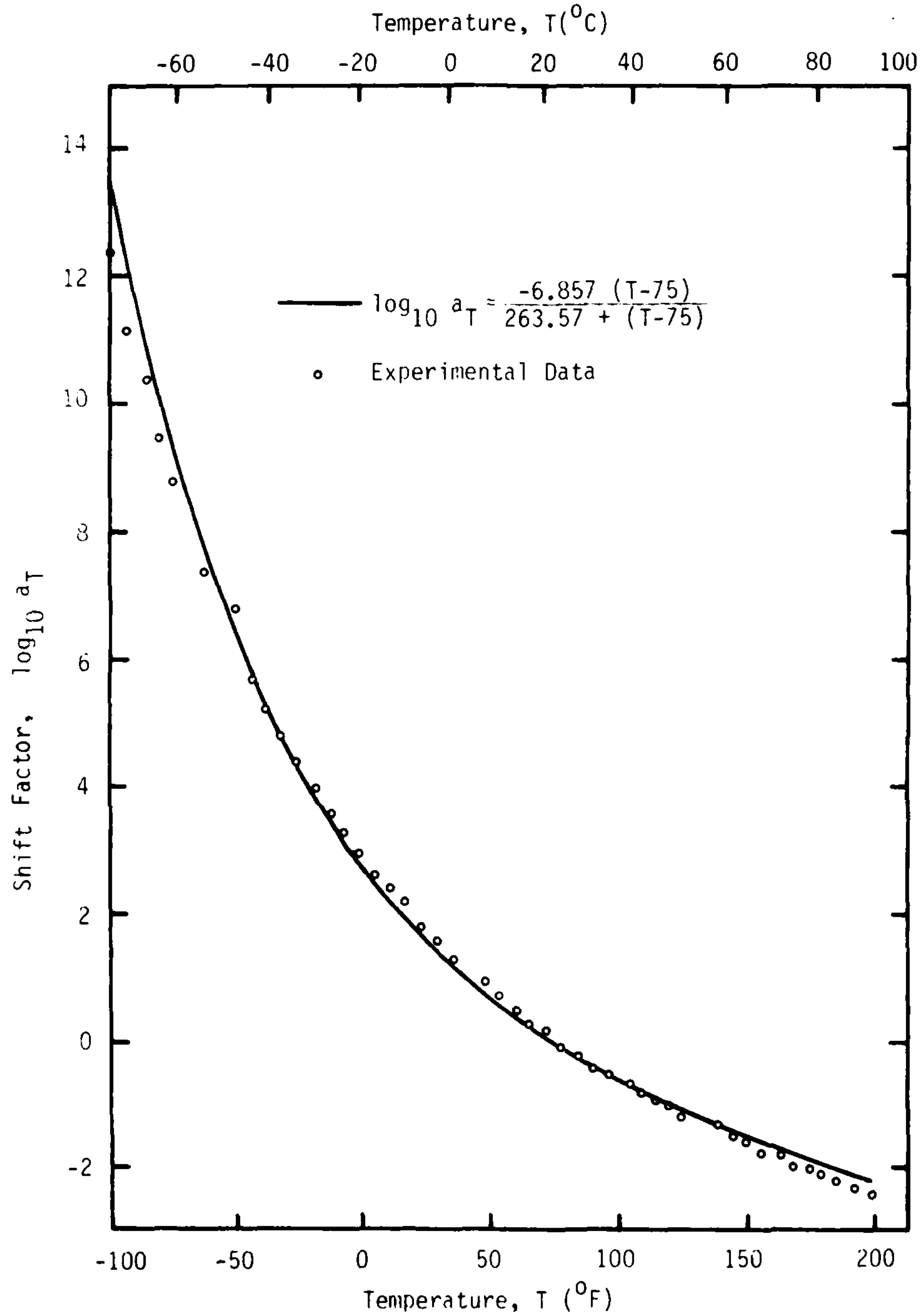


Figure 10 - Shift Function for GBP Propellant.

MARKOV STATE MODEL

A Markov model is a probabilistic engineering concept which describes a system subject to change in state. Such a model is used as the principal tool in combining environmental data with motor logistics to form the probabilistic model of motor external environment. A Markov model containing the essential features needed to represent solid rocket motor temperature environment has been developed and executed.

Markov models are a function of two random variables, the state of the system and the time of observation. An illustration of the state of the system for solid rocket motors is shown in Figure 11. For example, the state could be either storage in a moderate climate or aboard ship in an arctic climate. Substates within the states are also used in the model. These substates are illustrated as environments 1, 2, 3, etc., in the desert storage state. Each environment (substate) is described in the model by a temperature (probabilistically chosen) versus time trace (see section titled "Rocket Motor External Temperature Model"). The basic concept is that, at any given time, any particular solid motor will reside in one of the state spaces.

Any Markov model is defined by a set of probabilities, P_{ij}, which define the probability of transition from state i to state j. For example, in Figure 11, if shipboard/tropics* is labeled as state 3 and aircraft/tropics is labeled as state 7, then P_{37} is the probability that a motor which is on-board ship in the tropics at time t

---

* For Sidewinder rockets the shipboard data combine all climates. Therefore, no distinction is made among tropical, arctic or moderate climates.

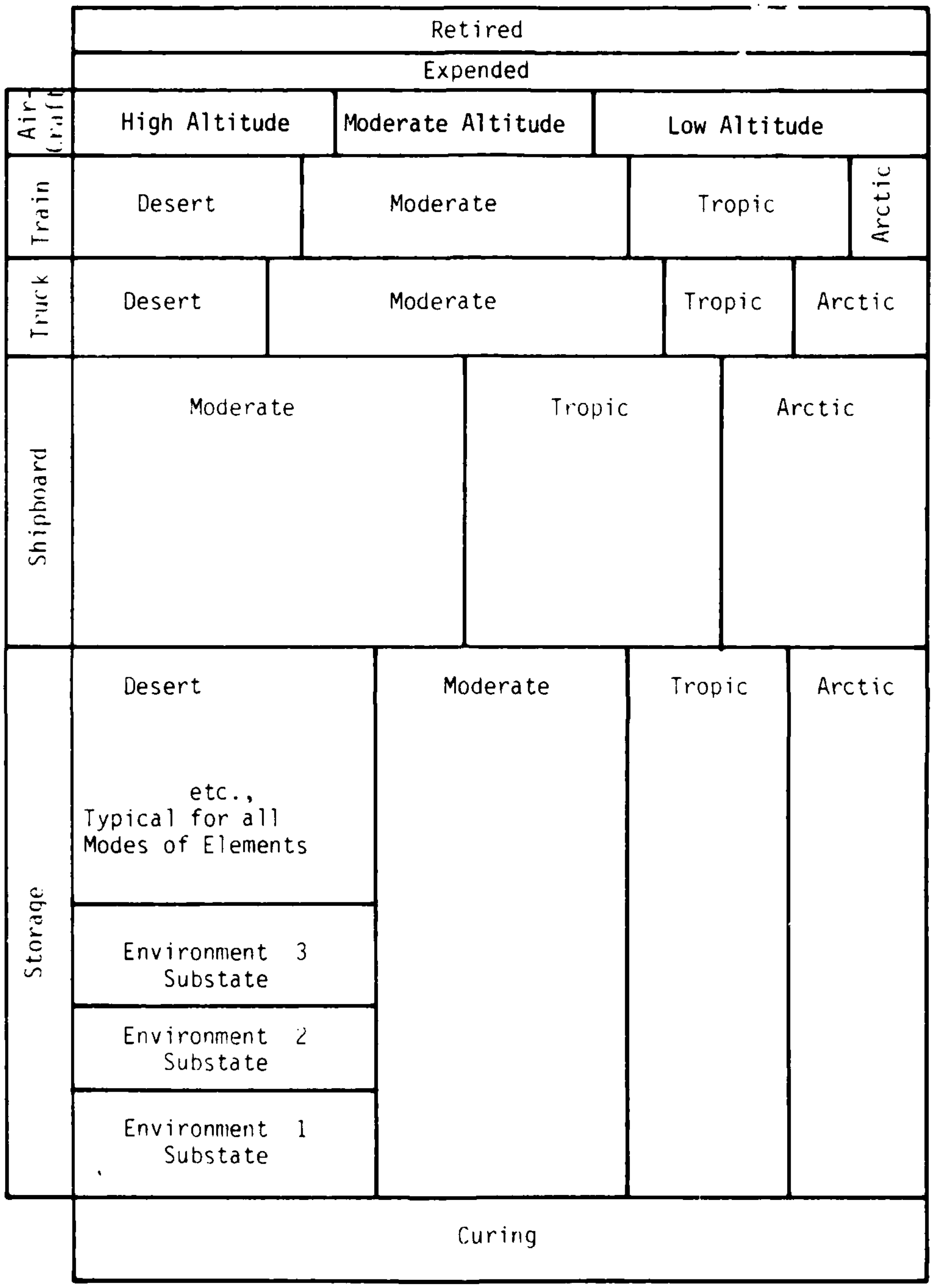


Figure 11 - Schematic of State Space for Markov Model.

will be on-board an aircraft in the tropics at time $(t + \Delta t)$. Note that P_{ii} is the probability that the system will remain in its present state. Also note that, in this context, the substates as defined above can be regarded mathematically as states. The transition probabilities can be conveniently displayed in matrix form as shown in Figure 12.

$$\begin{bmatrix} P_{11} & P_{12} & P_{13} & \cdots & P_{1n} \\ P_{21} & P_{22} & P_{23} & \cdots & \\ P_{31} & P_{32} & P_{33} & \cdots & \\ \vdots & \vdots & \vdots & & \\ \vdots & \cdots & P_{ij} & \cdots & \\ \vdots & & \vdots & & \\ P_{n1} & & & & \end{bmatrix}$$

P_{ij} = Probability of passage from state i to state j.

Figure 12 - Transition Matrix for Markov Model.

Obviously, in a practical Markov model of solid rocket motor environments, many of the P_{ij}'s will be zero because it will be physically impossible to reach some states from certain other states. Other special cases are that in the expended or retired states (denoted as absorbing states), $P_{ii} = 1$, and $P_{ij} = 0$, $i \neq j$.

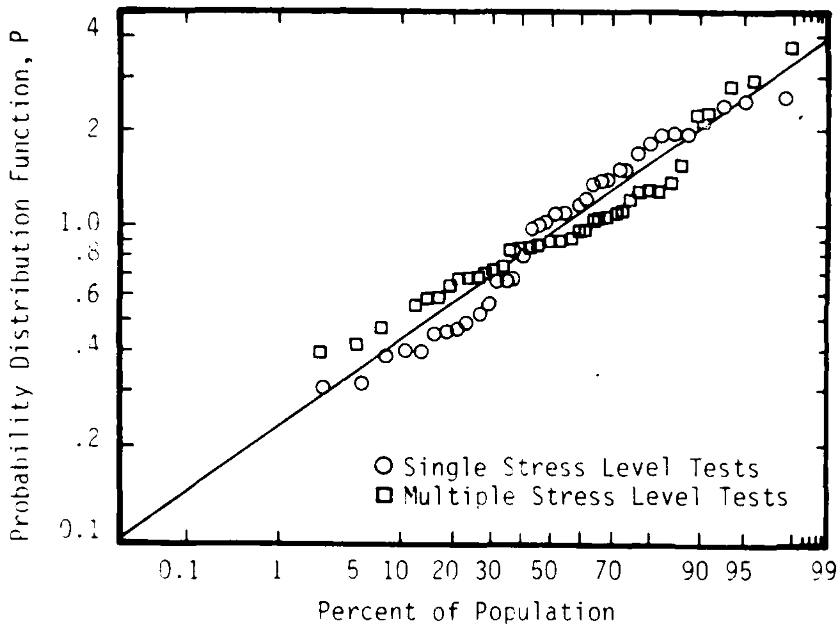
In applying this Markov model to solid rocket motors, it may be important to provide for damage accumulation during transfer to another location. This would require refinements to the present modeling to add transient to the steady-state temperature and stress solutions.

The basic model is general enough to apply to any solid rocket motor. However, the transition probabilities, P_{ij}, are dependent upon the particular weapons system and are, therefore, regarded as input based on motor logistics data.

Motor failure is assumed to be governed by linear cumulative damage as expressed by Bills and Wiegand.[1] However, as also demonstrated by Bills and Wiegand, the cumulative damage required to produce propellant failure is not deterministic, but should be regarded as a random variable.

It will be assumed that the damage tolerance of solid motors is a random variable which can be characterized by a parameter such as P (see Figure 13) in the Bills and Wiegand damage equation. The damage accumulation model will be described in detail in the section titled "Damage Model."

The model presented above is particularly adaptable to Monte Carlo simulation, and this is the approach used here. The Monte Carlo approach has the maximum generality that can be included in probabilistic models. The simulation proceeds essentially as follows:



Log-Normal Distribution (standard deviation in log space ≈ .25)

Figure 13 - Effect of Number of Stress Levels on Normalizing Parameter P

1. A motor (with damage tolerance characteristics selected at random from a probability distribution based on available data) begins in the curing state, and the cumulative damage in curing is calculated according to the Bills and Wiegand damage model and recorded.

2. At the end of the curing cycle, the motor passes to one of the other states, selected randomly in accordance with the transition probabilities, P_{ij}. The damage while residing in the second state is calculated and added to the curing damage.

3. At the end of a preselected time interval, the motor passes from the second state to a third state, randomly selected in accordance with the transition probabilities P_{ij}. Damages in passing to and within this third state are calculated and added to the previous damage.

4. The process in Step 3 is repeated until damage is sufficient for motor failure or until it, by chance, passes to one of the absorbing states (expended or retired). When the motor fails, its time to failure is recorded.

5. Another motor (with damage tolerance characteristics selected at random) begins in the curing state, and Steps 1 through 4 are repeated.

6. Step 5 is repeated until a sufficient number of motors have been simulated to establish a probability distribution of time to motor failure.

In the next subsection, the Markov model will be specialized for Sidewinder rockets.

MARKOV MODEL FOR SIDEWINDER ROCKET MOTOR

The probability matrix for the Sidewinder rocket motors was determined by using past h story of the rocket fleet beginning in 1972. This information was furnished by the Fleet Analysis Center (FLTAC) of the Naval Weapons Station, Seal Beach, Corona Annex, Corona, California. The FLTAC's Sidewinder tape contains information on rocket locations and on dates of arrival and departure from each location, and it describes whether there are captive flights and their durations. Appendix C shows the complete transition matrix. There are a total of 68 independent locations plus captive flight and transportation Markov states. A list of Sidewinder locations was indicated earlier, in Table 1.

Table 2 shows a typical example of the transition probabilities from one of the states (NWS, Yorktown Ord Department) to the other locations. These numbers represent one of the Markov matrix rows. Note that a rocket that is located in Yorktown would have the highest probability of moving to location 9, which corresponds to USS-JFK CV-67, and the probability of this transition is P = 0.157.

A computer program LOGISTIC was developed that analyzed the digital logistics data and produced the probability matrix for the Sidewinder rocket fleet. This program is written in general form and can be used for other missile systems as well, so long as there is a recorded history of past rocket movements. For new systems, analysts should be able to construct an approximate matrix from knowledge of anticipated system deployment.

Table 2 - Transition Probabilities from NWS Yorktown to Other Locations

Location	Transition Probability	Location	Transition Probability
1	0.0	36	0.0
2	0.004301	37	0.0
3	0.032258	38	0.002151
4	0.008602	39	0.010753
5	0.0	40	0.017204
6	0.002151	41	0.002151
7	0.023656	42	0.0
8	0.0	43	0.002151
9	0.156989	44	0.047312
10	0.017204	45	0.0
11	0.008602	46	0.0
12	0.0	47	0.010753
13	0.010753	48	0.0

Table 2 - Transition Probabilities from NWS Yorktown to Other Locations (cont'd.)

Location	Transition Probability	Location	Transition Probability
14	0.006452	49	0.015054
15	0.004301	50	0.0
16	0.025806	51	0.0
17	0.0	52	0.0
18	0.004301	53	0.0
19	0.004301	54	0.0
20	0.006452	55	0.004301
21	0.0	56	0.0
22	0.0	57	0.004301
23	0.0	58	0.0
24	0.010753	59	0.002151
25	0.008602	60	0.0
26	0.06 2366	61	0.0
27	0.0	62	0.0
28	0.101075	63	0.0
29	0.049462	64	0.0
30	0.030108	65	0.0
31	0.002151	66	0.0
32	0.017204	67	0.0
33	0.002151	68	0.0
34	0.0	69*	0.281720
35	0.0		

DAMAGE MODEL

In applying the Markov model to solid rocket motors, the damage due to thermal stresses is determined while a rocket motor resides in a particular state, as well as when the motor passes from one state to another. Limited information exists to relate these time variant stresses and strains to propellant failure. One relationship that

*Location 69 designates captive flight.

has shown considerable promise as a failure criterion is the linear cumulative damage model proposed by Bills and Wiegand.[1] For a number of discrete, constantly imposed stresses this relation is

$$D = \frac{1}{P} \sum_{i=1}^{n} \frac{\Delta t_i}{t_{fi}} \tag{19}$$

where

D is the cumulative damage

P is the normalizing term used to define the probability distribution of failures

Δt_i is the time the specimen is exposed to the i^{th} stress level

t_{fi} is the time to failure if the specimen is exposed to only the i^{th} stress level.

The fact that the cumulative damage to cause failure is a random phenomenon is taken into account by the parameter P. A demonstration of the form of the distribution of P and the applicability of the cumulative damage equation was given by Bills and Wiegand,[1] as shown in Figure 13. Note the importance of treating the damage to cause failure as a random variable compared to using the deterministic value of P = 1. Although each rocket motor has its unique critical damage characteristic, many of the rockets will always remain together when travelling from one Markov state to the other. This fact can eliminate some of the computation cost because fewer than the total number of rockets in a fleet need to be considered in the Markov process.

Bills and Wiegand[1] also suggested the following relationship for determining the time to failure, t_f, of a propellant under constant stress.

$$t_f = a_T t_o \left(\frac{\sigma_{to} - \sigma_{cr}}{\sigma_t - \sigma_{cr}} \right)^B \tag{20}$$

where

t_f is the mean time-to-failure of the specimen when held under a constant "true" stress, σ_t

σ_t is the "true" stress applied to the specimen

t_o is the unit value of the time for whatever units are used in measuring t_f

σ_{to} is the true stress required to fail the specimen in the time t_o

σ_{cr} is the critical true stress, below which no failures are observed

a_T is the time-temperature shift factor

B is a constant

A typical curve illustrating this relationship from Bischel and Wiegand[21] is shown in Figure 14.* When Eqs. (19) and (20) are combined, and a_T and σ_t vary continuously with time and temperature, the cumulative damage is given by

* An exponent, B, of 9.3 (derived from N-29 propellant) was used for Sidewinder rockets.

$$D = \frac{\int_0^t \frac{(\sigma_t - \sigma_{cr})^B}{a_T(t)}\,dt}{Pt_o(\sigma_{to} - \sigma_{cr})^B} \tag{21}$$

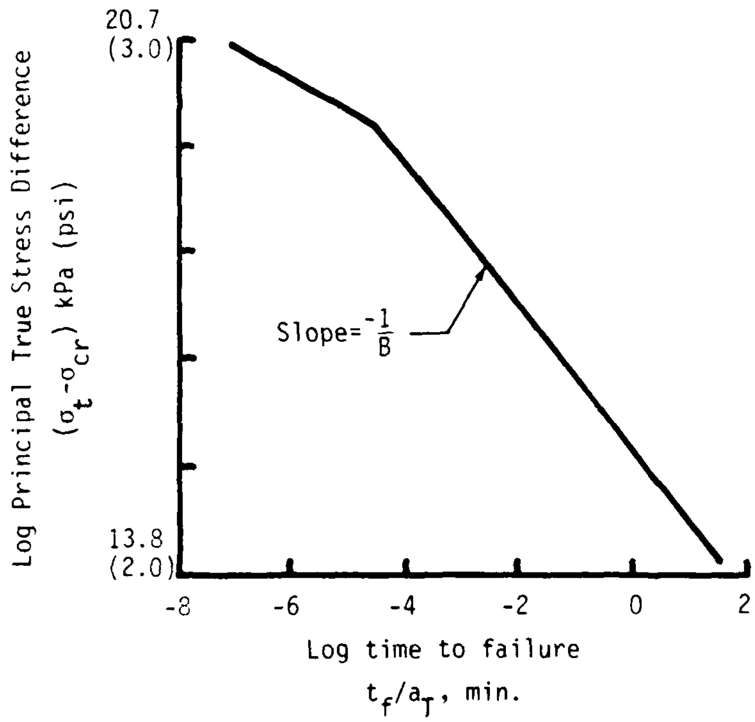


Figure 14 - Maximum Principal Failure Stress Curve for Typical Propellant

The stress applied to a point in the propellant of a motor and the temperature at the same point are largely functions of the external temperature fluctuations (environment) applied to the motor. Both the stress and temperature are important, since the factor a_T is strongly dependent on the temperature. Figure 15 from Cost and Dagen[2] illustrates a typical environmental temperature history for a one-year period. In addition to these seasonal fluctuations, the diurnal cycle may also be important. The attenuation and delay of the temperature response at several locations r in a typical motor propellant from Heller[3] is shown in Figure 16. Typical stresses in two propellant locations due to diurnal temperature cycling are illustrated in Figure 17, also from Heller.[3]

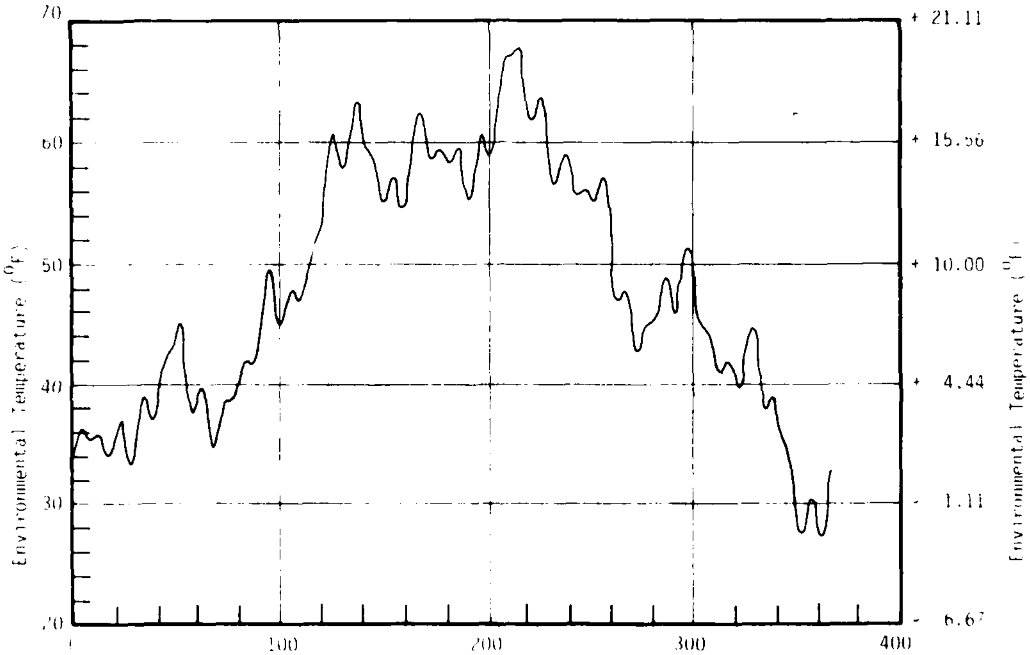


Figure 15 - Environmental Temperature History for One Year at Huntsville, Alabama

From the foregoing descriptions of the propellant behavior, cumulative damage, environmental fluctuations, and response of the propellant, it is evident that an adequate description of the environment is an essential step in predicting motor service life. Since the environment itself and the traverse of a fleet of motors through the environment are random in nature, it is mandatory that the problem be treated probabilistically.

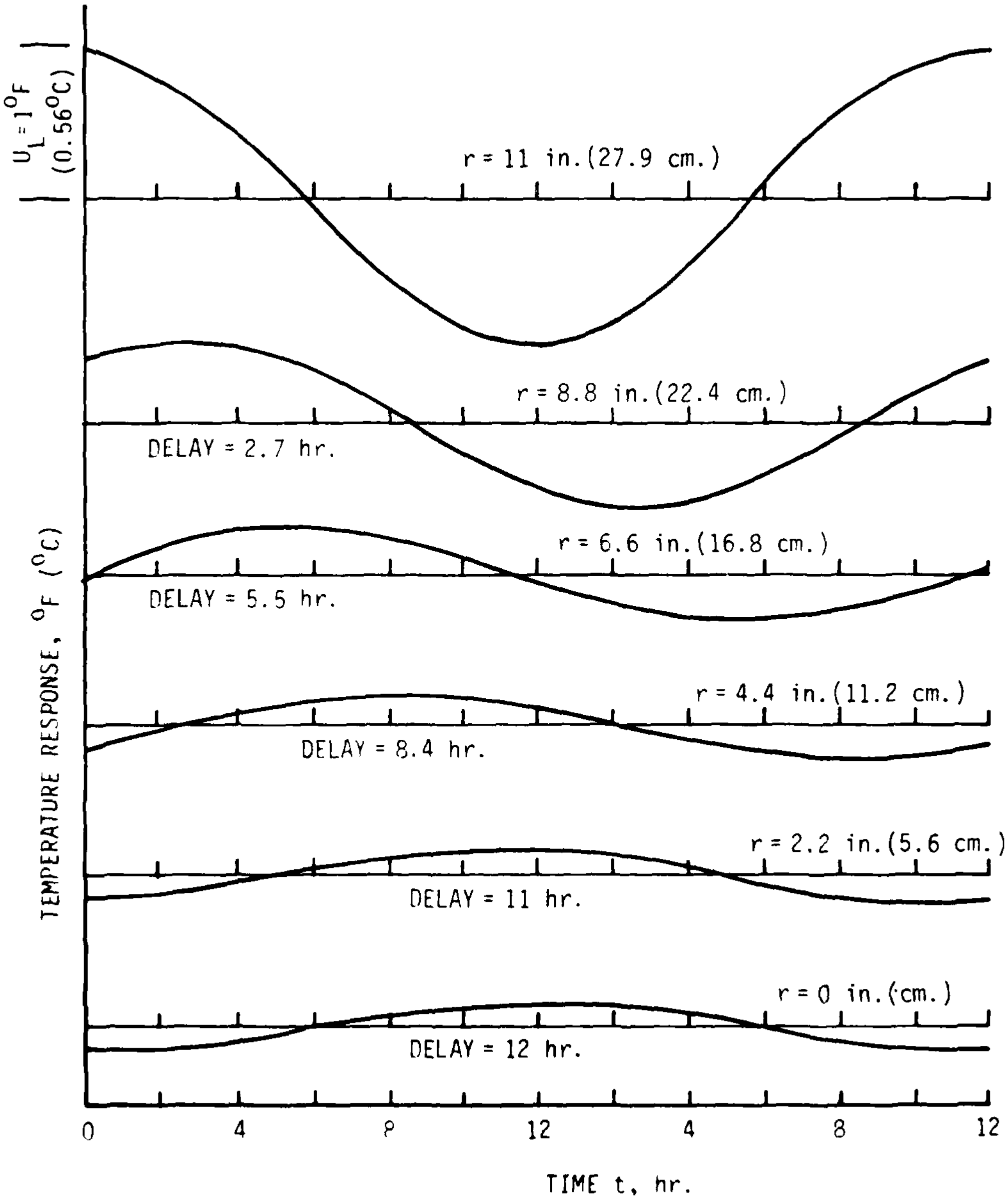


Figure 16 - Attenuation and Delay of Temperature Cycle.

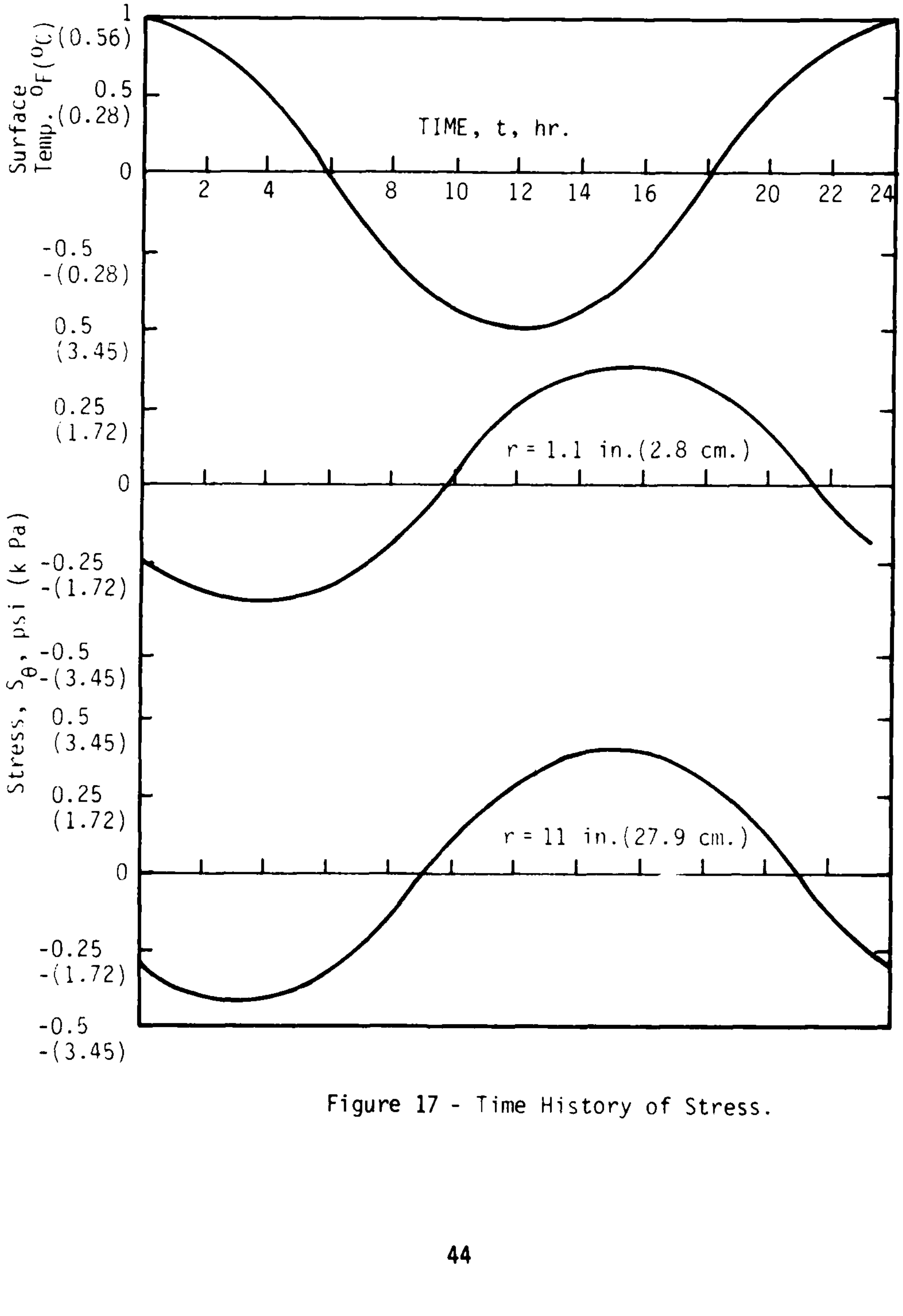


Figure 17 - Time History of Stress.

In the following section, examples will be used to demonstrate the probability distribution of time to motor failure for Sidewinder rocket motors.

NUMERICAL EXAMPLES

Several numerical examples are given here which demonstrate the probabilistic damage calculations. A sensitivity analysis of parameters used in the damage calculation was completed to provide guidance on the relative importance of those parameters. Finally, the model is demonstrated for Sidewinder rockets. The techniques used here can be easily generalized to other rocket systems.

PARAMETRIC SENSITIVITY ANALYSES

The cumulative damage formula described earlier in the section entitled "Damage Model" (becomes, in the deterministic case, P = 1, with consistent units):

$$D = \int_0^t \frac{(\sigma - \sigma_{cr})^B dt}{a_T(\sigma_{to} - \sigma_{cr})^B}$$

uses several parameters (B, σ_{to}, σ_{cr}, a_T) which depend on the material properties of the propellant. For Sidewinders, B was measured to be 9.3 (using uniaxial data) as an average value. Small variations in stresses σ, σ_{to}, σ_{cr} can cause large changes in the integrand and hence the value of the damage. Physically, a large exponent in the damage formula implies a greater influence on damage for larger applied stresses σ. For example, a 100 psi stress applied for one hour is far more damaging than 10 psi stress applied for ten hours (because of the large exponent). Therefore, the influence (on damage) of the stress concentration factor, K_t (a multiplying factor on round bore analyzed stress to obtain real stress) at geometric discontinuities in the propellant can be quite significant.

As expected, the results shown in Figures 18 through 20 indicate that damage is very sensitive with respect to changes in the stress parameters. A small change in one of these parameters σ_{to}, σ_{cr} or K_t causes an exponentially large difference in the value of the damage.

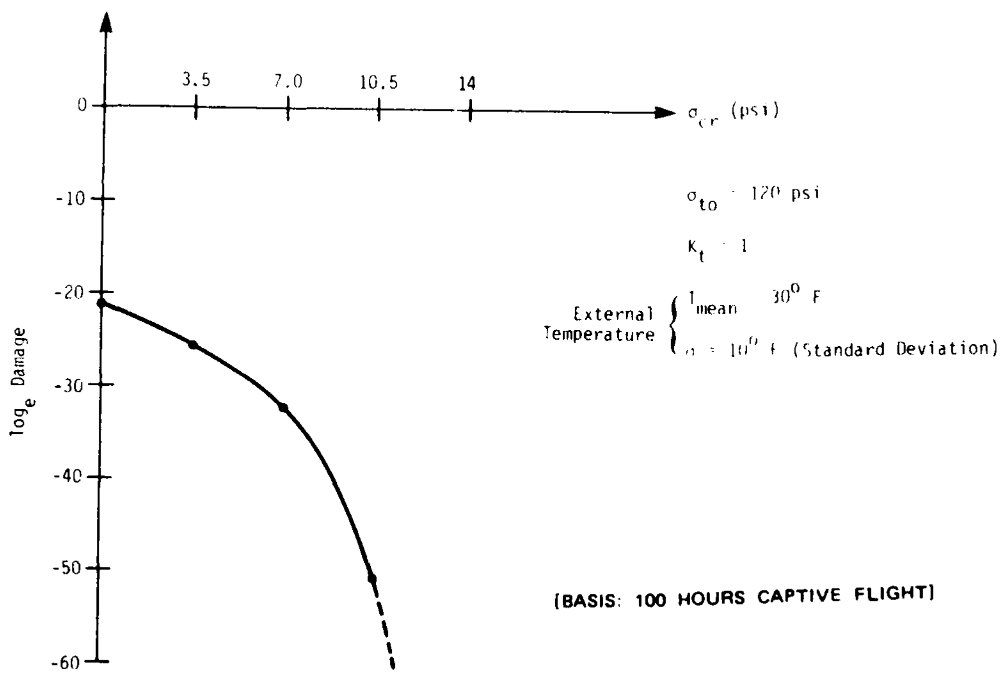


Figure 18 - Damage vs. Critical Stress Below Which No Damage Occurs

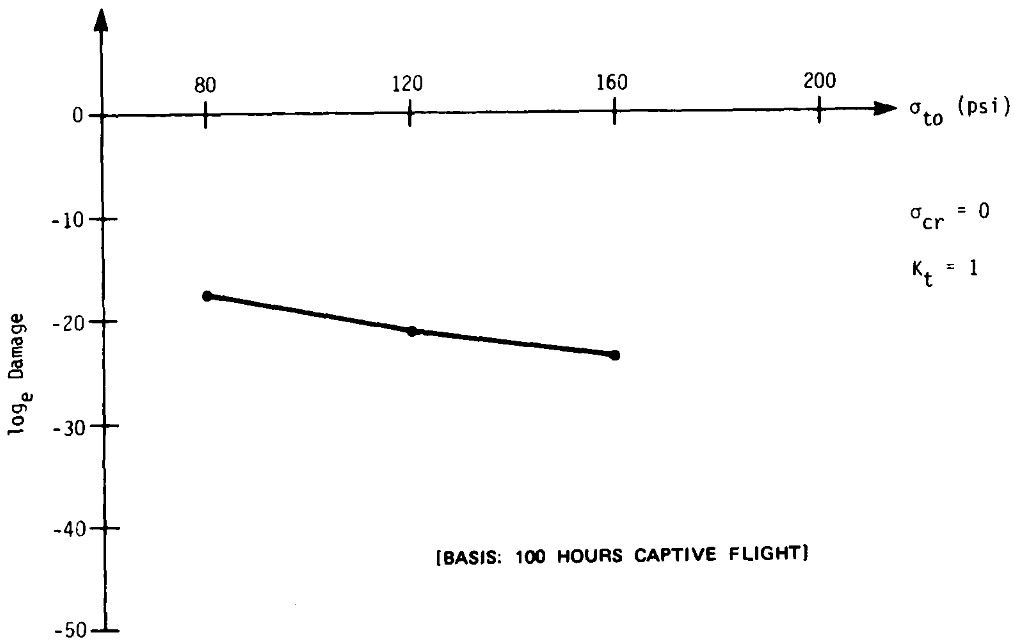


Figure 19 - Damage vs. Stress Required to Fail Specimen in One Minute

The cumulative damage for 100 hours of simulated captive flight was evaluated by varying σ_{to}, σ_{cr} and the stress concentration factor K_t which is applied to the thermal stress solution of a long cylindrical geometry with a cylindrical bore (i.e., plane-strain, axisymmetric, see Figure 8) to obtain the maximum stress.

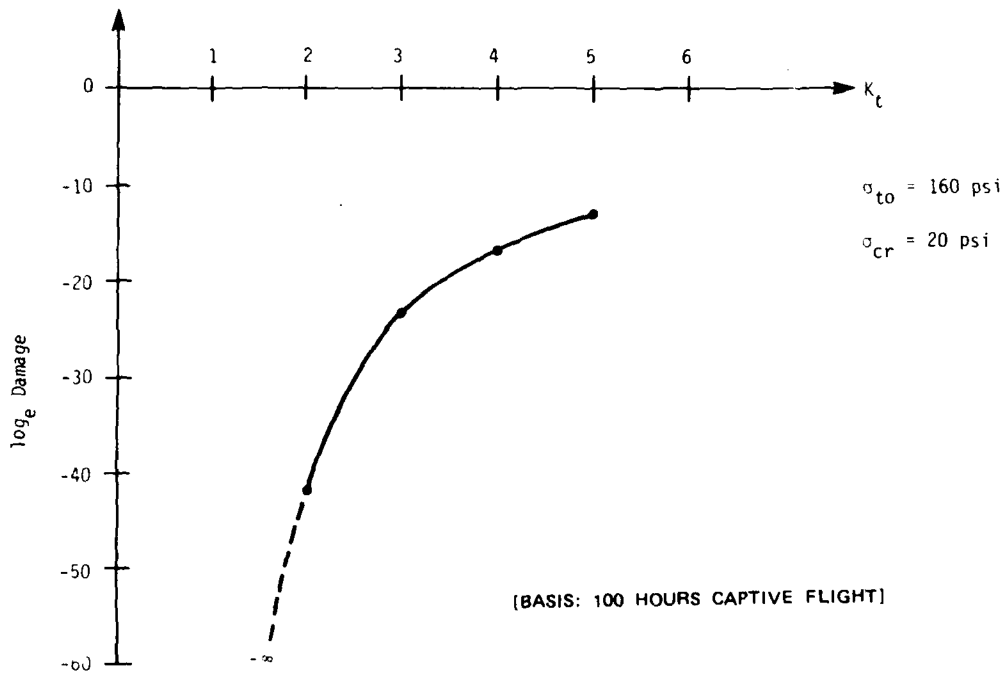


Figure 20 - Damage vs. Stress Concentration

The geometry and properties of the rocket motor were assumed to be:

Case outer radius	2.5 inches	
Case thickness	0.068 inch	
Radius of bore	0.9 inch	
Thermal diffusivity of propellant	1.1 in^2/hr	
Poisson's ratio of propellant	0.499	
Poisson's ratio of case	0.25	
Coefficient of thermal expansion	case	$6.0 \times 10^{-6}/^{0}F$
	propellant	$5.4 \times 10^{-5}/^{0}F$
Young's Modulus	case	30×10^{6} psi
	Propellant	300 psi
B	9.3 (using Bills' nomenclature)[1]	

For captive flight:

Outer temperature of rocket (assumed to be a Gaussian distribution)	T_{mean}	$30^{0}F$
	Standard deviation	$10^{0}F$

Time-temperature shift factor table:

°F	Log a_T
-60	5.59
-40	4.46
-20	3.47
0	2.59
20	1.81
40	1.11
60	0.48
80	-0.08
100	-0.59
120	-1.06
140	-1.48

An example of a hypothetical batch of 500 rockets was analyzed to probabilistically predict the time to failure of rockets in the batch. The properties of the rockets, the dimensions, temperature data and motor logistics (i.e., probability Markov matrix P_{ij}) used in the computer simulation were realistic assumed quantities. For example, normal distributions for annual and diurnal temperature amplitudes were assumed.

A computer program was developed, implementing the techniques discussed in earlier sections, to predict (probabilistically) the time to failure of rocket motors in a given batch. The procedure used in the computer simulation was discussed briefly in the section on Markov state models. Each rocket in the batch is allowed to move from one state to another according to the probability transition matrix P_{ij}, and during this process the damage is accumulated for that particular rocket. When the total damage for a rocket exceeds its tolerance (i.e., the random value of P), the next rocket is considered. This procedure is repeated until all rockets are analyzed, recording their time to failure. Seven different environments are considered with their corresponding thermal environment. These states are typical of (1) curing state, (2) moderate storage, (3) arctic storage, (4) shipboard moderate, (5) truck moderate, (6) train moderate, and (7) aircraft. Many rockets (shipment lots) experience the same environment when travelling from one Markov to another. Therefore, it is not necessary to analyze all 500 rockets separately. Instead, it is assumed that there will be 25 distinct complete travel paths; i.e., rockets will travel in groups of 20, and every rocket in a given group will be subjected to the same environment. However, each rocket within a group will have its own characteristic critical damage value determined probabilistically.

Each group of rockets (consisting of 20 rockets per group in this example) was allowed to travel for ten years, and the damage was accumulated for this period. Consequently, a relation between damage and time was established for every group of rockets for the period of ten years. Then, a critical damage value P (see Figure 13) was randomly selected for every rocket in a group, and from the damage versus time relation the failure time was predicted. Whenever the randomly chosen critical damage (damage capability) exceeded the maximum damage accumulated at ten years, failure time was determined by extrapolating the damage versus time curve. The purpose of analyzing each rocket up to only a limited number of years (ten years in this case) is to reduce the computational expense, because some rockets may probabilistically have very large critical damage values and hence will require an unusually long time to reach failure. In some cases, failure time could exceed 100 years. Such approximations are acceptable since the emphasis of the analysis is to predict early failures rather than those few cases where failure time is very long.

A sensitivity study was done to provide guidance on the relative importance of various temperature parameters affecting rocket motor life. Each (Markov) state in the environmental model is characterized by various temperature parameters such as the mean annual temperature, the seasonal temperature amplitude and standard deviations of amplitudes. Each parameter may have different effects on the rocket failure times. For example, increasing the standard deviations of temperature amplitudes in every state by 20% could cause more damage per unit time than increasing all mean annual temperatures by the same amount. The influence of each temperature parameter was analyzed by varying the basic data. Results are summarized in Figure 21, which shows the percentage of total rockets failed as a function of time.

The results shown in Figure 21 indicate that an increase in the amplitude of the temperature cycle increases the failure rate or, equivalently, shortens the rocket life. Increasing the standard

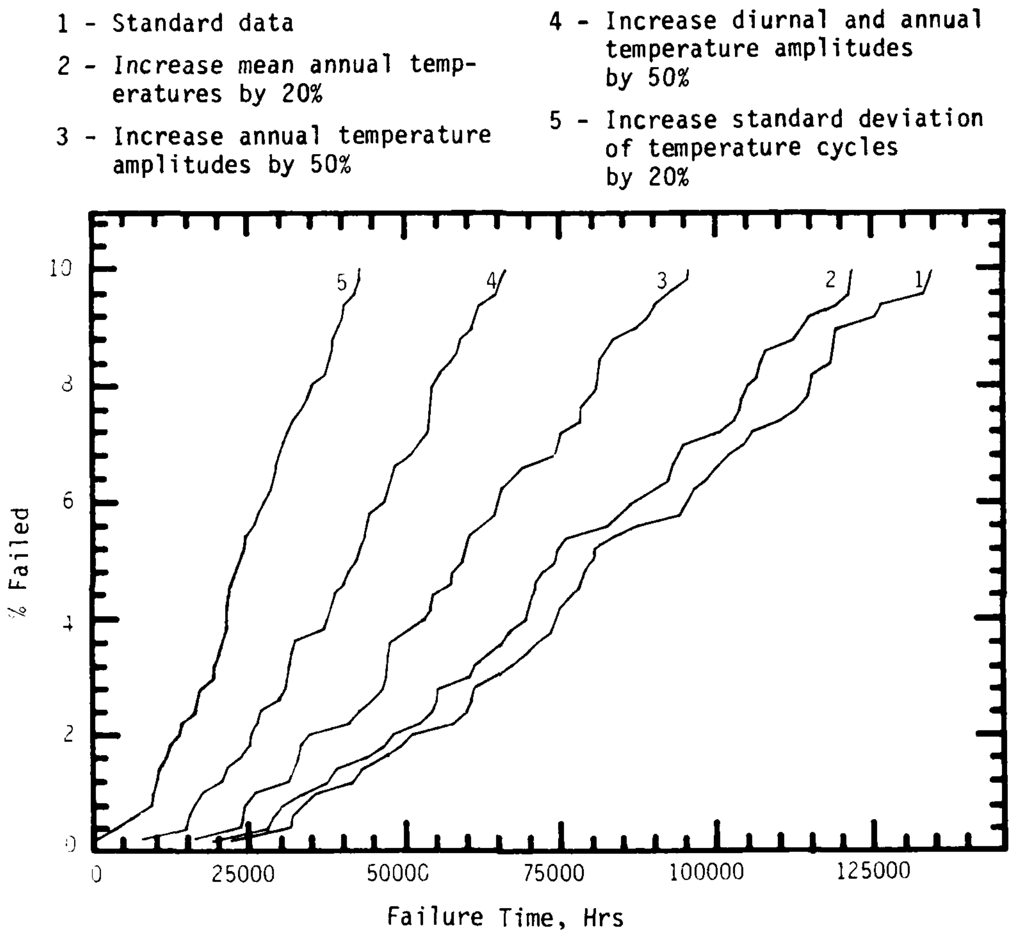


Figure 21 - Percentage of Rockets (Hypothetical Geometry) Failed As a Function of Time

deviations of temperature cycles was more damaging to the rockets than a similar (percentage) increase in the deterministic temperature amplitudes. A Gaussian distribution was assumed for the temperature amplitudes.

The most expensive part of such a computation is the thermal stress analysis. For example, to analyze 25 rockets for 10 years would cost approximately $100 on the Stanford University IBM 3033 computer. As a means of reducing the cost of computation, the damage per unit time for each location was established separately and computed only once, and in the Markov simulation this damage/time was entered as data rather than computed for each hour. Hence, for each different location (storage, captive flight, etc.), the damage for a specified period is computed by describing the probabilistic external temperature model representative for each location.* In the next subsection, this will be demonstrated for Sidewinder rockets.

EXAMPLE OF SIDEWINDER ROCKETS

Damage in Storage Locations

The damage per unit time was computed for each different Markov state. A computer program called STORAGE was developed to estimate (probabilistically) the damage per unit time for each different storage location. The program uses the random external temperature ($T = T_m + f_a T_a \sin \omega_a t + f_d T_d \sin \omega_d t$) model described in detail in the section titled "Sidewinder Storage Location External Tempera-

---

* A more rigorous stochastic method would use a distribution of damage per unit time for each location rather than a deterministic value.

ture." The closed form thermal stress solution described in the section entitled "Example Temperature Distributions in a Propellant" is used to compute maximum hoop stress at the rocket bore. The amplitudes T_d and T_a are randomly selected once every 24 hours for T_d and once every 30 days for T_a. Stresses are calculated, at hourly intervals, based on the random values of the amplitudes. Subsequently, the damage is computed using the cumulative damage formula. The total damage was calculated for each storage location for a ten year period. The geometry and materials properties of the Sidewinder were assumed (for this analysis) to be:

Outside radius	2.5 inches	
Case thickness	0.06 inch	
Radius of bore	0.9 inch	
Thermal diffusivity of propellant	1.1 in^2/hr	
Poisson's ratio of propellant	0.499	
Poisson's ratio of case	0.3	
Coefficient of thermal expansion	case	$6.0 \times 10^{-6}/^{0}F$
	propellant	$5.4 \times 10^{-5}/^{0}F$
Young's modulus	case	30×10^{6} psi
	propellant	450 psi
B (using Bills' nomenclature)	9.3	
σ_{to}		160 psi
σ_{cr}		8 psi
K_t		2

The time temperature shift factors are shown on page 49.

The resulting damage (for ten years) is shown in Table 3 for each location. Most of Navy's storage locations are in fairly mild climates; hence, the damage is relatively low. Generally, colder

Table 3 - Relative Damage

1.	Fallbrook, CA	DAMAGE = 0.40651D-09
2.	Roosevelt Roads, PR	DAMAGE = 0.0*
3.	Key West, FL	DAMAGE = 0.0
4.	Rota, Spain	DAMAGE = 0.53422D-09
5.	Norfolk Reg. VA	DAMAGE = 0.29005D-06
6.	Cherry Point, NC	DAMAGE = 0.21729D-07
7.	Oceana, VA	DAMAGE = 0.18543D-06
8.	Kaneohe Bay, HI	DAMAGE = 0.0
9.	Seal Beach, CA	DAMAGE = 0.45947D-09
10.	Guam/Agana	DAMAGE = 0.0
11.	Iwakuni, Japan	DAMAGE = 0.11127D-06
12.	Las Vegas Nellis, NV	DAMAGE = 0.17629D-06
13.	Dallas, TX	DAMAGE = 0.30505D-07
14.	Yuma, AZ	DAMAGE = 0.43977D-09
15.	Sigonella, Sicily	DAMAGE = 0.17995D-07
16.	Beaufort, SC	DAMAGE = 0.38339D-08
17.	Okinawa Is/Naha	DAMAGE = 0.16054D-16
18.	El Toro, CA	DAMAGE = 0.66235D-09
19.	Miramar, CA	DAMAGE = 0.47866D-09
20.	Yorktown, VA	DAMAGE = 0.20989D-06
21.	Point Mugu, CA	DAMAGE = 0.26757D-09
22.	Kadena/Okinawa	DAMAGE = 0.54705D-16
23.	Da Nang, Vietnam	DAMAGE = 0.0
24.	Nam Phong, Thailand	DAMAGE = 0.0
25.	Concord, CA	DAMAGE = 0.29840D-07
26.	Tokyo/Atsugi, Japan	DAMAGE = 0.33713D-06

*0.0 indicates a very small number, not "zero."

climates would cause more damage due to temperature alone. In Sidewinder's case the most damaging location was located in Tokyo/Japan; but, even there, the damage seems to be minimal. Most storage locations are well insulated (see Figures 22 and 23); therefore the rocket (skin) external temperature will be milder than the ambient temperature. This is considered by using a scaling factor for the amplitudes. In this case, f_a was chosen to be 0.667; and f_d = 0.4.

Figure 22 - Magazine 21HT4, Typical of the AT, BT, BTX, FT, HT, and WT Magazines of the Naval Ammunition Depot, Oahu, Hawaii.

Figure 23 - Magazine 10FT3, Typical of AT, BT, BTX, FT, HT, and WT Magazines at the Naval Ammunition Depot, Oahu, Hawaii

There may be locations in the Sidewinder motor where the combined overall stress and stress concentration factor K_t result in higher local stresses than those obtained from bore stresses with $K_t = 2$. In such locations, the damage might be dramatically increased (see Figure 20).

Damage During Captive Flight

Unlike storage environment, captive flight can be very severe for Sidewinders. The rocket is generally exposed to ambient temperatures, and during high altitude flights, temperatures can be very low; and, hence, cause large damage. Fortunately, most flights are flown at low to medium altitude (below 25,000 ft) where the temperatures are generally above 30°F. In the few instances of slow speed, 45,000 ft altitude flight, the case temperatures can be lower than -40°F.

A computer program called CAPTIVE FLIGHT was written to calculate the damage during captive flight. The program assumes a Gaussian form of external temperature distribution. The mean and the standard deviations are input as data. A constant temperature is maintained during each flight, but the temperature is randomly selected for each flight from the specified normal temperature distribution. All captive flights regardless of their origin are assumed to be subjected to the same external temperature distributions. That is, the distribution is assumed to be representative of all captive flights.* The revised input parameters (changes to the data above) are:

For Captive Flight:

Outer Temperature of Rocket (assuming a Gaussian distribution) $\begin{cases} T_{mean} & = 30°F \\ \text{One standard deviation } \sigma & = 20°F \end{cases}$

Young's Modulus = 800 psi

*Gathering more data is recommended.

The damage was computed for 1000 hours (applied in two-hour blocks, since average flight is two hours in duration, each with randomly selected T) to be:

Cumulative damage during 1000 hours of captive flight $\quad 0.150 \times 10^{-2}$

Note that captive flight is far more damaging than storage (compare to maximum storage damage of 0.337×10^{-6} in ten years). In spite of the relatively little time spent in captive flight as opposed to storage, the captive flight resulted in more damage. Specifically, the FLTAC data shows that on the average, a rocket spends 0.0272% of its life (23.8 hours in 10 years) in captive flight. In spite of this relatively short portion of life that is spent in captive flight, (on the average) captive flight is responsible for **106 times** more damage than storage entirely in the most severe storage location, i.e.:

$$\frac{0.150 \times 10^{-2}}{1000} \times \frac{365 \times 24 \times 10}{0.337 \times 10^{-6}} \times 2.72 \times 10^{-4} = 106$$

$$\left(\frac{\text{captive-flight}}{\text{damage per hour}}\right) \times \left(\frac{1}{\text{damage per hour storage}}\right) \times \left(\frac{\text{captive hours}}{\text{storage hours}}\right) = \left(\frac{\text{captive flight damage}}{\text{storage damage}}\right)$$

Sidewinder Damage During Ship Carry/Stowage

The temperature model described in the section, "External Temperature for Ship Transport," ($T_{ship} = T_m + T_a \sin \omega_a t + T_d \sin \omega_d t$), was used in a program similar to STORAGE to compute the damage per unit time. The distribution for the diurnal amplitude T_d is shown in

Figure 5. The mean temperature T_m equals 74°F and a deterministic value of 4°F was used for the seasonal temperature amplitude. The damage was computed to be 0.115×10^{-16} in two years. This is relatively a very small amount of damage.

Sidewinder Damage During Ground Transportation

During ground transportation (train or truck), the rockets are exposed more directly to the ambient temperature; therefore the insulation provided is minimal. This implies that the scale factors f_a and f_d used in the equation $T = T_m + f_a T_a \sin \omega_a t + f_d T_d \sin \omega_d t$ are close to unity. In this analysis f_a and f_d were taken to be unity, and a damage at each storage location area was computed. The damage that occurs when a rocket moves from location A to location B which required T_o hours of transportation is estimated by using the sum of the damage that occurs in $\frac{T_o}{2}$ hours at location A, plus the damage that occurs at location B for the same period $\frac{T_o}{2}$. The relevant temperatures are those of the arrival and departure locations and the time is divided equally.

The damage per unit time at each storage area (using $f_a = f_d = 1$) is listed in Table 4. Appendix D lists all possible transportation between locations and distances between these locations. The duration of the transportation then is simply the distance divided by the speed of transportation.

Table 4. Cumulative Damage in Five* Years ($f_a = f_d = 1$)

1.	Fallbrook, CA	DAMAGE = 0.50457D-08
2.	Roosevelt Roads, PR	DAMAGE = 0.0
3.	Key West, FL	DAMAGE = 0.10093D-15
4.	Rota, Spain	DAMAGE = 0.47872D-08
5.	Norfolk Reg., VA	DAMAGE = 0.10161D-04
6.	Cherry Point, NC	DAMAGE = 0.37204D-05
7.	Oceana, VA	DAMAGE = 0.64862D-05
8.	Kaneohe Bay, HI	DAMAGE = 0.0
9.	Seal Beach, CA	DAMAGE = 0.42584D-08
10.	Guam/Agana	DAMAGE = 0.0
11.	Iwakuni, Japan	DAMAGE = 0.44686D-05
12.	Las Vegas Nellis, NV	DAMAGE = 0.35502D-04
13.	Dallas, TX	DAMAGE = 0.72915D-05
14.	Yuma, AZ	DAMAGE = 0.21993D-05
15.	Sigonella, Sicily	DAMAGE = 0.29079D-05
16.	Beaufort, SC	DAMAGE = 0.24641D-05
17.	Okinawa Is./Naha	DAMAGE = 0.35618D-10
18.	El Toro, CA	DAMAGE = 0.46481D-06
19.	Miramar, CA	DAMAGE = 0.25572D-06
20.	Yorktown, VA	DAMAGE = 0.98086D-05
21.	Point Mugu, CA	DAMAGE = 0.14555D-06
22.	Kadena/Okinawa	DAMAGE = 0.53948D-09
23.	Da Nang, Vietnam	DAMAGE = 0.46830D-12
24.	Nam Phong, Thailand	DAMAGE = 0.44726D-19
25.	Concord, CA	DAMAGE = 0.58973D-05
26.	Tokyo/Atsugi, Japan	DAMAGE = 0.14412D-04

*Note this relatively short time compared to actual experience.

Sidewinder Damage During Air Transportation

During air transportation the temperatures are fairly mild. Assuming a mean temperature of 60°F and a standard deviation of 12°F, the damage was computed to be 0.885×10^{-5} in 1000 hours.

CAPTIVE FLIGHT computer code was used to determine the air transportation damage. This damage is 170 times smaller than captive flight damage.

MARKOV SIMULATION OF SIDEWINDER ROCKETS

The values of damage per unit time from each specific type of location (e.g., captive flight, storage, ship carry) computed earlier were used as input in a program called MARKOV to determine probabilistically the damage that occurs in a rocket fleet. An example of a fleet with 1000 rockets was analyzed. MARKOV uses as input the probability transition (MARKOV) matrix of the rocket in consideration. In this example, the Sidewinder transition matrix derived from FLTAC data was used. The characteristic "damages per unit time" are next read as data input. The program then simulates the rocket history and the damage accumulation. For Sidewinder rockets, as expected, those rockets that were extensively used in captive flight experienced the most severe damage (Tables 5 and 6 show the result of the MARKOV computer run). Magnitudes of damage, however, seem low. The maximum damage was computed to be 1.182×10^{-3} after 10 years of simulated rocket use. Figure 24 shows the cumulative density function of damage distribution, and Figure 25 shows the percentage of rockets with extreme damage. Because of the tremendous sensitivity of the damage value to the characteristic input parameters (see section titled "Parametric Sensitivity Analyses"), the values of damage should be interpreted in a relative sense, and a calibration of the damage formula as was discussed by Bills[1] may be necessary. Cyclic thermal loading experiments of propellants will be of great utility.

Table 5 - Sidewinder Rocket Damage.

IR=	1	DMG=	0.19017E-04	TIME=	93502.	NLOC=	5	NCAP=	7
IR=	2	DMG=	0.13748E-04	TIME=	94926.	NLOC=	5	NCAP=	5
IR=	3	DMG=	0.28519E-05	TIME=	106167.	NLOC=	5	NCAP=	1
IR=	4	DMG=	0.54417E-04	TIME=	96721.	NLOC=	6	NCAP=	20
IR=	5	DMG=	0.45927E-05	TIME=	104421.	NLOC=	7	NCAP=	1
IR=	6	DMG=	0.55930E-05	TIME=	91925.	NLOC=	4	NCAP=	2
IR=	7	DMG=	0.84219E-04	TIME=	108796.	NLOC=	3	NCAP=	31
IR=	8	DMG=	0.70805E-04	TIME=	110049.	NLOC=	5	NCAP=	26
IR=	9	DMG=	0.16313E-04	TIME=	102513.	NLOC=	6	NCAP=	6
IR=	10	DMG=	0.24533E-04	TIME=	106352.	NLOC=	4	NCAP=	9
IR=	11	DMG=	0.62567E-04	TIME=	102986.	NLOC=	5	NCAP=	23
IR=	12	DMG=	0.13584E-04	TIME=	136881.	NLOC=	3	NCAP=	5
IR=	13	DMG=	0.57077E-04	TIME=	96114.	NLOC=	7	NCAP=	21
IR=	14	DMG=	0.16457E-06	TIME=	99236.	NLOC=	6	NCAP=	0
IR=	15	DMG=	0.75440E-05	TIME=	120546.	NLOC=	5	NCAP=	2
IR=	16	DMG=	0.23762E-04	TIME=	95116.	NLOC=	4	NCAP=	8
IR=	17	DMG=	0.82325E-05	TIME=	103598.	NLOC=	5	NCAP=	3
IR=	18	DMG=	0.24539E-04	TIME=	92538.	NLOC=	5	NCAP=	9
IR=	19	DMG=	0.27992E-05	TIME=	117600.	NLOC=	3	NCAP=	1
IR=	20	DMG=	0.0	TIME=	90240.	NLOC=	2	NCAP=	0
IR=	21	DMG=	0.11962E-03	TIME=	150732.	NLOC=	2	NCAP=	44
IR=	22	DMG=	0.56802E-05	TIME=	120505.	NLOC=	5	NCAP=	2
IR=	23	DMG=	0.15408E-03	TIME=	107379.	NLOC=	5	NCAP=	56
IR=	24	DMG=	0.11818E-02	TIME=	151439.	NLOC=	2	NCAP=	435
IR=	25	DMG=	0.81584E-04	TIME=	143354.	NLOC=	5	NCAP=	30
IR=	26	DMG=	0.12895E-04	TIME=	114516.	NLOC=	5	NCAP=	4
IR=	27	DMG=	0.24451E-04	TIME=	93564.	NLOC=	4	NCAP=	9
IR=	28	DMG=	0.61160E-08	TIME=	104047.	NLOC=	4	NCAP=	0
IR=	29	DMG=	0.11031E-04	TIME=	95068.	NLOC=	4	NCAP=	4
IR=	30	DMG=	0.83970E-05	TIME=	117845.	NLOC=	5	NCAP=	3
IR=	31	DMG=	0.55990E-05	TIME=	89310.	NLOC=	4	NCAP=	2
IR=	32	DMG=	0.87018E-04	TIME=	92798.	NLOC=	5	NCAP=	32
IR=	33	DMG=	0.67155E-10	TIME=	102947.	NLOC=	2	NCAP=	0
IR=	34	DMG=	0.13554E-03	TIME=	129137.	NLOC=	2	NCAP=	50
IR=	35	DMG=	0.43632E-04	TIME=	109030.	NLOC=	5	NCAP=	16
IR=	36	DMG=	0.15447E-04	TIME=	101754.	NLOC=	5	NCAP=	5
IR=	37	DMG=	0.82327E-05	TIME=	92907.	NLOC=	5	NCAP=	3
IR=	38	DMG=	0.13592E-03	TIME=	94797.	NLOC=	5	NCAP=	50
IR=	39	DMG=	0.27250E-04	TIME=	117617.	NLOC=	3	NCAP=	10
IR=	40	DMG=	0.73353E-04	TIME=	133966.	NLOC=	7	NCAP=	27
IR=	41	DMG=	0.13748E-04	TIME=	91524.	NLOC=	5	NCAP=	5
IR=	42	DMG=	0.21628E-04	TIME=	104733.	NLOC=	5	NCAP=	8
IR=	43	DMG=	0.65366E-04	TIME=	110101.	NLOC=	5	NCAP=	24
IR=	44	DMG=	0.16455E-04	TIME=	113796.	NLOC=	4	NCAP=	6
IR=	45	DMG=	0.27332E-04	TIME=	103949.	NLOC=	4	NCAP=	10
IR=	46	DMG=	0.82252E-07	TIME=	95759.	NLOC=	2	NCAP=	0
IR=	47	DMG=	0.28812E-05	TIME=	94744.	NLOC=	4	NCAP=	1
IR=	48	DMG=	0.82252E-07	TIME=	87871.	NLOC=	3	NCAP=	0
IR=	49	DMG=	0.76074E-04	TIME=	89856.	NLOC=	4	NCAP=	28
IR=	50	DMG=	0.0	TIME=	88493.	NLOC=	1	NCAP=	0

IR = individual rocket (which can also be considered to be an average rocket of a group used as a shipment lot)
DMG = damage accumulated for time shown
TIME = total hours
NLOC = number of locations
NCAP = number of captive flights

Table 5 (Contd.)

IR=	51	DMG=	0.11038E-04	TIME=	110823.	NLOC=	7	NCAP=	
IR=	52	DMG=	0.83147E-05	TIME=	114619.	NLOC=	6	NCAP=	3
IR=	53	DMG=	0.67155E-10	TIME=	98971.	NLOC=	4	NCAP=	0
IR=	54	DMG=	0.15530E-04	TIME=	101160.	NLOC=	3	NCAP=	5
IR=	55	DMG=	0.22285E-03	TIME=	104060.	NLOC=	4	NCAP=	82
IR=	56	DMG=	0.20275E-04	TIME=	120594.	NLOC=	5	NCAP=	6
IR=	57	DMG=	0.23762E-04	TIME=	100875.	NLOC=	4	NCAP=	8
IR=	58	DMG=	0.55160E-05	TIME=	88904.	NLOC=	3	NCAP=	2
IR=	59	DMG=	0.73434E-04	TIME=	93679.	NLOC=	3	NCAP=	27
IR=	60	DMG=	0.84307E-04	TIME=	103706.	NLOC=	5	NCAP=	31
IR=	61	DMG=	0.12709E-03	TIME=	118135.	NLOC=	5	NCAP=	46
IR=	62	DMG=	0.10893E-03	TIME=	103538.	NLOC=	4	NCAP=	40
IR=	63	DMG=	0.48086E-03	TIME=	116662.	NLOC=	1	NCAP=	177
IR=	64	DMG=	0.48638E-03	TIME=	186402.	NLOC=	4	NCAP=	179
IR=	65	DMG=	0.21734E-03	TIME=	93566.	NLOC=	6	NCAP=	80
IR=	66	DMG=	0.38199E-04	TIME=	93828.	NLOC=	5	NCAP=	14
IR=	67	DMG=	0.16450E-06	TIME=	103935.	NLOC=	4	NCAP=	0
IR=	68	DMG=	0.46627E-05	TIME=	93538.	NLOC=	5	NCAP=	1
IR=	69	DMG=	0.11031E-04	TIME=	90118.	NLOC=	4	NCAP=	4
IR=	70	DMG=	0.82325E-05	TIME=	92653.	NLOC=	4	NCAP=	3
IR=	71	DMG=	0.19023E-04	TIME=	115535.	NLOC=	4	NCAP=	7
IR=	72	DMG=	0.65312E-04	TIME=	119414.	NLOC=	7	NCAP=	24
IR=	73	DMG=	0.11410E-03	TIME=	93493.	NLOC=	4	NCAP=	42
IR=	74	DMG=	0.56040E-05	TIME=	107233.	NLOC=	6	NCAP=	2
IR=	75	DMG=	0.35482E-04	TIME=	100843.	NLOC=	4	NCAP=	13
IR=	76	DMG=	0.45414E-04	TIME=	95407.	NLOC=	4	NCAP=	16
IR=	77	DMG=	0.12731E-04	TIME=	120936.	NLOC=	4	NCAP=	4
IR=	78	DMG=	0.19029E-04	TIME=	88672.	NLOC=	4	NCAP=	7
IR=	79	DMG=	0.16450E-06	TIME=	147724.	NLOC=	4	NCAP=	0
IR=	80	DMG=	0.43632E-04	TIME=	88705.	NLOC=	3	NCAP=	16
IR=	81	DMG=	0.16029E-03	TIME=	183896.	NLOC=	6	NCAP=	59
IR=	82	DMG=	0.13748E-04	TIME=	124478.	NLOC=	4	NCAP=	5
IR=	83	DMG=	0.19182E-04	TIME=	98994.	NLOC=	4	NCAP=	7
IR=	84	DMG=	0.55157E-05	TIME=	115026.	NLOC=	4	NCAP=	2
IR=	85	DMG=	0.21199E-03	TIME=	107129.	NLOC=	5	NCAP=	78
IR=	86	DMG=	0.45420E-04	TIME=	92396.	NLOC=	5	NCAP=	16
IR=	87	DMG=	0.32765E-04	TIME=	100842.	NLOC=	4	NCAP=	12
IR=	88	DMG=	0.11032E-04	TIME=	124795.	NLOC=	4	NCAP=	4
IR=	89	DMG=	0.42652E-03	TIME=	175696.	NLOC=	5	NCAP=	157
IR=	90	DMG=	0.30049E-04	TIME=	94284.	NLOC=	4	NCAP=	11
IR=	91	DMG=	0.32683E-04	TIME=	98891.	NLOC=	3	NCAP=	12
IR=	92	DMG=	0.28812E-05	TIME=	109692.	NLOC=	4	NCAP=	1
IR=	93	DMG=	0.32683E-04	TIME=	114763.	NLOC=	6	NCAP=	12
IR=	94	DMG=	0.11031E-04	TIME=	103938.	NLOC=	4	NCAP=	4
IR=	95	DMG=	0.81562E-05	TIME=	100130.	NLOC=	6	NCAP=	3
IR=	96	DMG=	0.54499E-04	TIME=	106367.	NLOC=	4	NCAP=	20
IR=	97	DMG=	0.49066E-04	TIME=	92068.	NLOC=	3	NCAP=	18
IR=	98	DMG=	0.51792E-04	TIME=	109610.	NLOC=	5	NCAP=	19
IR=	99	DMG=	0.51789E-04	TIME=	98597.	NLOC=	5	NCAP=	19
IR=	100	DMG=	0.21616E-04	TIME=	144394.	NLOC=	4	NCAP=	8
IR=	101	DMG=	0.19182E-04	TIME=	93092.	NLOC=	5	NCAP=	7
IR=	102	DMG=	0.95250E-04	TIME=	94340.	NLOC=	4	NCAP=	35
IR=	103	DMG=	0.57216E-04	TIME=	99019.	NLOC=	4	NCAP=	21
IR=	104	DMG=	0.24616E-04	TIME=	105662.	NLOC=	4	NCAP=	9
IR=	105	DMG=	0.13749E-04	TIME=	100510.	NLOC=	4	NCAP=	5
IR=	106	DMG=	0.46267E-04	TIME=	110316.	NLOC=	4	NCAP=	17
IR=	107	DMG=	0.10949E-04	TIME=	93639.	NLOC=	3	NCAP=	4
IR=	108	DMG=	0.57140E-04	TIME=	96944.	NLOC=	5	NCAP=	21
IR=	109	DMG=	0.55980E-05	TIME=	94746.	NLOC=	4	NCAP=	2
IR=	110	DMG=	0.10178E-04	TIME=	112766.	NLOC=	5	NCAP=	3

Table 5 (Contd.)

IR=	111	DMG=	0.82385E-05	TIME=	105658.	NLOC=	4	NCAP=	[illegible]
IR=	112	DMG=	0.57134E-04	TIME=	125612.	NLOC=	4	NCAP=	21
IR=	113	DMG=	0.13830E-04	TIME=	118934.	NLOC=	5	NCAP=	5
IR=	114	DMG=	0.25461E-04	TIME=	102831.	NLOC=	5	NCAP=	8
IR=	115	DMG=	0.12895E-04	TIME=	88710.	NLOC=	5	NCAP=	4
IR=	116	DMG=	0.28260E-04	TIME=	89591.	NLOC=	5	NCAP=	9
IR=	117	DMG=	0.13856E-03	TIME=	101314.	NLOC=	5	NCAP=	51
IR=	118	DMG=	0.67918E-04	TIME=	94234.	NLOC=	5	NCAP=	25
IR=	119	DMG=	0.67966E-04	TIME=	95149.	NLOC=	9	NCAP=	25
IR=	120	DMG=	0.13666E-04	TIME=	87880.	NLOC=	3	NCAP=	5
IR=	121	DMG=	0.16450E-06	TIME=	104915.	NLOC=	4	NCAP=	0
IR=	122	DMG=	0.48993E-04	TIME=	186120.	NLOC=	3	NCAP=	18
IR=	123	DMG=	0.55980E-05	TIME=	93083.	NLOC=	5	NCAP=	2
IR=	124	DMG=	0.19182E-04	TIME=	147737.	NLOC=	4	NCAP=	7
IR=	125	DMG=	0.82319E-07	TIME=	102325.	NLOC=	4	NCAP=	0
IR=	126	DMG=	0.82512E-05	TIME=	94741.	NLOC=	5	NCAP=	3
IR=	127	DMG=	0.12785E-03	TIME=	114811.	NLOC=	6	NCAP=	47
IR=	128	DMG=	0.21899E-04	TIME=	133673.	NLOC=	4	NCAP=	8
IR=	129	DMG=	0.14144E-03	TIME=	191582.	NLOC=	5	NCAP=	52
IR=	130	DMG=	0.16300E-04	TIME=	89889.	NLOC=	3	NCAP=	6
IR=	131	DMG=	0.21734E-04	TIME=	88069.	NLOC=	3	NCAP=	8
IR=	132	DMG=	0.82252E-07	TIME=	122804.	NLOC=	2	NCAP=	0
IR=	133	DMG=	0.32683E-04	TIME=	102117.	NLOC=	4	NCAP=	12
IR=	134	DMG=	0.17393E-04	TIME=	111448.	NLOC=	6	NCAP=	5
IR=	135	DMG=	0.15230E-03	TIME=	109792.	NLOC=	4	NCAP=	56
IR=	136	DMG=	0.12819E-04	TIME=	89409.	NLOC=	5	NCAP=	4
IR=	137	DMG=	0.28054E-05	TIME=	120565.	NLOC=	5	NCAP=	1
IR=	138	DMG=	0.55157E-05	TIME=	88775.	NLOC=	3	NCAP=	2
IR=	139	DMG=	0.19099E-04	TIME=	111647.	NLOC=	5	NCAP=	7
IR=	140	DMG=	0.14128E-03	TIME=	114668.	NLOC=	7	NCAP=	52
IR=	141	DMG=	0.95086E-04	TIME=	116405.	NLOC=	1	NCAP=	35
IR=	142	DMG=	0.12956E-03	TIME=	107727.	NLOC=	6	NCAP=	47
IR=	143	DMG=	0.24451E-04	TIME=	89694.	NLOC=	3	NCAP=	9
IR=	144	DMG=	0.11114E-04	TIME=	120190.	NLOC=	5	NCAP=	4
IR=	145	DMG=	0.17055E-06	TIME=	104058.	NLOC=	4	NCAP=	0
IR=	146	DMG=	0.12313E-04	TIME=	115024.	NLOC=	5	NCAP=	4
IR=	147	DMG=	0.83[illegible]47E-05	TIME=	95182.	NLOC=	4	NCAP=	3
IR=	148	DMG=	[illegible]3E-04	TIME=	94239.	NLOC=	5	NCAP=	4
IR=	149	DMG=	0.29566E-04	TIME=	146351.	NLOC=	5	NCAP=	11
IR=	150	DMG=	0.645[illegible]3E-04	TIME=	103450.	NLOC=	5	NCAP=	23
IR=	151	DMG=	0.27228E-05	TIME=	91063.	NLOC=	4	NCAP=	1
IR=	152	DMG=	0.35406E-04	TIME=	101401.	NLOC=	5	NCAP=	13
IR=	153	DMG=	0.16465E-04	TIME=	98187.	NLOC=	4	NCAP=	6
IR=	154	DMG=	0.15612E-04	TIME=	94706.	NLOC=	5	NCAP=	5
IR=	155	DMG=	0.47450E-05	TIME=	95103.	NLOC=	4	NCAP=	1
IR=	156	DMG=	0.10324E-03	TIME=	100329.	NLOC=	5	NCAP=	38
IR=	157	DMG=	0.12098E-07	TIME=	116588.	NLOC=	4	NCAP=	0
IR=	158	DMG=	0.47450E-05	TIME=	95104.	NLOC=	4	NCAP=	1
IR=	159	DMG=	0.17124E-03	TIME=	107355.	NLOC=	4	NCAP=	63
IR=	160	DMG=	0.32765E-04	TIME=	95082.	NLOC=	4	NCAP=	12
IR=	161	DMG=	0.55920E-05	TIME=	138539.	NLOC=	4	NCAP=	2
IR=	162	DMG=	0.12513E-03	TIME=	109774.	NLOC=	4	NCAP=	46
IR=	163	DMG=	0.24730E-03	TIME=	121292.	NLOC=	5	NCAP=	91
IR=	164	DMG=	0.70805E-04	TIME=	112426.	NLOC=	5	NCAP=	26
IR=	165	DMG=	0.89658E-04	TIME=	118317.	NLOC=	6	NCAP=	33
IR=	166	DMG=	0.81690E-05	TIME=	100556.	NLOC=	5	NCAP=	3
IR=	167	DMG=	0.12225E-03	TIME=	143582.	NLOC=	3	NCAP=	45
IR=	168	DMG=	0.16450E-06	TIME=	92036.	NLOC=	3	NCAP=	0
IR=	169	DMG=	0.27230E-05	TIME=	101071.	NLOC=	3	NCAP=	1
IR=	170	DMG=	0.16588E-03	TIME=	109798.	NLOC=	4	NCAP=	61

Table 5 (Contd.)

IR=	171	DMG=	0.28812E-05	TIME=	138537.	NLOC=	4	NCAP=	[illegible]
IR=	172	DMG=	0.59080E-04	TIME=	100899.	NLOC=	4	NCAP=	21
IR=	173	DMG=	0.54456E-05	TIME=	92987.	NLOC=	5	NCAP=	2
IR=	174	DMG=	0.10340E-03	TIME=	193814.	NLOC=	4	NCAP=	38
IR=	175	DMG=	0.12901E-04	TIME=	112890.	NLOC=	5	NCAP=	4
IR=	176	DMG=	0.56802E-05	TIME=	119096.	NLOC=	6	NCAP=	2
IR=	177	DMG=	0.95174E-04	TIME=	89576.	NLOC=	5	NCAP=	35
IR=	178	DMG=	0.83147E-05	TIME=	100825.	NLOC=	4	NCAP=	3
IR=	179	DMG=	0.59775E-04	TIME=	134162.	NLOC=	2	NCAP=	22
IR=	180	DMG=	0.11031E-04	TIME=	96138.	NLOC=	3	NCAP=	4
IR=	181	DMG=	0.97804E-04	TIME=	95096.	NLOC=	6	NCAP=	36
IR=	182	DMG=	0.24676E-06	TIME=	118535.	NLOC=	5	NCAP=	0
IR=	183	DMG=	0.24533E-04	TIME=	99780.	NLOC=	6	NCAP=	9
IR=	184	DMG=	0.16860E-03	TIME=	110975.	NLOC=	5	NCAP=	62
IR=	185	DMG=	0.10893E-03	TIME=	100990.	NLOC=	4	NCAP=	40
IR=	186	DMG=	0.16471E-04	TIME=	121260.	NLOC=	5	NCAP=	6
IR=	187	DMG=	0.14594E-04	TIME=	100630.	NLOC=	6	NCAP=	4
IR=	188	DMG=	0.26479E-04	TIME=	118532.	NLOC=	5	NCAP=	9
IR=	189	DMG=	0.55960E-05	TIME=	103342.	NLOC=	4	NCAP=	2
IR=	190	DMG=	0.12098E-07	TIME=	107374.	NLOC=	4	NCAP=	0
IR=	191	DMG=	0.93255E-05	TIME=	95108.	NLOC=	6	NCAP=	2
IR=	192	DMG=	0.13749E-04	TIME=	103940.	NLOC=	4	NCAP=	5
IR=	193	DMG=	0.92325E-05	TIME=	88009.	NLOC=	4	NCAP=	3
IR=	194	DMG=	0.70717E-04	TIME=	93275.	NLOC=	4	NCAP=	26
IR=	195	DMG=	0.94350E-07	TIME=	116239.	NLOC=	6	NCAP=	0
IR=	196	DMG=	0.13748E-04	TIME=	92045.	NLOC=	3	NCAP=	5
IR=	197	DMG=	0.82252E-07	TIME=	122804.	NLOC=	2	NCAP=	0
IR=	198	DMG=	0.21893E-04	TIME=	103946.	NLOC=	4	NCAP=	8
IR=	199	DMG=	0.28050E-05	TIME=	113542.	NLOC=	3	NCAP=	1
IR=	200	DMG=	0.29635E-05	TIME=	118927.	NLOC=	5	NCAP=	1
IR=	201	DMG=	0.82252E-07	TIME=	95759.	NLOC=	2	NCAP=	0
IR=	202	DMG=	0.32765E-04	TIME=	94605.	NLOC=	4	NCAP=	12
IR=	203	DMG=	0.83147E-05	TIME=	95761.	NLOC=	4	NCAP=	3
IR=	204	DMG=	0.11410E-03	TIME=	89578.	NLOC=	3	NCAP=	42
IR=	205	DMG=	0.25721E-09	TIME=	109719.	NLOC=	4	NCAP=	0
IR=	206	DMG=	0.27990E-05	TIME=	95761.	NLOC=	2	NCAP=	1
IR=	207	DMG=	0.11139E-03	TIME=	114575.	NLOC=	5	NCAP=	41
IR=	208	DMG=	0.10121E-04	TIME=	101682.	NLOC=	5	NCAP=	3
IR=	209	DMG=	0.28050E-05	TIME=	88300.	NLOC=	3	NCAP=	1
IR=	210	DMG=	0.83147E-05	TIME=	122183.	NLOC=	4	NCAP=	3
IR=	211	DMG=	0.40763E-04	TIME=	101391.	NLOC=	5	NCAP=	15
IR=	212	DMG=	0.15530E-04	TIME=	101160.	NLOC=	3	NCAP=	5
IR=	213	DMG=	0.28312E-05	TIME=	93426.	NLOC=	4	NCAP=	1
IR=	214	DMG=	0.38116E-04	TIME=	100602.	NLOC=	4	NCAP=	14
IR=	215	DMG=	0.35488E-04	TIME=	104031.	NLOC=	4	NCAP=	13
IR=	216	DMG=	0.11031E-04	TIME=	104604.	NLOC=	4	NCAP=	4
IR=	217	DMG=	0.19460E-05	TIME=	93536.	NLOC=	5	NCAP=	0
IR=	218	DMG=	0.54417E-04	TIME=	90307.	NLOC=	3	NCAP=	20
IR=	219	DMG=	0.19099E-04	TIME=	96350.	NLOC=	4	NCAP=	7
IR=	220	DMG=	0.13748E-04	TIME=	103940.	NLOC=	4	NCAP=	5
IR=	221	DMG=	0.17328E-06	TIME=	104058.	NLOC=	3	NCAP=	0
IR=	222	DMG=	0.21899E-04	TIME=	103468.	NLOC=	4	NCAP=	8
IR=	223	DMG=	0.27992E-05	TIME=	91174.	NLOC=	3	NCAP=	1
IR=	224	DMG=	0.27990E-03	TIME=	97773.	NLOC=	6	NCAP=	103
IR=	225	DMG=	0.55161E-05	TIME=	110292.	NLOC=	3	NCAP=	2
IR=	226	DMG=	0.10332E-03	TIME=	90293.	NLOC=	7	NCAP=	38
IR=	227	DMG=	0.0	TIME=	88493.	NLOC=	1	NCAP=	0
IR=	228	DMG=	0.65366E-04	TIME=	92441.	NLOC=	4	NCAP=	24
IR=	229	DMG=	0.29978E-04	TIME=	90931.	NLOC=	5	NCAP=	11
IR=	230	DMG=	0.12042E-04	TIME=	108141.	NLOC=	5	NCAP=	3

Table 5 (Contd.)

IR=	231	DMG=	0.54396E-05	TIME=	91064.	NLOC=	4	NCAP=	[illegible]
IR=	232	DMG=	0.54398E-05	TIME=	109972.	NLOC=	4	NCAP=	2
IR=	233	DMG=	0.24676E-06	TIME=	120183.	NLOC=	5	NCAP=	0
IR=	234	DMG=	0.55980E-05	TIME=	147584.	NLOC=	5	NCAP=	2
IR=	235	DMG=	0.28812E-05	TIME=	95063.	NLOC=	4	NCAP=	1
IR=	236	DMG=	0.35318E-04	TIME=	113028.	NLOC=	3	NCAP=	13
IR=	237	DMG=	0.11114E-04	TIME=	111892.	NLOC=	5	NCAP=	4
IR=	238	DMG=	0.29975E-04	TIME=	97275.	NLOC=	3	NCAP=	11
IR=	239	DMG=	0.83147E-05	TIME=	138860.	NLOC=	4	NCAP=	3
IR=	240	DMG=	0.76068E-04	TIME=	116413.	NLOC=	6	NCAP=	28
IR=	241	DMG=	0.19100E-04	TIME=	112065.	NLOC=	4	NCAP=	7
IR=	242	DMG=	0.11031E-04	TIME=	103938.	NLOC=	4	NCAP=	4
IR=	243	DMG=	0.13590E-04	TIME=	108174.	NLOC=	6	NCAP=	5
IR=	244	DMG=	0.43469E-04	TIME=	134699.	NLOC=	6	NCAP=	16
IR=	245	DMG=	0.24842E-07	TIME=	95502.	NLOC=	5	NCAP=	0
IR=	246	DMG=	0.11155E-03	TIME=	103064.	NLOC=	4	NCAP=	41
IR=	247	DMG=	0.40751E-03	TIME=	171724.	NLOC=	5	NCAP=	150
IR=	248	DMG=	0.28812E-05	TIME=	139551.	NLOC=	4	NCAP=	1
IR=	249	DMG=	0.24615E-04	TIME=	90069.	NLOC=	5	NCAP=	9
IR=	250	DMG=	0.51782E-04	TIME=	94298.	NLOC=	4	NCAP=	19
IR=	251	DMG=	0.82330E-05	TIME=	92732.	NLOC=	4	NCAP=	3
IR=	252	DMG=	0.13600E-03	TIME=	121201.	NLOC=	5	NCAP=	50
IR=	253	DMG=	0.86947E-04	TIME=	94226.	NLOC=	5	NCAP=	32
IR=	254	DMG=	0.59850E-04	TIME=	97070.	NLOC=	5	NCAP=	22
IR=	255	DMG=	0.57057E-04	TIME=	87945.	NLOC=	5	NCAP=	21
IR=	256	DMG=	0.74617E-05	TIME=	95105.	NLOC=	4	NCAP=	2
IR=	257	DMG=	0.57216E-04	TIME=	95099.	NLOC=	4	NCAP=	21
IR=	258	DMG=	0.32765E-04	TIME=	100841.	NLOC=	4	NCAP=	12
IR=	259	DMG=	0.10883E-03	TIME=	99052.	NLOC=	4	NCAP=	40
IR=	260	DMG=	0.21734E-04	TIME=	93431.	NLOC=	4	NCAP=	8
IR=	261	DMG=	0.79335E-03	TIME=	168357.	NLOC=	3	NCAP=	292
IR=	262	DMG=	0.92533E-04	TIME=	92097.	NLOC=	3	NCAP=	34
IR=	263	DMG=	0.95250E-04	TIME=	92099.	NLOC=	3	NCAP=	35
IR=	264	DMG=	0.28815E-05	TIME=	124790.	NLOC=	4	NCAP=	1
IR=	265	DMG=	0.24615E-04	TIME=	94599.	NLOC=	4	NCAP=	9
IR=	266	DMG=	0.95250E-04	TIME=	103994.	NLOC=	4	NCAP=	35
IR=	267	DMG=	0.83147E-05	TIME=	95761.	NLOC=	4	NCAP=	3
IR=	268	DMG=	0.73982E-05	TIME=	115159.	NLOC=	6	NCAP=	2
IR=	269	DMG=	0.11682E-03	TIME=	137308.	NLOC=	4	NCAP=	43
IR=	270	DMG=	0.82328E-05	TIME=	94322.	NLOC=	3	NCAP=	3
IR=	271	DMG=	0.55980E-05	TIME=	103457.	NLOC=	4	NCAP=	2
IR=	272	DMG=	0.89658E-04	TIME=	98229.	NLOC=	6	NCAP=	33
IR=	273	DMG=	0.54456E-05	TIME=	97087.	NLOC=	5	NCAP=	2
IR=	274	DMG=	0.24451E-04	TIME=	98509.	NLOC=	4	NCAP=	9
IR=	275	DMG=	0.27332E-04	TIME=	93098.	NLOC=	5	NCAP=	10
IR=	276	DMG=	0.62509E-04	TIME=	88157.	NLOC=	6	NCAP=	23
IR=	277	DMG=	0.29635E-05	TIME=	114191.	NLOC=	5	NCAP=	1
IR=	278	DMG=	0.73434E-04	TIME=	103914.	NLOC=	6	NCAP=	27
IR=	279	DMG=	0.82252E-07	TIME=	95759.	NLOC=	2	NCAP=	0
IR=	280	DMG=	0.54417E-04	TIME=	119437.	NLOC=	4	NCAP=	20
IR=	281	DMG=	0.46267E-04	TIME=	101496.	NLOC=	6	NCAP=	17
IR=	282	DMG=	0.20963E-04	TIME=	107883.	NLOC=	4	NCAP=	7
IR=	283	DMG=	0.13748E-04	TIME=	95070.	NLOC=	4	NCAP=	5
IR=	284	DMG=	0.27228E-05	TIME=	94824.	NLOC=	5	NCAP=	1
IR=	285	DMG=	0.65372E-04	TIME=	99150.	NLOC=	4	NCAP=	24
IR=	286	DMG=	0.11114E-04	TIME=	120509.	NLOC=	5	NCAP=	4
IR=	287	DMG=	0.10332E-03	TIME=	91775.	NLOC=	4	NCAP=	38
IR=	288	DMG=	0.68000E-04	TIME=	118945.	NLOC=	5	NCAP=	25
IR=	289	DMG=	0.21898E-04	TIME=	100268.	NLOC=	4	NCAP=	8
IR=	290	DMG=	0.55157E-05	TIME=	95762.	NLOC=	2	NCAP=	2

Table 5 (Contd.)

IR=	291	DMG=	0.82252E-07	TIME=	95759.	NLOC=	2	NCAP=	
IR=	292	DMG=	0.73352E-04	TIME=	97270.	NLOC=	4	NCAP=	27
IR=	293	DMG=	0.16465E-04	TIME=	92410.	NLOC=	4	NCAP=	6
IR=	294	DMG=	0.76157E-04	TIME=	108103.	NLOC=	4	NCAP=	28
IR=	295	DMG=	0.43196E-03	TIME=	129334.	NLOC=	2	NCAP=	159
IR=	296	DMG=	0.81565E-05	TIME=	103102.	NLOC=	6	NCAP=	3
IR=	297	DMG=	0.73516E-04	TIME=	94328.	NLOC=	4	NCAP=	27
IR=	298	DMG=	0.11031E-04	TIME=	92043.	NLOC=	3	NCAP=	4
IR=	299	DMG=	0.35482E-04	TIME=	103012.	NLOC=	5	NCAP=	13
IR=	300	DMG=	0.84383E-04	TIME=	103987.	NLOC=	4	NCAP=	31
IR=	301	DMG=	0.45496E-04	TIME=	103915.	NLOC=	5	NCAP=	16
IR=	302	DMG=	0.11031E-04	TIME=	103938.	NLOC=	4	NCAP=	4
IR=	303	DMG=	0.25822E-09	TIME=	89674.	NLOC=	4	NCAP=	0
IR=	304	DMG=	0.30049E-04	TIME=	102995.	NLOC=	4	NCAP=	11
IR=	305	DMG=	0.83150E-05	TIME=	132001.	NLOC=	5	NCAP=	3
IR=	306	DMG=	0.20283E-05	TIME=	101970.	NLOC=	5	NCAP=	0
IR=	307	DMG=	0.16465E-04	TIME=	100318.	NLOC=	4	NCAP=	6
IR=	308	DMG=	0.10604E-03	TIME=	99235.	NLOC=	5	NCAP=	39
IR=	309	DMG=	0.59933E-04	TIME=	103971.	NLOC=	4	NCAP=	22
IR=	310	DMG=	0.92375E-04	TIME=	101216.	NLOC=	5	NCAP=	34
IR=	311	DMG=	0.13566E-04	TIME=	94987.	NLOC=	4	NCAP=	5
IR=	312	DMG=	0.19162E-04	TIME=	100190.	NLOC=	5	NCAP=	7
IR=	313	DMG=	0.13748E-04	TIME=	101126.	NLOC=	6	NCAP=	5
IR=	314	DMG=	0.65326E-05	TIME=	95365.	NLOC=	6	NCAP=	1
IR=	315	DMG=	0.15765E-03	TIME=	92675.	NLOC=	4	NCAP=	58
IR=	316	DMG=	0.30895E-04	TIME=	93881.	NLOC=	5	NCAP=	10
IR=	317	DMG=	0.27990E-05	TIME=	104545.	NLOC=	3	NCAP=	1
IR=	318	DMG=	0.55160E-05	TIME=	88904.	NLOC=	3	NCAP=	2
IR=	319	DMG=	0.97808E-04	TIME=	93157.	NLOC=	6	NCAP=	36
IR=	320	DMG=	0.43632E-04	TIME=	93034.	NLOC=	4	NCAP=	16
IR=	321	DMG=	0.28050E-05	TIME=	105998.	NLOC=	4	NCAP=	1
IR=	322	DMG=	0.83147E-05	TIME=	95066.	NLOC=	4	NCAP=	3
IR=	323	DMG=	0.27990E-05	TIME=	95761.	NLOC=	2	NCAP=	1
IR=	324	DMG=	0.27332E-04	TIME=	98194.	NLOC=	4	NCAP=	10
IR=	325	DMG=	0.13584E-04	TIME=	93426.	NLOC=	4	NCAP=	5
IR=	326	DMG=	0.13584E-04	TIME=	98980.	NLOC=	4	NCAP=	5
IR=	327	DMG=	0.16465E-04	TIME=	103946.	NLOC=	4	NCAP=	6
IR=	328	DMG=	0.19460E-05	TIME=	93536.	NLOC=	5	NCAP=	0
IR=	329	DMG=	0.27332E-04	TIME=	103949.	NLOC=	4	NCAP=	10
IR=	330	DMG=	0.28111E-05	TIME=	99771.	NLOC=	5	NCAP=	1
IR=	331	DMG=	0.10179E-04	TIME=	93126.	NLOC=	5	NCAP=	3
IR=	332	DMG=	0.29884E-04	TIME=	91598.	NLOC=	4	NCAP=	11
IR=	333	DMG=	0.27990E-05	TIME=	95761.	NLOC=	2	NCAP=	1
IR=	334	DMG=	0.83147E-05	TIME=	95647.	NLOC=	4	NCAP=	3
IR=	335	DMG=	0.55980E-05	TIME=	118985.	NLOC=	4	NCAP=	2
IR=	336	DMG=	0.82252E-07	TIME=	95759.	NLOC=	2	NCAP=	0
IR=	337	DMG=	0.11031E-04	TIME=	94590.	NLOC=	4	NCAP=	4
IR=	338	DMG=	0.78791E-04	TIME=	96473.	NLOC=	3	NCAP=	29
IR=	339	DMG=	0.63060E-08	TIME=	91594.	NLOC=	4	NCAP=	0
IR=	340	DMG=	0.10949E-04	TIME=	90600.	NLOC=	3	NCAP=	4
IR=	341	DMG=	0.19833E-03	TIME=	98169.	NLOC=	5	NCAP=	73
IR=	342	DMG=	0.28812E-05	TIME=	120203.	NLOC=	5	NCAP=	1
IR=	343	DMG=	0.82252E-07	TIME=	87871.	NLOC=	3	NCAP=	0
IR=	344	DMG=	0.13748E-04	TIME=	108039.	NLOC=	4	NCAP=	5
IR=	345	DMG=	0.70644E-04	TIME=	151824.	NLOC=	2	NCAP=	26
IR=	346	DMG=	0.16306E-04	TIME=	90943.	NLOC=	4	NCAP=	6
IR=	347	DMG=	0.19462E-05	TIME=	89951.	NLOC=	4	NCAP=	0
IR=	348	DMG=	0.28812E-05	TIME=	147726.	NLOC=	4	NCAP=	1
IR=	349	DMG=	0.95174E-04	TIME=	104318.	NLOC=	5	NCAP=	35
IR=	350	DMG=	0.76074E-04	TIME=	89679.	NLOC=	5	NCAP=	28

Table 5 (Contd.)

IR=	351	DMG=	0.16465E-04	TIME=	109701.	NLOC=	4	NCAP=	6
IR=	352	DMG=	0.74617E-05	TIME=	147364.	NLOC=	5	NCAP=	2
IR=	353	DMG=	0.19182E-04	TIME=	93092.	NLOC=	5	NCAP=	7
IR=	354	DMG=	0.55980E-05	TIME=	95065.	NLOC=	4	NCAP=	2
IR=	355	DMG=	0.28050E-05	TIME=	89632.	NLOC=	5	NCAP=	1
IR=	356	DMG=	0.24842E-07	TIME=	97487.	NLOC=	6	NCAP=	0
IR=	357	DMG=	0.27228E-05	TIME=	91063.	NLOC=	4	NCAP=	1
IR=	358	DMG=	0.35482E-04	TIME=	100844.	NLOC=	4	NCAP=	13
IR=	359	DMG=	0.55150E-03	TIME=	147195.	NLOC=	3	NCAP=	203
IR=	360	DMG=	0.36133E-03	TIME=	96314.	NLOC=	8	NCAP=	133
IR=	361	DMG=	0.15485E-03	TIME=	89230.	NLOC=	3	NCAP=	57
IR=	362	DMG=	0.46464E-03	TIME=	150962.	NLOC=	2	NCAP=	171
IR=	363	DMG=	0.10873E-04	TIME=	106055.	NLOC=	5	NCAP=	4
IR=	364	DMG=	0.27993E-05	TIME=	96148.	NLOC=	3	NCAP=	1
IR=	365	DMG=	0.27332E-04	TIME=	91729.	NLOC=	4	NCAP=	10
IR=	366	DMG=	0.21698E-04	TIME=	113800.	NLOC=	4	NCAP=	8
IR=	367	DMG=	0.83147E-05	TIME=	93085.	NLOC=	5	NCAP=	3
IR=	368	DMG=	0.11031E-04	TIME=	95068.	NLOC=	4	NCAP=	4
IR=	369	DMG=	0.55158E-05	TIME=	93140.	NLOC=	4	NCAP=	2
IR=	370	DMG=	0.11427E-03	TIME=	95432.	NLOC=	4	NCAP=	42
IR=	371	DMG=	0.24615E-04	TIME=	95077.	NLOC=	4	NCAP=	9
IR=	372	DMG=	0.19182E-04	TIME=	103944.	NLOC=	4	NCAP=	7
IR=	373	DMG=	0.55982E-05	TIME=	124792.	NLOC=	4	NCAP=	2
IR=	374	DMG=	0.19112E-04	TIME=	124329.	NLOC=	5	NCAP=	7
IR=	375	DMG=	0.19460E-05	TIME=	93536.	NLOC=	5	NCAP=	0
IR=	376	DMG=	0.83147E-05	TIME=	95066.	NLOC=	4	NCAP=	3
IR=	377	DMG=	0.55950E-05	TIME=	95760.	NLOC=	4	NCAP=	2
IR=	378	DMG=	0.11698E-03	TIME=	109768.	NLOC=	4	NCAP=	43
IR=	379	DMG=	0.27990E-05	TIME=	91708.	NLOC=	4	NCAP=	1
IR=	380	DMG=	0.16393E-04	TIME=	90604.	NLOC=	3	NCAP=	6
IR=	381	DMG=	0.24539E-04	TIME=	113975.	NLOC=	5	NCAP=	9
IR=	382	DMG=	0.10326E-03	TIME=	90056.	NLOC=	8	NCAP=	38
IR=	383	DMG=	0.15758E-03	TIME=	89908.	NLOC=	4	NCAP=	58
IR=	384	DMG=	0.10604E-03	TIME=	98349.	NLOC=	4	NCAP=	39
IR=	385	DMG=	0.48993E-04	TIME=	146933.	NLOC=	6	NCAP=	18
IR=	386	DMG=	0.29635E-05	TIME=	115179.	NLOC=	5	NCAP=	1
IR=	387	DMG=	0.16450E-06	TIME=	96131.	NLOC=	3	NCAP=	0
IR=	388	DMG=	0.41850E-04	TIME=	95888.	NLOC=	6	NCAP=	14
IR=	389	DMG=	0.95168E-04	TIME=	106399.	NLOC=	4	NCAP=	35
IR=	390	DMG=	0.28873E-05	TIME=	100674.	NLOC=	5	NCAP=	1
IR=	391	DMG=	0.83147E-05	TIME=	91508.	NLOC=	5	NCAP=	3
IR=	392	DMG=	0.24615E-04	TIME=	103948.	NLOC=	4	NCAP=	9
IR=	393	DMG=	0.88301E-07	TIME=	87998.	NLOC=	3	NCAP=	0
IR=	394	DMG=	0.16465E-04	TIME=	145754.	NLOC=	5	NCAP=	6
IR=	395	DMG=	0.56802E-05	TIME=	120505.	NLOC=	5	NCAP=	2
IR=	396	DMG=	0.40915E-04	TIME=	103958.	NLOC=	4	NCAP=	15
IR=	397	DMG=	0.28111E-05	TIME=	101993.	NLOC=	5	NCAP=	1
IR=	398	DMG=	0.82252E-07	TIME=	95759.	NLOC=	2	NCAP=	0
IR=	399	DMG=	0.57057E-04	TIME=	93586.	NLOC=	6	NCAP=	21
IR=	400	DMG=	0.60488E-08	TIME=	108021.	NLOC=	3	NCAP=	0
IR=	401	DMG=	0.11960E-04	TIME=	95695.	NLOC=	5	NCAP=	3
IR=	402	DMG=	0.55980E-05	TIME=	95065.	NLOC=	4	NCAP=	2
IR=	403	DMG=	0.16465E-04	TIME=	93495.	NLOC=	4	NCAP=	6
IR=	404	DMG=	0.82325E-05	TIME=	93461.	NLOC=	4	NCAP=	3
IR=	405	DMG=	0.74617E-05	TIME=	113425.	NLOC=	5	NCAP=	2
IR=	406	DMG=	0.10333E-03	TIME=	94194.	NLOC=	5	NCAP=	38
IR=	407	DMG=	0.27256E-04	TIME=	115175.	NLOC=	5	NCAP=	10
IR=	408	DMG=	0.46185E-04	TIME=	90622.	NLOC=	5	NCAP=	17
IR=	409	DMG=	0.70717E-04	TIME=	103345.	NLOC=	4	NCAP=	26
IR=	410	DMG=	0.28812E-05	TIME=	92037.	NLOC=	3	NCAP=	1

Table 5 (Contd.)

IR=	411	DMG=	0.82252E-07	TIME=	95759.	NLOC=	2	NCAP=	
IR=	412	DMG=	0.10060E-03	TIME=	135688.	NLOC=	5	NCAP=	37
IR=	413	DMG=	0.16450E-06	TIME=	137277.	NLOC=	4	NCAP=	0
IR=	414	DMG=	0.19182E-04	TIME=	94669.	NLOC=	5	NCAP=	7
IR=	415	DMG=	0.73795E-05	TIME=	88303.	NLOC=	4	NCAP=	2
IR=	416	DMG=	0.28818E-05	TIME=	97907.	NLOC=	5	NCAP=	1
IR=	417	DMG=	0.14670E-03	TIME=	93665.	NLOC=	6	NCAP=	54
IR=	418	DMG=	0.11962E-03	TIME=	92406.	NLOC=	5	NCAP=	44
IR=	419	DMG=	0.54417E-04	TIME=	93742.	NLOC=	5	NCAP=	20
IR=	420	DMG=	0.55986E-05	TIME=	99890.	NLOC=	4	NCAP=	2
IR=	421	DMG=	0.81818E-05	TIME=	92258.	NLOC=	6	NCAP=	3
IR=	422	DMG=	0.17055E-06	TIME=	112958.	NLOC=	5	NCAP=	0
IR=	423	DMG=	0.82252E-07	TIME=	95759.	NLOC=	2	NCAP=	0
IR=	424	DMG=	0.55980E-05	TIME=	104066.	NLOC=	5	NCAP=	2
IR=	425	DMG=	0.32601E-04	TIME=	144794.	NLOC=	3	NCAP=	12
IR=	426	DMG=	0.16450E-06	TIME=	113312.	NLOC=	4	NCAP=	0
IR=	427	DMG=	0.74621E-05	TIME=	88023.	NLOC=	5	NCAP=	2
IR=	428	DMG=	0.35324E-04	TIME=	136763.	NLOC=	5	NCAP=	13
IR=	429	DMG=	0.22829E-03	TIME=	105215.	NLOC=	4	NCAP=	84
IR=	430	DMG=	0.27169E-05	TIME=	117898.	NLOC=	5	NCAP=	1
IR=	431	DMG=	0.82252E-07	TIME=	95759.	NLOC=	2	NCAP=	0
IR=	432	DMG=	0.27256E-04	TIME=	97717.	NLOC=	5	NCAP=	10
IR=	433	DMG=	0.13835E-04	TIME=	116677.	NLOC=	6	NCAP=	5
IR=	434	DMG=	0.40339E-04	TIME=	90310.	NLOC=	5	NCAP=	15
IR=	435	DMG=	0.20664E-03	TIME=	120303.	NLOC=	5	NCAP=	76
IR=	436	DMG=	0.60488E-08	TIME=	142180.	NLOC=	3	NCAP=	0
IR=	437	DMG=	0.16450E-06	TIME=	94742.	NLOC=	4	NCAP=	0
IR=	438	DMG=	0.11427E-03	TIME=	114506.	NLOC=	5	NCAP=	42
IR=	439	DMG=	0.83970E-05	TIME=	118930.	NLOC=	5	NCAP=	3
IR=	440	DMG=	0.10178E-04	TIME=	95107.	NLOC=	4	NCAP=	3
IR=	441	DMG=	0.21734E-04	TIME=	103356.	NLOC=	3	NCAP=	8
IR=	442	DMG=	0.16465E-04	TIME=	110810.	NLOC=	5	NCAP=	6
IR=	443	DMG=	0.65284E-04	TIME=	118409.	NLOC=	5	NCAP=	24
IR=	444	DMG=	0.27174E-04	TIME=	134141.	NLOC=	2	NCAP=	10
IR=	445	DMG=	0.65214E-04	TIME=	88703.	NLOC=	4	NCAP=	24
IR=	446	DMG=	0.83147E-05	TIME=	94748.	NLOC=	4	NCAP=	3
IR=	447	DMG=	0.67918E-04	TIME=	93675.	NLOC=	5	NCAP=	25
IR=	448	DMG=	0.40833E-04	TIME=	95654.	NLOC=	4	NCAP=	15
IR=	449	DMG=	0.15460E-04	TIME=	100678.	NLOC=	5	NCAP=	5
IR=	450	DMG=	0.82328E-05	TIME=	96152.	NLOC=	3	NCAP=	3
IR=	451	DMG=	0.16471E-04	TIME=	90834.	NLOC=	6	NCAP=	6
IR=	452	DMG=	0.55160E-05	TIME=	99351.	NLOC=	3	NCAP=	2
IR=	453	DMG=	0.10949E-04	TIME=	90484.	NLOC=	3	NCAP=	4
IR=	454	DMG=	0.46355E-04	TIME=	106699.	NLOC=	5	NCAP=	17
IR=	455	DMG=	0.10332E-03	TIME=	99858.	NLOC=	6	NCAP=	38
IR=	456	DMG=	0.57216E-04	TIME=	105683.	NLOC=	4	NCAP=	21
IR=	457	DMG=	0.67155E-10	TIME=	111808.	NLOC=	3	NCAP=	0
IR=	458	DMG=	0.39199E-04	TIME=	103492.	NLOC=	4	NCAP=	14
IR=	459	DMG=	0.21740E-04	TIME=	100911.	NLOC=	7	NCAP=	8
IR=	460	DMG=	0.82252E-07	TIME=	95759.	NLOC=	2	NCAP=	0
IR=	461	DMG=	0.13772E-03	TIME=	90976.	NLOC=	5	NCAP=	50
IR=	462	DMG=	0.27175E-05	TIME=	106341.	NLOC=	6	NCAP=	1
IR=	463	DMG=	0.19099E-04	TIME=	97738.	NLOC=	3	NCAP=	7
IR=	464	DMG=	0.12813E-04	TIME=	88307.	NLOC=	4	NCAP=	4
IR=	465	DMG=	0.97967E-04	TIME=	113012.	NLOC=	5	NCAP=	36
IR=	466	DMG=	0.11031E-04	TIME=	93491.	NLOC=	4	NCAP=	4
IR=	467	DMG=	0.13830E-04	TIME=	120192.	NLOC=	5	NCAP=	5
IR=	468	DMG=	0.20104E-03	TIME=	129180.	NLOC=	2	NCAP=	74
IR=	469	DMG=	0.13672E-04	TIME=	103841.	NLOC=	5	NCAP=	5
IR=	470	DMG=	0.28812E-05	TIME=	94585.	NLOC=	4	NCAP=	1

Table 5 (Contd.)

IR=	471	DMG=	0.83970E-05	TIME=	117845.	NLOC=	5	NCAP=	[illegible]
IR=	472	DMG=	0.21898E-04	TIME=	94757.	NLOC=	4	NCAP=	8
IR=	473	DMG=	0.28812E-05	TIME=	93486.	NLOC=	4	NCAP=	1
IR=	474	DMG=	0.16465E-04	TIME=	115824.	NLOC=	5	NCAP=	6
IR=	475	DMG=	0.29635E-05	TIME=	115295.	NLOC=	5	NCAP=	1
IR=	476	DMG=	0.83147E-05	TIME=	103937.	NLOC=	4	NCAP=	3
IR=	477	DMG=	0.40915E-04	TIME=	103480.	NLOC=	4	NCAP=	15
IR=	478	DMG=	0.42615E-04	TIME=	90280.	NLOC=	3	NCAP=	15
IR=	479	DMG=	0.83147E-05	TIME=	113318.	NLOC=	4	NCAP=	3
IR=	480	DMG=	0.54499E-04	TIME=	109321.	NLOC=	5	NCAP=	20
IR=	481	DMG=	0.0	TIME=	89493.	NLOC=	1	NCAP=	0
IR=	482	DMG=	0.21898E-04	TIME=	93499.	NLOC=	4	NCAP=	8
IR=	483	DMG=	0.21045E-04	TIME=	100873.	NLOC=	4	NCAP=	7
IR=	484	DMG=	0.10060E-03	TIME=	96936.	NLOC=	5	NCAP=	37
IR=	485	DMG=	0.40833E-04	TIME=	95652.	NLOC=	5	NCAP=	15
IR=	486	DMG=	0.54398E-05	TIME=	91901.	NLOC=	5	NCAP=	2
IR=	487	DMG=	0.13748E-04	TIME=	121787.	NLOC=	5	NCAP=	5
IR=	488	DMG=	0.81590E-04	TIME=	96990.	NLOC=	6	NCAP=	30
IR=	489	DMG=	0.11031E-04	TIME=	93491.	NLOC=	4	NCAP=	4
IR=	490	DMG=	0.10068E-03	TIME=	120536.	NLOC=	7	NCAP=	37
IR=	491	DMG=	0.10178E-04	TIME=	95107.	NLOC=	4	NCAP=	3
IR=	492	DMG=	0.13049E-03	TIME=	94193.	NLOC=	4	NCAP=	48
IR=	493	DMG=	0.19105E-04	TIME=	106834.	NLOC=	6	NCAP=	7
IR=	494	DMG=	0.10949E-04	TIME=	90484.	NLOC=	3	NCAP=	4
IR=	495	DMG=	0.19182E-04	TIME=	103944.	NLOC=	4	NCAP=	7
IR=	496	DMG=	0.12098E-07	TIME=	114435.	NLOC=	5	NCAP=	0
IR=	497	DMG=	0.70717E-04	TIME=	100160.	NLOC=	4	NCAP=	26
IR=	498	DMG=	0.54499E-04	TIME=	109073.	NLOC=	5	NCAP=	20
IR=	499	DMG=	0.16465E-04	TIME=	93495.	NLOC=	4	NCAP=	6
IR=	500	DMG=	0.10060E-03	TIME=	102142.	NLOC=	5	NCAP=	37
IR=	501	DMG=	0.43556E-04	TIME=	99817.	NLOC=	6	NCAP=	16
IR=	502	DMG=	0.13749E-04	TIME=	99418.	NLOC=	4	NCAP=	5
IR=	503	DMG=	0.16457E-06	TIME=	88453.	NLOC=	4	NCAP=	0
IR=	504	DMG=	0.62579E-04	TIME=	101082.	NLOC=	5	NCAP=	23
IR=	505	DMG=	0.25537E-03	TIME=	165122.	NLOC=	3	NCAP=	94
IR=	506	DMG=	0.46197E-04	TIME=	94096.	NLOC=	7	NCAP=	17
IR=	507	DMG=	0.12825E-04	TIME=	108818.	NLOC=	6	NCAP=	4
IR=	508	DMG=	0.27991E-05	TIME=	101849.	NLOC=	4	NCAP=	1
IR=	509	DMG=	0.83147E-05	TIME=	92404.	NLOC=	4	NCAP=	3
IR=	510	DMG=	0.11952E-03	TIME=	148002.	NLOC=	5	NCAP=	44
IR=	511	DMG=	0.46627E-05	TIME=	93538.	NLOC=	5	NCAP=	1
IR=	512	DMG=	0.55980E-05	TIME=	87732.	NLOC=	4	NCAP=	2
IR=	513	DMG=	0.82252E-07	TIME=	95759.	NLOC=	2	NCAP=	0
IR=	514	DMG=	0.18203E-03	TIME=	99427.	NLOC=	9	NCAP=	67
IR=	515	DMG=	0.65201E-04	TIME=	110105.	NLOC=	7	NCAP=	24
IR=	516	DMG=	0.11031E-04	TIME=	103939.	NLOC=	4	NCAP=	4
IR=	517	DMG=	0.48907E-04	TIME=	110292.	NLOC=	6	NCAP=	18
IR=	518	DMG=	0.23762E-04	TIME=	103900.	NLOC=	5	NCAP=	8
IR=	519	DMG=	0.60486E-08	TIME=	105554.	NLOC=	5	NCAP=	0
IR=	520	DMG=	0.35342E-04	TIME=	92110.	NLOC=	8	NCAP=	13
IR=	521	DMG=	0.76151E-04	TIME=	93681.	NLOC=	3	NCAP=	28
IR=	522	DMG=	0.35482E-04	TIME=	95085.	NLOC=	4	NCAP=	13
IR=	523	DMG=	0.83147E-05	TIME=	101955.	NLOC=	5	NCAP=	3
IR=	524	DMG=	0.92375E-04	TIME=	90594.	NLOC=	7	NCAP=	34
IR=	525	DMG=	0.42779E-04	TIME=	106303.	NLOC=	4	NCAP=	15
IR=	526	DMG=	0.55160E-05	TIME=	99351.	NLOC=	3	NCAP=	2
IR=	527	DMG=	0.13748E-04	TIME=	95765.	NLOC=	4	NCAP=	5
IR=	528	DMG=	0.55157E-05	TIME=	95762.	NLOC=	2	NCAP=	2
IR=	529	DMG=	0.27991E-05	TIME=	90795.	NLOC=	4	NCAP=	1
IR=	530	DMG=	0.15222E-03	TIME=	88834.	NLOC=	4	NCAP=	56

Table 5 (Contd.)

IR=	531	DMG=	0.12901E-04	TIME=	109823.	NLOC=	6	NCAP=	[illegible]
IR=	532	DMG=	0.40833E-04	TIME=	133619.	NLOC=	6	NCAP=	15
IR=	533	DMG=	0.12692E-03	TIME=	88063.	NLOC=	6	NCAP=	46
IR=	534	DMG=	0.24537E-04	TIME=	94626.	NLOC=	5	NCAP=	9
IR=	535	DMG=	0.70636E-04	TIME=	88608.	NLOC=	4	NCAP=	26
IR=	536	DMG=	0.13748E-04	TIME=	95070.	NLOC=	4	NCAP=	5
IR=	537	DMG=	0.83970E-05	TIME=	120188.	NLOC=	5	NCAP=	3
IR=	538	DMG=	0.66068E-05	TIME=	114870.	NLOC=	5	NCAP=	1
IR=	539	DMG=	0.21835E-04	TIME=	101892.	NLOC=	6	NCAP=	8
IR=	540	DMG=	0.40915E-04	TIME=	109239.	NLOC=	4	NCAP=	15
IR=	541	DMG=	0.28812E-05	TIME=	98320.	NLOC=	5	NCAP=	1
IR=	542	DMG=	0.32538E-09	TIME=	90097.	NLOC=	4	NCAP=	0
IR=	543	DMG=	0.35330E-04	TIME=	91257.	NLOC=	5	NCAP=	13
IR=	544	DMG=	0.82252E-07	TIME=	95759.	NLOC=	2	NCAP=	0
IR=	545	DMG=	0.27355E-05	TIME=	91754.	NLOC=	5	NCAP=	1
IR=	546	DMG=	0.55980E-05	TIME=	95065.	NLOC=	4	NCAP=	2
IR=	547	DMG=	0.14671E-03	TIME=	93198.	NLOC=	7	NCAP=	54
IR=	548	DMG=	0.62485E-04	TIME=	116383.	NLOC=	1	NCAP=	23
IR=	549	DMG=	0.17388E-03	TIME=	93958.	NLOC=	6	NCAP=	64
IR=	550	DMG=	0.47577E-05	TIME=	96406.	NLOC=	5	NCAP=	1
IR=	551	DMG=	0.23908E-03	TIME=	162995.	NLOC=	4	NCAP=	88
IR=	552	DMG=	0.55930E-05	TIME=	95759.	NLOC=	4	NCAP=	2
IR=	553	DMG=	0.54456E-05	TIME=	88663.	NLOC=	4	NCAP=	2
IR=	554	DMG=	0.25537E-03	TIME=	130943.	NLOC=	6	NCAP=	94
IR=	555	DMG=	0.55157E-05	TIME=	109739.	NLOC=	3	NCAP=	2
IR=	556	DMG=	0.43633E-04	TIME=	112882.	NLOC=	5	NCAP=	16
IR=	557	DMG=	0.16401E-04	TIME=	105441.	NLOC=	5	NCAP=	6
IR=	558	DMG=	0.48901E-04	TIME=	100557.	NLOC=	4	NCAP=	18
IR=	559	DMG=	0.16547E-04	TIME=	120512.	NLOC=	5	NCAP=	6
IR=	560	DMG=	0.16465E-04	TIME=	92410.	NLOC=	4	NCAP=	6
IR=	561	DMG=	0.61821E-08	TIME=	116670.	NLOC=	5	NCAP=	0
IR=	562	DMG=	0.46349E-04	TIME=	104771.	NLOC=	4	NCAP=	17
IR=	563	DMG=	0.70641E-04	TIME=	95782.	NLOC=	4	NCAP=	26
IR=	564	DMG=	0.43632E-04	TIME=	94611.	NLOC=	4	NCAP=	16
IR=	565	DMG=	0.82252E-07	TIME=	117591.	NLOC=	3	NCAP=	0
IR=	566	DMG=	0.82252E-07	TIME=	122804.	NLOC=	2	NCAP=	0
IR=	567	DMG=	0.38034E-04	TIME=	98518.	NLOC=	4	NCAP=	14
IR=	568	DMG=	0.19061E-04	TIME=	100334.	NLOC=	8	NCAP=	7
IR=	569	DMG=	0.55930E-05	TIME=	91122.	NLOC=	4	NCAP=	2
IR=	570	DMG=	0.23652E-03	TIME=	92193.	NLOC=	3	NCAP=	87
IR=	571	DMG=	0.57140E-04	TIME=	119203.	NLOC=	5	NCAP=	21
IR=	572	DMG=	0.20963E-04	TIME=	99486.	NLOC=	4	NCAP=	7
IR=	573	DMG=	0.55930E-05	TIME=	95065.	NLOC=	4	NCAP=	2
IR=	574	DMG=	0.10503E-03	TIME=	90830.	NLOC=	6	NCAP=	39
IR=	575	DMG=	0.21899E-04	TIME=	99901.	NLOC=	4	NCAP=	8
IR=	576	DMG=	0.10197E-04	TIME=	105291.	NLOC=	5	NCAP=	3
IR=	577	DMG=	0.82252E-07	TIME=	93631.	NLOC=	3	NCAP=	0
IR=	578	DMG=	0.21931E-04	TIME=	121211.	NLOC=	5	NCAP=	8
IR=	579	DMG=	0.0	TIME=	100800.	NLOC=	6	NCAP=	0
IR=	580	DMG=	0.11139E-03	TIME=	122601.	NLOC=	6	NCAP=	41
IR=	581	DMG=	0.82509E-07	TIME=	87693.	NLOC=	5	NCAP=	0
IR=	582	DMG=	0.39199E-04	TIME=	94608.	NLOC=	4	NCAP=	14
IR=	583	DMG=	0.29596E-04	TIME=	102675.	NLOC=	7	NCAP=	11
IR=	584	DMG=	0.13596E-04	TIME=	98304.	NLOC=	6	NCAP=	5
IR=	585	DMG=	0.83147E-05	TIME=	111741.	NLOC=	4	NCAP=	3
IR=	586	DMG=	0.54499E-04	TIME=	95097.	NLOC=	4	NCAP=	20
IR=	587	DMG=	0.60488E-08	TIME=	142180.	NLOC=	3	NCAP=	0
IR=	588	DMG=	0.18211E-03	TIME=	119999.	NLOC=	5	NCAP=	67
IR=	589	DMG=	0.19182E-04	TIME=	94595.	NLOC=	4	NCAP=	7
IR=	590	DMG=	0.10324E-03	TIME=	186337.	NLOC=	5	NCAP=	38

Table 5 (Contd.)

IR=	591	DMG=	0.16471E-04	TIME=	112534.	NLOC=	5	NCAP=	[illegible]
IR=	592	DMG=	0.32766E-04	TIME=	118159.	NLOC=	4	NCAP=	12
IR=	593	DMG=	0.48933E-04	TIME=	93236.	NLOC=	5	NCAP=	18
IR=	594	DMG=	0.16401E-04	TIME=	111664.	NLOC=	6	NCAP=	6
IR=	595	DMG=	0.13748E-04	TIME=	92564.	NLOC=	4	NCAP=	5
IR=	596	DMG=	0.82511E-07	TIME=	130304.	NLOC=	4	NCAP=	0
IR=	597	DMG=	0.24451E-03	TIME=	116504.	NLOC=	1	NCAP=	90
IR=	598	DMG=	0.21740E-04	TIME=	95670.	NLOC=	4	NCAP=	8
IR=	599	DMG=	0.40834E-04	TIME=	94776.	NLOC=	4	NCAP=	15
IR=	600	DMG=	0.27228E-05	TIME=	91063.	NLOC=	4	NCAP=	1
IR=	601	DMG=	0.16465E-04	TIME=	103942.	NLOC=	4	NCAP=	6
IR=	602	DMG=	0.13748E-04	TIME=	92592.	NLOC=	5	NCAP=	5
IR=	603	DMG=	0.27256E-04	TIME=	97717.	NLOC=	5	NCAP=	10
IR=	604	DMG=	0.83147E-05	TIME=	89380.	NLOC=	4	NCAP=	3
IR=	605	DMG=	0.12977E-04	TIME=	120549.	NLOC=	5	NCAP=	4
IR=	606	DMG=	0.10968E-04	TIME=	87783.	NLOC=	4	NCAP=	4
IR=	607	DMG=	0.82252E-07	TIME=	122804.	NLOC=	2	NCAP=	0
IR=	608	DMG=	0.61160E-08	TIME=	104047.	NLOC=	4	NCAP=	0
IR=	609	DMG=	0.21699E-04	TIME=	94757.	NLOC=	4	NCAP=	8
IR=	610	DMG=	0.54337E-05	TIME=	108933.	NLOC=	2	NCAP=	2
IR=	611	DMG=	0.13590E-04	TIME=	101035.	NLOC=	6	NCAP=	5
IR=	612	DMG=	0.13322E-03	TIME=	104415.	NLOC=	6	NCAP=	49
IR=	613	DMG=	0.23592E-04	TIME=	102472.	NLOC=	4	NCAP=	8
IR=	614	DMG=	0.22751E-04	TIME=	99930.	NLOC=	6	NCAP=	7
IR=	615	DMG=	0.43633E-04	TIME=	91045.	NLOC=	4	NCAP=	16
IR=	616	DMG=	0.62579E-04	TIME=	114366.	NLOC=	6	NCAP=	23
IR=	617	DMG=	0.24615E-04	TIME=	94614.	NLOC=	4	NCAP=	9
IR=	618	DMG=	0.81503E-05	TIME=	95171.	NLOC=	4	NCAP=	3
IR=	619	DMG=	0.24615E-04	TIME=	116020.	NLOC=	3	NCAP=	9
IR=	620	DMG=	0.45805E-05	TIME=	92827.	NLOC=	6	NCAP=	1
IR=	621	DMG=	0.54337E-05	TIME=	97495.	NLOC=	5	NCAP=	2
IR=	622	DMG=	0.15530E-04	TIME=	87835.	NLOC=	4	NCAP=	5
IR=	623	DMG=	0.72976E-05	TIME=	104264.	NLOC=	5	NCAP=	2
IR=	624	DMG=	0.82252E-07	TIME=	95759.	NLOC=	2	NCAP=	0
IR=	625	DMG=	0.16383E-04	TIME=	102336.	NLOC=	4	NCAP=	6
IR=	626	DMG=	0.11683E-03	TIME=	115701.	NLOC=	6	NCAP=	43
IR=	627	DMG=	0.19105E-04	TIME=	100982.	NLOC=	5	NCAP=	7
IR=	628	DMG=	0.18762E-03	TIME=	100945.	NLOC=	4	NCAP=	69
IR=	629	DMG=	0.82252E-07	TIME=	95759.	NLOC=	2	NCAP=	0
IR=	630	DMG=	0.0	TIME=	90240.	NLOC=	2	NCAP=	0
IR=	631	DMG=	0.27228E-05	TIME=	92205.	NLOC=	4	NCAP=	1
IR=	632	DMG=	0.27990E-05	TIME=	87873.	NLOC=	3	NCAP=	1
IR=	633	DMG=	0.40763E-04	TIME=	98610.	NLOC=	5	NCAP=	15
IR=	634	DMG=	0.16465E-04	TIME=	96141.	NLOC=	3	NCAP=	6
IR=	635	DMG=	0.73352E-04	TIME=	130367.	NLOC=	2	NCAP=	27
IR=	636	DMG=	0.82327E-05	TIME=	105112.	NLOC=	3	NCAP=	3
IR=	637	DMG=	0.32765E-04	TIME=	93972.	NLOC=	4	NCAP=	12
IR=	638	DMG=	0.19182E-04	TIME=	130367.	NLOC=	4	NCAP=	7
IR=	639	DMG=	0.62567E-04	TIME=	99388.	NLOC=	3	NCAP=	23
IR=	640	DMG=	0.10949E-04	TIME=	109743.	NLOC=	3	NCAP=	4
IR=	641	DMG=	0.29635E-05	TIME=	118927.	NLOC=	5	NCAP=	1
IR=	642	DMG=	0.29972E-04	TIME=	100934.	NLOC=	5	NCAP=	11
IR=	643	DMG=	0.11698E-03	TIME=	92113.	NLOC=	3	NCAP=	43
IR=	644	DMG=	0.32601E-04	TIME=	116363.	NLOC=	1	NCAP=	12
IR=	645	DMG=	0.55980E-05	TIME=	93656.	NLOC=	5	NCAP=	2
IR=	646	DMG=	0.55158E-05	TIME=	102329.	NLOC=	4	NCAP=	2
IR=	647	DMG=	0.19105E-04	TIME=	112136.	NLOC=	6	NCAP=	7
IR=	648	DMG=	0.13666E-04	TIME=	99357.	NLOC=	3	NCAP=	5
IR=	649	DMG=	0.96949E-04	TIME=	93615.	NLOC=	6	NCAP=	35
IR=	650	DMG=	0.21746E-04	TIME=	111552.	NLOC=	6	NCAP=	8

Table 5 (Cc)

IR=	651	DMG=	0.83147E-05	TIME=	1221.	NLOC=	4	NCAP=	[illegible]
IR=	652	DMG=	0.27174E-04	TIME=	115047.	NLOC=	6	NCAP=	10
IR=	653	DMG=	0.45805E-05	TIME=	92827.	NLOC=	6	NCAP=	1
IR=	654	DMG=	0.55980E-05	TIME=	89356.	NLOC=	4	NCAP=	2
IR=	655	DMG=	0.97802E-04	TIME=	113608.	NLOC=	6	NCAP=	36
IR=	656	DMG=	0.11114E-04	TIME=	119021.	NLOC=	5	NCAP=	4
IR=	657	DMG=	0.27173E-04	TIME=	137914.	NLOC=	5	NCAP=	10
IR=	658	DMG=	0.19182E-04	TIME=	103944.	NLOC=	4	NCAP=	7
IR=	659	DMG=	0.54583E-05	TIME=	90429.	NLOC=	5	NCAP=	2
IR=	660	DMG=	0.20963E-04	TIME=	99486.	NLOC=	4	NCAP=	7
IR=	661	DMG=	0.64503E-05	TIME=	93203.	NLOC=	5	NCAP=	1
IR=	662	DMG=	0.28812E-05	TIME=	92401.	NLOC=	4	NCAP=	1
IR=	663	DMG=	0.43474E-04	TIME=	89775.	NLOC=	5	NCAP=	16
IR=	664	DMG=	0.60488E-08	TIME=	98493.	NLOC=	4	NCAP=	0
IR=	665	DMG=	0.13584E-04	TIME=	89755.	NLOC=	5	NCAP=	5
IR=	666	DMG=	0.23762E-04	TIME=	95116.	NLOC=	4	NCAP=	8
IR=	667	DMG=	0.19182E-04	TIME=	99899.	NLOC=	4	NCAP=	7
IR=	668	DMG=	0.55980E-05	TIME=	113316.	NLOC=	4	NCAP=	2
IR=	669	DMG=	0.12813E-04	TIME=	94398.	NLOC=	5	NCAP=	4
IR=	670	DMG=	0.35475E-04	TIME=	106017.	NLOC=	6	NCAP=	11
IR=	671	DMG=	0.29535E-05	TIME=	120185.	NLOC=	5	NCAP=	1
IR=	672	DMG=	0.35324E-04	TIME=	100783.	NLOC=	5	NCAP=	13
IR=	673	DMG=	0.38199E-04	TIME=	95086.	NLOC=	4	NCAP=	14
IR=	674	DMG=	0.55980E-05	TIME=	147250.	NLOC=	4	NCAP=	2
IR=	675	DMG=	0.46197E-04	TIME=	105719.	NLOC=	6	NCAP=	17
IR=	676	DMG=	0.19182E-04	TIME=	95074.	NLOC=	4	NCAP=	7
IR=	677	DMG=	0.81505E-05	TIME=	111554.	NLOC=	4	NCAP=	3
IR=	678	DMG=	0.19182E-04	TIME=	88761.	NLOC=	4	NCAP=	7
IR=	679	DMG=	0.28818E-05	TIME=	88356.	NLOC=	4	NCAP=	1
IR=	680	DMG=	0.28312E-05	TIME=	103933.	NLOC=	4	NCAP=	1
IR=	681	DMG=	0.55157E-05	TIME=	104228.	NLOC=	3	NCAP=	2
IR=	682	DMG=	0.16465E-04	TIME=	113796.	NLOC=	4	NCAP=	6
IR=	683	DMG=	0.83368E-07	TIME=	102452.	NLOC=	4	NCAP=	0
IR=	684	DMG=	0.29967E-04	TIME=	99367.	NLOC=	3	NCAP=	11
IR=	685	DMG=	0.83147E-05	TIME=	113317.	NLOC=	4	NCAP=	3
IR=	686	DMG=	0.19460E-05	TIME=	93536.	NLOC=	5	NCAP=	0
IR=	687	DMG=	0.62649E-04	TIME=	109254.	NLOC=	4	NCAP=	23
IR=	688	DMG=	0.40915E-04	TIME=	103957.	NLOC=	4	NCAP=	15
IR=	689	DMG=	0.40833E-04	TIME=	103325.	NLOC=	4	NCAP=	15
IR=	690	DMG=	0.13328E-03	TIME=	133927.	NLOC=	5	NCAP=	49
IR=	691	DMG=	0.35324E-04	TIME=	99568.	NLOC=	6	NCAP=	13
IR=	692	DMG=	0.27288E-05	TIME=	95722.	NLOC=	5	NCAP=	1
IR=	693	DMG=	0.76233E-04	TIME=	115500.	NLOC=	4	NCAP=	28
IR=	694	DMG=	0.27167E-04	TIME=	112043.	NLOC=	4	NCAP=	10
IR=	695	DMG=	0.82327E-05	TIME=	99353.	NLOC=	3	NCAP=	3
IR=	696	DMG=	0.19131E-04	TIME=	106546.	NLOC=	8	NCAP=	7
IR=	697	DMG=	0.83147E-05	TIME=	95067.	NLOC=	4	NCAP=	3
IR=	698	DMG=	0.40833E-04	TIME=	102020.	NLOC=	5	NCAP=	15
IR=	699	DMG=	0.38199E-04	TIME=	98528.	NLOC=	4	NCAP=	14
IR=	700	DMG=	0.73358E-04	TIME=	94448.	NLOC=	5	NCAP=	27
IR=	701	DMG=	0.47450E-05	TIME=	95103.	NLOC=	4	NCAP=	1
IR=	702	DMG=	0.62485E-04	TIME=	88170.	NLOC=	3	NCAP=	23
IR=	703	DMG=	0.22285E-03	TIME=	100471.	NLOC=	6	NCAP=	82
IR=	704	DMG=	0.74617E-05	TIME=	106279.	NLOC=	5	NCAP=	2
IR=	705	DMG=	0.82252E-07	TIME=	95759.	NLOC=	2	NCAP=	0
IR=	706	DMG=	0.62649E-04	TIME=	108975.	NLOC=	6	NCAP=	23
IR=	707	DMG=	0.11031E-04	TIME=	93491.	NLOC=	4	NCAP=	4
IR=	708	DMG=	0.62649E-04	TIME=	93729.	NLOC=	5	NCAP=	23
IR=	709	DMG=	0.18793E-07	TIME=	94250.	NLOC=	6	NCAP=	0
IR=	710	DMG=	0.10868E-03	TIME=	98185.	NLOC=	4	NCAP=	40

Table 5 (Contd.)

IR=	711	DMG=	0.16465E-04	TIME=	93495.	NLOC=	4	NCAP=	0
IR=	712	DMG=	0.56040E-05	TIME=	104061.	NLOC=	4	NCAP=	2
IR=	713	DMG=	0.20963E-04	TIME=	101742.	NLOC=	5	NCAP=	7
IR=	714	DMG=	0.11031E-04	TIME=	103938.	NLOC=	4	NCAP=	4
IR=	715	DMG=	0.55986E-05	TIME=	99890.	NLOC=	4	NCAP=	2
IR=	716	DMG=	0.11147E-03	TIME=	104601.	NLOC=	5	NCAP=	41
IR=	717	DMG=	0.13672E-04	TIME=	108061.	NLOC=	4	NCAP=	5
IR=	718	DMG=	0.17124E-03	TIME=	95069.	NLOC=	4	NCAP=	63
IR=	719	DMG=	0.19182E-04	TIME=	99899.	NLOC=	4	NCAP=	7
IR=	720	DMG=	0.27990E-05	TIME=	95761.	NLOC=	2	NCAP=	1
IR=	721	DMG=	0.11147E-03	TIME=	92166.	NLOC=	4	NCAP=	41
IR=	722	DMG=	0.16450E-06	TIME=	93484.	NLOC=	4	NCAP=	0
IR=	723	DMG=	0.28111E-05	TIME=	112842.	NLOC=	5	NCAP=	1
IR=	724	DMG=	0.19182E-04	TIME=	95074.	NLOC=	4	NCAP=	7
IR=	725	DMG=	0.43550E-04	TIME=	94426.	NLOC=	3	NCAP=	16
IR=	726	DMG=	0.82252E-07	TIME=	108733.	NLOC=	4	NCAP=	0
IR=	727	DMG=	0.84225E-04	TIME=	99262.	NLOC=	5	NCAP=	31
IR=	728	DMG=	0.12895E-04	TIME=	94704.	NLOC=	5	NCAP=	4
IR=	729	DMG=	0.19460E-05	TIME=	100150.	NLOC=	5	NCAP=	0
IR=	730	DMG=	0.78950E-04	TIME=	120697.	NLOC=	5	NCAP=	29
IR=	731	DMG=	0.24539E-04	TIME=	101408.	NLOC=	5	NCAP=	9
IR=	732	DMG=	0.21816E-04	TIME=	97429.	NLOC=	4	NCAP=	8
IR=	733	DMG=	0.29884E-03	TIME=	116541.	NLOC=	1	NCAP=	110
IR=	734	DMG=	0.55218E-05	TIME=	115510.	NLOC=	4	NCAP=	2
IR=	735	DMG=	0.11684E-03	TIME=	89914.	NLOC=	5	NCAP=	43
IR=	736	DMG=	0.27993E-05	TIME=	110287.	NLOC=	4	NCAP=	1
IR=	737	DMG=	0.10178E-04	TIME=	100866.	NLOC=	4	NCAP=	3
IR=	738	DMG=	0.46627E-05	TIME=	89622.	NLOC=	4	NCAP=	1
IR=	739	DMG=	0.89653E-04	TIME=	129106.	NLOC=	2	NCAP=	33
IR=	740	DMG=	0.29635E-05	TIME=	118927.	NLOC=	5	NCAP=	1
IR=	741	DMG=	0.12505E-03	TIME=	90006.	NLOC=	4	NCAP=	46
IR=	742	DMG=	0.74617E-05	TIME=	95105.	NLOC=	4	NCAP=	2
IR=	743	DMG=	0.43468E-04	TIME=	164981.	NLOC=	3	NCAP=	16
IR=	744	DMG=	0.19182E-04	TIME=	95074.	NLOC=	4	NCAP=	7
IR=	745	DMG=	0.19833E-03	TIME=	100832.	NLOC=	7	NCAP=	73
IR=	746	DMG=	0.38034E-04	TIME=	89829.	NLOC=	3	NCAP=	14
IR=	747	DMG=	0.24697E-04	TIME=	120517.	NLOC=	5	NCAP=	9
IR=	748	DMG=	0.22279E-03	TIME=	89932.	NLOC=	6	NCAP=	82
IR=	749	DMG=	0.60488E-08	TIME=	119712.	NLOC=	5	NCAP=	0
IR=	750	DMG=	0.23915E-03	TIME=	88802.	NLOC=	7	NCAP=	88
IR=	751	DMG=	0.38199E-04	TIME=	95036.	NLOC=	4	NCAP=	14
IR=	752	DMG=	0.56040E-05	TIME=	93614.	NLOC=	4	NCAP=	2
IR=	753	DMG=	0.83147E-05	TIME=	95066.	NLOC=	4	NCAP=	3
IR=	754	DMG=	0.70667E-04	TIME=	99810.	NLOC=	8	NCAP=	26
IR=	755	DMG=	0.19460E-05	TIME=	93073.	NLOC=	4	NCAP=	0
IR=	756	DMG=	0.28812E-05	TIME=	90002.	NLOC=	4	NCAP=	1
IR=	757	DMG=	0.51700E-04	TIME=	93182.	NLOC=	4	NCAP=	19
IR=	758	DMG=	0.27992E-05	TIME=	90595.	NLOC=	3	NCAP=	1
IR=	759	DMG=	0.15612E-04	TIME=	94706.	NLOC=	5	NCAP=	5
IR=	760	DMG=	0.40833E-04	TIME=	101773.	NLOC=	3	NCAP=	15
IR=	761	DMG=	0.55980E-05	TIME=	88457.	NLOC=	4	NCAP=	2
IR=	762	DMG=	0.55980E-05	TIME=	103939.	NLOC=	4	NCAP=	2
IR=	763	DMG=	0.82252E-07	TIME=	95759.	NLOC=	2	NCAP=	0
IR=	764	DMG=	0.16307E-04	TIME=	95070.	NLOC=	4	NCAP=	6
IR=	765	DMG=	0.31912E-04	TIME=	109346.	NLOC=	5	NCAP=	11
IR=	766	DMG=	0.18202E-03	TIME=	108201.	NLOC=	6	NCAP=	67
IR=	767	DMG=	0.54398E-05	TIME=	91598.	NLOC=	4	NCAP=	2
IR=	768	DMG=	0.10955E-04	TIME=	90768.	NLOC=	5	NCAP=	4
IR=	769	DMG=	0.10950E-04	TIME=	110608.	NLOC=	4	NCAP=	4
IR=	770	DMG=	0.16450E-06	TIME=	89853.	NLOC=	4	NCAP=	0

Table 5 (Contd.)

IR=	771	DMG=	0.82386E-05	TIME=	102457.	NLOC=	4	NCAP=	[illegible]
IR=	772	DMG=	0.13856E-03	TIME=	181597.	NLOC=	3	NCAP=	51
IR=	773	DMG=	0.27992E-05	TIME=	95502.	NLOC=	3	NCAP=	1
IR=	774	DMG=	0.40759E-03	TIME=	150924.	NLOC=	2	NCAP=	150
IR=	775	DMG=	0.35400E-04	TIME=	95693.	NLOC=	3	NCAP=	13
IR=	776	DMG=	0.27415E-05	TIME=	94168.	NLOC=	5	NCAP=	1
IR=	777	DMG=	0.16383E-04	TIME=	93043.	NLOC=	4	NCAP=	6
IR=	778	DMG=	0.55980E-05	TIME=	95065.	NLOC=	4	NCAP=	2
IR=	779	DMG=	0.47450E-05	TIME=	129776.	NLOC=	5	NCAP=	1
IR=	780	DMG=	0.82325E-05	TIME=	92947.	NLOC=	4	NCAP=	3
IR=	781	DMG=	0.28873E-05	TIME=	113744.	NLOC=	5	NCAP=	1
IR=	782	DMG=	0.24730E-03	TIME=	104074.	NLOC=	4	NCAP=	91
IR=	783	DMG=	0.18203E-03	TIME=	110518.	NLOC=	4	NCAP=	67
IR=	784	DMG=	0.32765E-04	TIME=	100842.	NLOC=	4	NCAP=	12
IR=	785	DMG=	0.30049E-04	TIME=	100640.	NLOC=	4	NCAP=	11
IR=	786	DMG=	0.74617E-05	TIME=	121822.	NLOC=	5	NCAP=	2
IR=	787	DMG=	0.43632E-04	TIME=	103482.	NLOC=	4	NCAP=	16
IR=	788	DMG=	0.15612E-04	TIME=	95025.	NLOC=	5	NCAP=	5
IR=	789	DMG=	0.81508E-04	TIME=	91899.	NLOC=	4	NCAP=	30
IR=	790	DMG=	0.47450E-05	TIME=	89348.	NLOC=	4	NCAP=	1
IR=	791	DMG=	0.74617E-05	TIME=	95105.	NLOC=	4	NCAP=	2
IR=	792	DMG=	0.29884E-04	TIME=	111276.	NLOC=	3	NCAP=	11
IR=	793	DMG=	0.19460E-05	TIME=	100906.	NLOC=	4	NCAP=	0
IR=	794	DMG=	0.55980E-05	TIME=	103935.	NLOC=	4	NCAP=	2
IR=	795	DMG=	0.19182E-04	TIME=	92412.	NLOC=	4	NCAP=	7
IR=	796	DMG=	0.28900E-05	TIME=	104060.	NLOC=	3	NCAP=	1
IR=	797	DMG=	0.11031E-04	TIME=	90744.	NLOC=	5	NCAP=	4
IR=	798	DMG=	0.97802E-04	TIME=	105765.	NLOC=	6	NCAP=	36
IR=	799	DMG=	0.83147E-05	TIME=	118733.	NLOC=	4	NCAP=	3
IR=	800	DMG=	0.81624E-05	TIME=	100050.	NLOC=	6	NCAP=	3
IR=	801	DMG=	0.83970E-05	TIME=	120188.	NLOC=	5	NCAP=	3
IR=	802	DMG=	0.24615E-04	TIME=	106959.	NLOC=	5	NCAP=	9
IR=	803	DMG=	0.54396E-05	TIME=	96380.	NLOC=	5	NCAP=	2
IR=	804	DMG=	0.16301E-04	TIME=	100475.	NLOC=	3	NCAP=	6
IR=	805	DMG=	0.55980E-05	TIME=	92403.	NLOC=	4	NCAP=	2
IR=	806	DMG=	0.12895E-04	TIME=	94705.	NLOC=	5	NCAP=	4
IR=	807	DMG=	0.56802E-05	TIME=	114192.	NLOC=	5	NCAP=	2
IR=	808	DMG=	0.15612E-04	TIME=	104965.	NLOC=	4	NCAP=	5
IR=	809	DMG=	0.82319E-07	TIME=	91878.	NLOC=	4	NCAP=	0
IR=	810	DMG=	0.82252E-07	TIME=	95759.	NLOC=	2	NCAP=	0
IR=	811	DMG=	0.73434E-04	TIME=	89612.	NLOC=	6	NCAP=	27
IR=	812	DMG=	0.54335E-04	TIME=	90379.	NLOC=	4	NCAP=	20
IR=	813	DMG=	0.24615E-04	TIME=	98538.	NLOC=	5	NCAP=	9
IR=	814	DMG=	0.11333E-03	TIME=	95218.	NLOC=	4	NCAP=	41
IR=	815	DMG=	0.10955E-04	TIME=	123315.	NLOC=	5	NCAP=	4
IR=	816	DMG=	0.55157E-05	TIME=	131668.	NLOC=	3	NCAP=	2
IR=	817	DMG=	0.11140E-03	TIME=	103394.	NLOC=	7	NCAP=	41
IR=	818	DMG=	0.37275E-05	TIME=	117279.	NLOC=	6	NCAP=	0
IR=	819	DMG=	0.22014E-03	TIME=	168580.	NLOC=	3	NCAP=	81
IR=	820	DMG=	0.43632E-04	TIME=	107738.	NLOC=	5	NCAP=	16
IR=	821	DMG=	0.27991E-05	TIME=	146120.	NLOC=	4	NCAP=	1
IR=	822	DMG=	0.10949E-04	TIME=	99355.	NLOC=	3	NCAP=	4
IR=	823	DMG=	0.32765E-04	TIME=	92057.	NLOC=	3	NCAP=	12
IR=	824	DMG=	0.14759E-04	TIME=	94745.	NLOC=	5	NCAP=	4
IR=	825	DMG=	0.28873E-05	TIME=	121425.	NLOC=	6	NCAP=	1
IR=	826	DMG=	0.97802E-04	TIME=	101773.	NLOC=	4	NCAP=	36
IR=	827	DMG=	0.14677E-04	TIME=	106986.	NLOC=	5	NCAP=	4
IR=	828	DMG=	0.11691E-03	TIME=	102335.	NLOC=	4	NCAP=	43
IR=	829	DMG=	0.19460E-05	TIME=	93536.	NLOC=	5	NCAP=	0
IR=	830	DMG=	0.20110E-04	TIME=	96281.	NLOC=	5	NCAP=	5

Table 5 (Contd.)

IR=	831	DMG=	0.55980E-05	TIME=	103935.	NLOC=	4	NCAP=	2
IR=	832	DMG=	0.92533E-04	TIME=	115511.	NLOC=	4	NCAP=	34
IR=	833	DMG=	0.32765E-04	TIME=	109712.	NLOC=	4	NCAP=	12
IR=	834	DMG=	0.55980E-05	TIME=	94746.	NLOC=	4	NCAP=	2
IR=	835	DMG=	0.27992E-05	TIME=	99349.	NLOC=	3	NCAP=	1
IR=	836	DMG=	0.37270E-04	TIME=	91300.	NLOC=	5	NCAP=	13
IR=	837	DMG=	0.28812E-05	TIME=	95758.	NLOC=	4	NCAP=	1
IR=	838	DMG=	0.70717E-04	TIME=	93678.	NLOC=	3	NCAP=	26
IR=	839	DMG=	0.21822E-04	TIME=	95851.	NLOC=	6	NCAP=	8
IR=	840	DMG=	0.12895E-04	TIME=	103893.	NLOC=	5	NCAP=	4
IR=	841	DMG=	0.28812E-05	TIME=	95063.	NLOC=	4	NCAP=	1
IR=	842	DMG=	0.10949E-04	TIME=	139733.	NLOC=	4	NCAP=	4
IR=	843	DMG=	0.72972E-05	TIME=	96839.	NLOC=	5	NCAP=	2
IR=	844	DMG=	0.78786E-04	TIME=	92663.	NLOC=	6	NCAP=	29
IR=	845	DMG=	0.60482E-08	TIME=	89593.	NLOC=	3	NCAP=	0
IR=	846	DMG=	0.13745E-04	TIME=	100262.	NLOC=	4	NCAP=	5
IR=	847	DMG=	0.32057E-03	TIME=	165166.	NLOC=	3	NCAP=	118
IR=	848	DMG=	0.57051E-04	TIME=	114594.	NLOC=	5	NCAP=	21
IR=	849	DMG=	0.55278E-05	TIME=	115454.	NLOC=	6	NCAP=	2
IR=	850	DMG=	0.12769E-03	TIME=	116426.	NLOC=	1	NCAP=	47
IR=	851	DMG=	0.11950E-04	TIME=	89261.	NLOC=	6	NCAP=	3
IR=	852	DMG=	0.27332E-04	TIME=	103949.	NLOC=	4	NCAP=	10
IR=	853	DMG=	0.0	TIME=	100900.	NLOC=	6	NCAP=	0
IR=	854	DMG=	0.82513E-05	TIME=	114069.	NLOC=	8	NCAP=	3
IR=	855	DMG=	0.55160E-05	TIME=	94320.	NLOC=	3	NCAP=	2
IR=	856	DMG=	0.29113E-04	TIME=	89539.	NLOC=	4	NCAP=	10
IR=	857	DMG=	0.83147E-05	TIME=	94748.	NLOC=	4	NCAP=	3
IR=	858	DMG=	0.16393E-04	TIME=	118246.	NLOC=	5	NCAP=	6
IR=	859	DMG=	0.27992E-05	TIME=	83902.	NLOC=	3	NCAP=	1
IR=	860	DMG=	0.24539E-04	TIME=	89317.	NLOC=	4	NCAP=	9
IR=	861	DMG=	0.20283E-05	TIME=	89346.	NLOC=	4	NCAP=	0
IR=	862	DMG=	0.74617E-05	TIME=	115064.	NLOC=	5	NCAP=	2
IR=	863	DMG=	0.20192E-04	TIME=	113791.	NLOC=	5	NCAP=	6
IR=	864	DMG=	0.46190E-04	TIME=	99255.	NLOC=	5	NCAP=	17
IR=	865	DMG=	0.70641E-04	TIME=	102345.	NLOC=	4	NCAP=	26
IR=	866	DMG=	0.19182E-04	TIME=	113797.	NLOC=	4	NCAP=	7
IR=	867	DMG=	0.38116E-04	TIME=	93063.	NLOC=	4	NCAP=	14
IR=	868	DMG=	0.67941E-04	TIME=	105363.	NLOC=	7	NCAP=	25
IR=	869	DMG=	0.82252E-07	TIME=	95759.	NLOC=	2	NCAP=	0
IR=	870	DMG=	0.29535E-03	TIME=	91659.	NLOC=	7	NCAP=	108
IR=	871	DMG=	0.83147E-05	TIME=	95066.	NLOC=	4	NCAP=	3
IR=	872	DMG=	0.82252E-07	TIME=	95759.	NLOC=	2	NCAP=	0
IR=	873	DMG=	0.23908E-03	TIME=	134282.	NLOC=	2	NCAP=	88
IR=	874	DMG=	0.27332E-04	TIME=	89871.	NLOC=	4	NCAP=	10
IR=	875	DMG=	0.27166E-05	TIME=	111809.	NLOC=	3	NCAP=	1
IR=	876	DMG=	0.83147E-05	TIME=	95066.	NLOC=	4	NCAP=	3
IR=	877	DMG=	0.19026E-03	TIME=	90896.	NLOC=	5	NCAP=	70
IR=	878	DMG=	0.24676E-06	TIME=	118606.	NLOC=	5	NCAP=	0
IR=	879	DMG=	0.10516E-03	TIME=	94127.	NLOC=	4	NCAP=	38
IR=	880	DMG=	0.65208E-04	TIME=	90863.	NLOC=	6	NCAP=	24
IR=	881	DMG=	0.16465E-04	TIME=	103946.	NLOC=	4	NCAP=	6
IR=	882	DMG=	0.49172E-03	TIME=	130645.	NLOC=	2	NCAP=	181
IR=	883	DMG=	0.21898E-04	TIME=	96598.	NLOC=	4	NCAP=	8
IR=	884	DMG=	0.31748E-04	TIME=	95754.	NLOC=	6	NCAP=	11
IR=	885	DMG=	0.55157E-03	TIME=	199618.	NLOC=	4	NCAP=	203
IR=	886	DMG=	0.10053E-03	TIME=	124066.	NLOC=	5	NCAP=	37
IR=	887	DMG=	0.16450E-06	TIME=	112993.	NLOC=	4	NCAP=	0
IR=	888	DMG=	0.11031E-04	TIME=	103938.	NLOC=	4	NCAP=	4
IR=	889	DMG=	0.29635E-05	TIME=	114191.	NLOC=	5	NCAP=	1
IR=	890	DMG=	0.82252E-07	TIME=	95759.	NLOC=	2	NCAP=	0

Table 5 (Contd.)

IR=	891	DMG=	0.29979E-04	TIME=	107030.	NLOC=	5	NCAP=	1.
IR=	892	DMG=	0.83970E-05	TIME=	120507.	NLOC=	5	NCAP=	3
IR=	893	DMG=	0.18638E-05	TIME=	92825.	NLOC=	6	NCAP=	0
IR=	894	DMG=	0.46273E-04	TIME=	126663.	NLOC=	5	NCAP=	17
IR=	895	DMG=	0.49066E-04	TIME=	99919.	NLOC=	4	NCAP=	18
IR=	896	DMG=	0.12096E-07	TIME=	95895.	NLOC=	6	NCAP=	0
IR=	897	DMG=	0.32766E-04	TIME=	102596.	NLOC=	5	NCAP=	12
IR=	898	DMG=	0.19460E-05	TIME=	93536.	NLOC=	5	NCAP=	0
IR=	899	DMG=	0.26628E-03	TIME=	94154.	NLOC=	9	NCAP=	98
IR=	900	DMG=	0.55980E-05	TIME=	95065.	NLOC=	4	NCAP=	2
IR=	901	DMG=	0.55980E-05	TIME=	89309.	NLOC=	4	NCAP=	2
IR=	902	DMG=	0.10096E-04	TIME=	90523.	NLOC=	3	NCAP=	3
IR=	903	DMG=	0.13748E-04	TIME=	95070.	NLOC=	4	NCAP=	5
IR=	904	DMG=	0.17399E-03	TIME=	92980.	NLOC=	6	NCAP=	64
IR=	905	DMG=	0.11032E-04	TIME=	116011.	NLOC=	3	NCAP=	4
IR=	906	DMG=	0.0	TIME=	97354.	NLOC=	2	NCAP=	0
IR=	907	DMG=	0.10178E-04	TIME=	93126.	NLOC=	5	NCAP=	3
IR=	908	DMG=	0.19560E-03	TIME=	89934.	NLOC=	3	NCAP=	72
IR=	909	DMG=	0.10096E-04	TIME=	93918.	NLOC=	5	NCAP=	3
IR=	910	DMG=	0.27990E-05	TIME=	95761.	NLOC=	2	NCAP=	1
IR=	911	DMG=	0.83147E-05	TIME=	95067.	NLOC=	4	NCAP=	3
IR=	912	DMG=	0.13748E-04	TIME=	114636.	NLOC=	4	NCAP=	5
IR=	913	DMG=	0.11031E-04	TIME=	103939.	NLOC=	4	NCAP=	4
IR=	914	DMG=	0.27192E-04	TIME=	88605.	NLOC=	5	NCAP=	10
IR=	915	DMG=	0.60488E-08	TIME=	144791.	NLOC=	4	NCAP=	0
IR=	916	DMG=	0.16393E-04	TIME=	105117.	NLOC=	3	NCAP=	6
IR=	917	DMG=	0.32765E-04	TIME=	93506.	NLOC=	4	NCAP=	12
IR=	918	DMG=	0.54417E-04	TIME=	89247.	NLOC=	4	NCAP=	20
IR=	919	DMG=	0.59933E-04	TIME=	100541.	NLOC=	4	NCAP=	22
IR=	920	DMG=	0.19460E-05	TIME=	88300.	NLOC=	4	NCAP=	0
IR=	921	DMG=	0.13748E-04	TIME=	100829.	NLOC=	4	NCAP=	5
IR=	922	DMG=	0.10596E-03	TIME=	95806.	NLOC=	4	NCAP=	39
IR=	923	DMG=	0.16389E-04	TIME=	105663.	NLOC=	4	NCAP=	6
IR=	924	DMG=	0.16399E-04	TIME=	113024.	NLOC=	7	NCAP=	6
IR=	925	DMG=	0.82252E-07	TIME=	95759.	NLOC=	2	NCAP=	0
IR=	926	DMG=	0.83147E-05	TIME=	98181.	NLOC=	4	NCAP=	3
IR=	927	DMG=	0.22622E-16	TIME=	107437.	NLOC=	4	NCAP=	0
IR=	928	DMG=	0.0	TIME=	90240.	NLOC=	2	NCAP=	0
IR=	929	DMG=	0.54499E-04	TIME=	88305.	NLOC=	4	NCAP=	20
IR=	930	DMG=	0.12098E-07	TIME=	101587.	NLOC=	5	NCAP=	0
IR=	931	DMG=	0.50929E-04	TIME=	109677.	NLOC=	5	NCAP=	18
IR=	932	DMG=	0.11954E-03	TIME=	100390.	NLOC=	5	NCAP=	44
IR=	933	DMG=	0.60493E-08	TIME=	89593.	NLOC=	3	NCAP=	0
IR=	934	DMG=	0.84301E-04	TIME=	106392.	NLOC=	4	NCAP=	31
IR=	935	DMG=	0.15612E-04	TIME=	106285.	NLOC=	5	NCAP=	5
IR=	936	DMG=	0.18474E-03	TIME=	91624.	NLOC=	5	NCAP=	68
IR=	937	DMG=	0.84225E-04	TIME=	97614.	NLOC=	5	NCAP=	31
IR=	938	DMG=	0.83147E-05	TIME=	94748.	NLOC=	4	NCAP=	3
IR=	939	DMG=	0.95250E-04	TIME=	100565.	NLOC=	4	NCAP=	35
IR=	940	DMG=	0.67925E-04	TIME=	134168.	NLOC=	2	NCAP=	25
IR=	941	DMG=	0.8[illegible]669E-07	TIME=	89786.	NLOC=	4	NCAP=	0
IR=	942	DMG=	0.18219E-03	TIME=	202735.	NLOC=	4	NCAP=	67
IR=	943	DMG=	0.65208E-04	TIME=	94276.	NLOC=	5	NCAP=	24
IR=	944	DMG=	0.24615E-04	TIME=	112254.	NLOC=	5	NCAP=	9
IR=	945	DMG=	0.19521E-05	TIME=	89746.	NLOC=	4	NCAP=	0
IR=	946	DMG=	0.55157E-05	TIME=	95762.	NLOC=	2	NCAP=	2
IR=	947	DMG=	0.16319E-04	TIME=	88471.	NLOC=	5	NCAP=	6
IR=	948	DMG=	0.29884E-04	TIME=	96477.	NLOC=	5	NCAP=	11
IR=	949	DMG=	0.21120E-04	TIME=	120555.	NLOC=	5	NCAP=	7
IR=	950	DMG=	0.84301E-04	TIME=	100633.	NLOC=	4	NCAP=	31

Table 5 (Contd.)

IR=	951	DMG=	0.24615E-04	TIME=	93500.	NLOC=	4	NCAP=	[illegible]
IR=	952	DMG=	0.55157E-05	TIME=	123271.	NLOC=	3	NCAP=	2
IR=	953	DMG=	0.48907E-04	TIME=	99906.	NLOC=	4	NCAP=	18
IR=	954	DMG=	0.20655E-03	TIME=	198167.	NLOC=	3	NCAP=	76
IR=	955	DMG=	0.25909E-09	TIME=	101842.	NLOC=	3	NCAP=	0
IR=	956	DMG=	0.19106E-04	TIME=	94989.	NLOC=	5	NCAP=	7
IR=	957	DMG=	0.16465E-04	TIME=	95072.	NLOC=	4	NCAP=	6
IR=	958	DMG=	0.56113E-05	TIME=	92293.	NLOC=	4	NCAP=	2
IR=	959	DMG=	0.19182E-04	TIME=	93496.	NLOC=	4	NCAP=	7
IR=	960	DMG=	0.59933E-04	TIME=	103507.	NLOC=	4	NCAP=	22
IR=	961	DMG=	0.12098E-07	TIME=	88049.	NLOC=	5	NCAP=	0
IR=	962	DMG=	0.16852E-03	TIME=	102850.	NLOC=	6	NCAP=	62
IR=	963	DMG=	0.42652E-03	TIME=	166721.	NLOC=	4	NCAP=	157
IR=	964	DMG=	0.10612E-03	TIME=	112309.	NLOC=	5	NCAP=	39
IR=	965	DMG=	0.13594E-03	TIME=	107101.	NLOC=	7	NCAP=	50
IR=	966	DMG=	0.55990E-05	TIME=	94597.	NLOC=	4	NCAP=	2
IR=	967	DMG=	0.19182E-04	TIME=	94755.	NLOC=	4	NCAP=	7
IR=	968	DMG=	0.16465E-04	TIME=	147735.	NLOC=	4	NCAP=	6
IR=	969	DMG=	0.27250E-04	TIME=	95687.	NLOC=	3	NCAP=	10
IR=	970	DMG=	0.13748E-04	TIME=	103940.	NLOC=	4	NCAP=	5
IR=	971	DMG=	0.43632E-04	TIME=	117936.	NLOC=	5	NCAP=	16
IR=	972	DMG=	0.16450E-06	TIME=	92036.	NLOC=	3	NCAP=	0
IR=	973	DMG=	0.39121E-03	TIME=	129307.	NLOC=	2	NCAP=	144
IR=	974	DMG=	0.61150E-08	TIME=	87916.	NLOC=	5	NCAP=	0
IR=	975	DMG=	0.89658E-04	TIME=	97722.	NLOC=	7	NCAP=	33
IR=	976	DMG=	0.11031E-04	TIME=	93491.	NLOC=	4	NCAP=	4
IR=	977	DMG=	0.74042E-05	TIME=	101681.	NLOC=	5	NCAP=	2
IR=	978	DMG=	0.13594E-03	TIME=	116432.	NLOC=	1	NCAP=	50
IR=	979	DMG=	0.11031E-04	TIME=	89313.	NLOC=	4	NCAP=	4
IR=	980	DMG=	0.40834E-04	TIME=	121477.	NLOC=	5	NCAP=	15
IR=	981	DMG=	0.55157E-05	TIME=	95762.	NLOC=	2	NCAP=	2
IR=	982	DMG=	0.25545E-03	TIME=	173834.	NLOC=	4	NCAP=	94
IR=	983	DMG=	0.30049E-04	TIME=	92056.	NLOC=	3	NCAP=	11
IR=	984	DMG=	0.32689E-04	TIME=	88034.	NLOC=	5	NCAP=	12
IR=	985	DMG=	0.21740E-04	TIME=	105569.	NLOC=	5	NCAP=	8
IR=	986	DMG=	0.81565E-05	TIME=	93430.	NLOC=	4	NCAP=	3
IR=	987	DMG=	0.46267E-04	TIME=	89878.	NLOC=	7	NCAP=	17
IR=	988	DMG=	0.29972E-04	TIME=	91372.	NLOC=	5	NCAP=	11
IR=	989	DMG=	0.16465E-04	TIME=	107720.	NLOC=	5	NCAP=	6
IR=	990	DMG=	0.32765E-04	TIME=	100842.	NLOC=	4	NCAP=	12
IR=	991	DMG=	0.87024E-04	TIME=	89540.	NLOC=	4	NCAP=	32
IR=	992	DMG=	0.11025E-04	TIME=	93972.	NLOC=	6	NCAP=	2
IR=	993	DMG=	0.35482E-04	TIME=	94606.	NLOC=	4	NCAP=	13
IR=	994	DMG=	0.15757E-03	TIME=	87814.	NLOC=	6	NCAP=	58
IR=	995	DMG=	0.55986E-05	TIME=	89443.	NLOC=	4	NCAP=	2
IR=	996	DMG=	0.26633E-03	TIME=	97288.	NLOC=	5	NCAP=	98
IR=	997	DMG=	0.10949E-04	TIME=	119090.	NLOC=	4	NCAP=	4
IR=	998	DMG=	0.60488E-08	TIME=	95161.	NLOC=	4	NCAP=	0
IR=	999	DMG=	0.83147E-05	TIME=	96136.	NLOC=	3	NCAP=	3
IR=	1000	DMG=	0.10950E-04	TIME=	96154.	NLOC=	3	NCAP=	4

Table 6 - Sidewinder Rocket Damage in 10 Years (Sorted in Ascending Order).

TIME= 87600. (Hrs.)

1	0.0	2	0.0	3	0.0	4	0.0	5	0.0
6	0.0	7	0.0	8	0.0	9	0.0	10	0.1024D-16
11	0.1433D-10	12	0.6715D-10	13	0.6715D-10	14	0.2572D-09	15	0.2582D-09
16	0.2591D-09	17	0.3138D-09	18	0.5211D-09	19	0.3477D-08	20	0.6008D-08
21	0.6049D-08	22	0.6049D-08	23	0.6049D-08	24	0.6049D-08	25	0.6049D-08
26	0.6049D-08	27	0.6049D-08	28	0.6049D-08	29	0.6049D-08	30	0.6116D-08
31	0.6182D-08	32	0.6285D-08	33	0.7339D-08	34	0.1195D-07	35	0.1210D-07
36	0.1210D-07	37	0.1210D-07	38	0.1210D-07	39	0.1653D-07	40	0.2215D-07
41	0.2484D-07	42	0.2569D-07	43	0.5276D-07	44	0.6269D-07	45	0.6269D-07
46	0.6269D-07	47	0.6269D-07	48	0.6269D-07	49	0.6269D-07	50	0.6269D-07
51	0.6269D-07	52	0.6780D-07	53	0.8225D-07	54	0.8225D-07	55	0.8225D-07
56	0.8225D-07	57	0.8225D-07	58	0.8225D-07	59	0.8225D-07	60	0.8225D-07
61	0.8225D-07	62	0.8225D-07	63	0.8225D-07	64	0.8225D-07	65	0.8225D-07
66	0.8225D-07	67	0.8225D-07	68	0.8225D-07	69	0.8225D-07	70	0.8225D-07
71	0.8225D-07	72	0.8225D-07	73	0.8225D-07	74	0.8228D-07	75	0.8232D-07
76	0.8232D-07	77	0.8251D-07	78	0.8343D-07	79	0.8735D-07	80	0.1036D-06
81	0.1098D-06	82	0.1253D-06	83	0.1311D-06	84	0.1338D-06	85	0.1441D-06
86	0.1504D-06	87	0.1539D-06	88	0.1539D-06	89	0.1645D-06	90	0.1645D-06
91	0.1645D-06	92	0.1645D-06	93	0.1645D-06	94	0.1645D-06	95	0.1645D-06
96	0.1645D-06	97	0.1686D-06	98	0.1724D-06	99	0.1726D-06	100	0.7902D-06
101	0.1864D-05	102	0.1946D-05	103	0.1946D-05	104	0.1946D-05	105	0.1946D-05
106	0.1946D-05	107	0.1946D-05	108	0.1946D-05	109	0.1946D-05	110	0.1946D-05
111	0.1946D-05	112	0.1947D-05	113	0.1994D-05	114	0.2024D-05	115	0.2717D-05
116	0.2717D-05	117	0.2717D-05	118	0.2722D-05	119	0.2723D-05	120	0.2723D-05
121	0.2723D-05	122	0.2723D-05	123	0.2723D-05	124	0.2726D-05	125	0.2729D-05
126	0.2730D-05	127	0.2741D-05	128	0.2751D-05	129	0.2779D-05	130	0.2779D-05
131	0.2779D-05	132	0.2782D-05	133	0.2799D-05	134	0.2799D-05	135	0.2799D-05
136	0.2799D-05	137	0.2799D-05	138	0.2799D-05	139	0.2799D-05	140	0.2799D-05
141	0.2799D-05	142	0.2799D-05	143	0.2799D-05	144	0.2799D-05	145	0.2799D-05
146	0.2799D-05	147	0.2799D-05	148	0.2799D-05	149	0.2800D-05	150	0.2804D-05
151	0.2805D-05	152	0.2805D-05	153	0.2806D-05	154	0.2806D-05	155	0.2825D-05
156	0.2837D-05	157	0.2842D-05	158	0.2851D-05	159	0.2856D-05	160	0.2856D-05
161	0.2863D-05	162	0.2864D-05	163	0.2871D-05	164	0.2881D-05	165	0.2881D-05
166	0.2881D-05	167	0.2881D-05	168	0.2881D-05	169	0.2881D-05	170	0.2881D-05
171	0.2881D-05	172	0.2881D-05	173	0.2881D-05	174	0.2881D-05	175	0.2881D-05
176	0.2881D-05	177	0.2881D-05	178	0.2882D-05	179	0.2885D-05	180	0.2888D-05
181	0.2892D-05	182	0.2893D-05	183	0.2897D-05	184	0.2897D-05	185	0.2900D-05
186	0.2900D-05	187	0.3160D-05	188	0.3728D-05	189	0.4376D-05	190	0.4580D-05
191	0.4580D-05	192	0.4587D-05	193	0.4663D-05	194	0.4663D-05	195	0.4663D-05
196	0.4727D-05	197	0.4745D-05	198	0.4745D-05	199	0.4753D-05	200	0.4787D-05
201	0.5267D-05	202	0.5434D-05	203	0.5434D-05	204	0.5438D-05	205	0.5440D-05
206	0.5440D-05	207	0.5440D-05	208	0.5440D-05	209	0.5442D-05	210	0.5444D-05
211	0.5445D-05	212	0.5450D-05	213	0.5455D-05	214	0.5457D-05	215	0.5461D-05
216	0.5462D-05	217	0.5463D-05	218	0.5496D-05	219	0.5496D-05	220	0.5513D-05
221	0.5516D-05	222	0.5516D-05	223	0.5516D-05	224	0.5516D-05	225	0.5516D-05
226	0.5516D-05	227	0.5516D-05	228	0.5516D-05	229	0.5516D-05	230	0.5516D-05
231	0.5516D-05	232	0.5516D-05	233	0.5523D-05	234	0.5536D-05	235	0.5557D-05
236	0.5559D-05	237	0.5560D-05	238	0.5580D-05	239	0.5581D-05	240	0.5586D-05
241	0.5590D-05	242	0.5590D-05	243	0.5590D-05	244	0.5593D-05	245	0.5596D-05
246	0.5598D-05	247	0.5598D-05	248	0.5598D-05	249	0.5598D-05	250	0.5598D-05
251	0.5598D-05	252	0.5598D-05	253	0.5598D-05	254	0.5598D-05	255	0.5598D-05
256	0.5598D-05	257	0.5598D-05	258	0.5598D-05	259	0.5598D-05	260	0.5598D-05
261	0.5598D-05	262	0.5598D-05	263	0.5598D-05	264	0.5598D-05	265	0.5598D-05
266	0.5598D-05	267	0.5598D-05	268	0.5598D-05	269	0.5598D-05	270	0.5598D-05
271	0.5599D-05	272	0.5601D-05	273	0.5601D-05	274	0.5605D-05	275	0.5605D-05
276	0.5616D-05	277	0.5877D-05	278	0.5877D-05	279	0.5877D-05	280	0.6543D-05
281	0.7231D-05	282	0.7297D-05	283	0.7298D-05	284	0.7398D-05	285	0.7399D-05
286	0.7400D-05	287	0.7417D-05	288	0.7462D-05	289	0.7462D-05	290	0.7462D-05
291	0.7[illegible]62D-05	292	0.7465D-05	293	0.8150D-05	294	0.8150D-05	295	0.8156D-05

Table 6 (Contd.)

296	0.8156D-05	297	0.8156D-05	298	0.8158D-05	299	0.8169D-05	300	0.8176D-05
301	0.8188D-05	302	0.8193D-05	303	0.8203D-05	304	0.8206D-05	305	0.8212D-05
306	0.8220D-05	307	0.8220D-05	308	0.8233D-05	309	0.8233D-05	310	0.8233D-05
311	0.8233D-05	312	0.8233D-05	313	0.8233D-05	314	0.8239D-05	315	0.8242D-05
316	0.8253D-05	317	0.8257D-05	318	0.8294D-05	319	0.8295D-05	320	0.8297D-05
321	0.8297D-05	322	0.8297D-05	323	0.8303D-05	324	0.8315D-05	325	0.8315D-05
326	0.8315D-05	327	0.8315D-05	328	0.8315D-05	329	0.8315D-05	330	0.8315D-05
331	0.8315D-05	332	0.8315D-05	333	0.8315D-05	334	0.8315D-05	335	0.8315D-05
336	0.8315D-05	337	0.8315D-05	338	0.8315D-05	339	0.8315D-05	340	0.8315D-05
341	0.8315D-05	342	0.8315D-05	343	0.8315D-05	344	0.8315D-05	345	0.8315D-05
346	0.8315D-05	347	0.8315D-05	348	0.8318D-05	349	0.8319D-05	350	0.8319D-05
351	0.8322D-05	352	0.8324D-05	353	0.8324D-05	354	0.8593D-05	355	0.9011D-05
356	0.9307D-05	357	0.9479D-05	358	0.1010D-04	359	0.1012D-04	360	0.1012D-04
361	0.1015D-04	362	0.1018D-04	363	0.1018D-04	364	0.1019D-04	365	0.1087D-04
366	0.1087D-04	367	0.1090D-04	368	0.1093D-04	369	0.1095D-04	370	0.1095D-04
371	0.1095D-04	372	0.1095D-04	373	0.1095D-04	374	0.1095D-04	375	0.1095D-04
376	0.1095D-04	377	0.1096D-04	378	0.1096D-04	379	0.1096D-04	380	0.1097D-04
381	0.1099D-04	382	0.1099D-04	383	0.1099D-04	384	0.1101D-04	385	0.1101D-04
386	0.1102D-04	387	0.1102D-04	388	0.1103D-04	389	0.1103D-04	390	0.1103D-04
391	0.1103D-04	392	0.1103D-04	393	0.1103D-04	394	0.1103D-04	395	0.1103D-04
396	0.1103D-04	397	0.1103D-04	398	0.1103D-04	399	0.1103D-04	400	0.1103D-04
401	0.1103D-04	402	0.1103D-04	403	0.1103D-04	404	0.1103D-04	405	0.1104D-04
406	0.1104D-04	407	0.1106D-04	408	0.1139D-04	409	0.1161D-04	410	0.1196D-04
411	0.1197D-04	412	0.1244D-04	413	0.1266D-04	414	0.1266D-04	415	0.1273D-04
416	0.1275D-04	417	0.1281D-04	418	0.1283D-04	419	0.1284D-04	420	0.1285D-04
421	0.1288D-04	422	0.1290D-04	423	0.1290D-04	424	0.1290D-04	425	0.1325D-04
426	0.1352D-04	427	0.1352D-04	428	0.1358D-04	429	0.1358D-04	430	0.1359D-04
431	0.1359D-04	432	0.1360D-04	433	0.1367D-04	434	0.1367D-04	435	0.1367D-04
436	0.1367D-04	437	0.1367D-04	438	0.1367D-04	439	0.1368D-04	440	0.1372D-04
441	0.1372D-04	442	0.1372D-04	443	0.1374D-04	444	0.1374D-04	445	0.1375D-04
446	0.1375D-04	447	0.1375D-04	448	0.1375D-04	449	0.1375D-04	450	0.1375D-04
451	0.1375D-04	452	0.1375D-04	453	0.1375D-04	454	0.1375D-04	455	0.1375D-04
456	0.1375D-04	457	0.1375D-04	458	0.1375D-04	459	0.1375D-04	460	0.1376D-04
461	0.1377D-04	462	0.1411D-04	463	0.1459D-04	464	0.1463D-04	465	0.1545D-04
466	0.1546D-04	467	0.1550D-04	468	0.1550D-04	469	0.1553D-04	470	0.1557D-04
471	0.1557D-04	472	0.1559D-04	473	0.1561D-04	474	0.1630D-04	475	0.1630D-04
476	0.1631D-04	477	0.1631D-04	478	0.1631D-04	479	0.1631D-04	480	0.1632D-04
481	0.1633D-04	482	0.1634D-04	483	0.1636D-04	484	0.1637D-04	485	0.1638D-04
486	0.1638D-04	487	0.1638D-04	488	0.1639D-04	489	0.1639D-04	490	0.1640D-04
491	0.1640D-04	492	0.1641D-04	493	0.1641D-04	494	0.1643D-04	495	0.1643D-04
496	0.1644D-04	497	0.1644D-04	498	0.1645D-04	499	0.1645D-04	500	0.1646D-04
501	0.1646D-04	502	0.1646D-04	503	0.1646D-04	504	0.1646D-04	505	0.1646D-04
506	0.1646D-04	507	0.1646D-04	508	0.1646D-04	509	0.1646D-04	510	0.1646D-04
511	0.1647D-04	512	0.1647D-04	513	0.1734D-04	514	0.1902D-04	515	0.1902D-04
516	0.1903D-04	517	0.1904D-04	518	0.1904D-04	519	0.1905D-04	520	0.1906D-04
521	0.1906D-04	522	0.1909D-04	523	0.1910D-04	524	0.1910D-04	525	0.1911D-04
526	0.1911D-04	527	0.1911D-04	528	0.1914D-04	529	0.1915D-04	530	0.1915D-04
531	0.1916D-04	532	0.1916D-04	533	0.1918D-04	534	0.1918D-04	535	0.1918D-04
536	0.1918D-04	537	0.1918D-04	538	0.1918D-04	539	0.1918D-04	540	0.1918D-04
541	0.1918D-04	542	0.1918D-04	543	0.1918D-04	544	0.1918D-04	545	0.1918D-04
546	0.1918D-04	547	0.1918D-04	548	0.1918D-04	549	0.2011D-04	550	0.2019D-04
551	0.2020D-04	552	0.2096D-04	553	0.2096D-04	554	0.2096D-04	555	0.2096D-04
556	0.2101D-04	557	0.2105D-04	558	0.2173D-04	559	0.2173D-04	560	0.2173D-04
561	0.2174D-04	562	0.2174D-04	563	0.2174D-04	564	0.2175D-04	565	0.2180D-04
566	0.2182D-04	567	0.2182D-04	568	0.2182D-04	569	0.2183D-04	570	0.2187D-04
571	0.2188D-04	572	0.2188D-04	573	0.2190D-04	574	0.2190D-04	575	0.2190D-04
576	0.2190D-04	577	0.2190D-04	578	0.2190D-04	579	0.2190D-04	580	0.2190D-04
581	0.2217D-04	582	0.2275D-04	583	0.2360D-04	584	0.2376D-04	585	0.2376D-04
586	0.2376D-04	587	0.2445D-04	588	0.2445D-04	589	0.2445D-04	590	0.2449D-04
591	0.[illegible]53D-04	592	0.2453D-04	593	0.2454D-04	594	0.2454D-04	595	0.2454D-04

Table 6 (Contd.)

596	0.2454D-04	597	0.2455D-04	598	0.2456D-04	599	0.2457D-04	600	0.2457D-04
601	0.2458D-04	602	0.2459D-04	603	0.2460D-04	604	0.2462D-04	605	0.2462D-04
606	0.2462D-04	607	0.2462D-04	608	0.2462D-04	609	0.2546D-04	610	0.2640D-04
611	0.2717D-04	612	0.2717D-04	613	0.2717D-04	614	0.2717D-04	615	0.2719D-04
616	0.2719D-04	617	0.2725D-04	618	0.2725D-04	619	0.2725D-04	620	0.2726D-04
621	0.2729D-04	622	0.2729D-04	623	0.2729D-04	624	0.2732D-04	625	0.2733D-04
626	0.2733D-04	627	0.2733D-04	628	0.2826D-04	629	0.2911D-04	630	0.2988D-04
631	0.2988D-04	632	0.2988D-04	633	0.2990D-04	634	0.2994D-04	635	0.2997D-04
636	0.2997D-04	637	0.2997D-04	638	0.2998D-04	639	0.2998D-04	640	0.2998D-04
641	0.3001D-04	642	0.3002D-04	643	0.3003D-04	644	0.3004D-04	645	0.3089D-04
646	0.3175D-04	647	0.3191D-04	648	0.3260D-04	649	0.3260D-04	650	0.3262D-04
651	0.3268D-04	652	0.3268D-04	653	0.3269D-04	654	0.3269D-04	655	0.3273D-04
656	0.3275D-04	657	0.3277D-04	658	0.3277D-04	659	0.3277D-04	660	0.3277D-04
661	0.3277D-04	662	0.3277D-04	663	0.3277D-04	664	0.3277D-04	665	0.3277D-04
666	0.3532D-04	667	0.3532D-04	668	0.3532D-04	669	0.3532D-04	670	0.3533D-04
671	0.3534D-04	672	0.3538D-04	673	0.3541D-04	674	0.3545D-04	675	0.3545D-04
676	0.3545D-04	677	0.3546D-04	678	0.3547D-04	679	0.3548D-04	680	0.3548D-04
681	0.3727D-04	682	0.3803D-04	683	0.3803D-04	684	0.3809D-04	685	0.3812D-04
686	0.3818D-04	687	0.3818D-04	688	0.3820D-04	689	0.3820D-04	690	0.3820D-04
691	0.3820D-04	692	0.4010D-04	693	0.4075D-04	694	0.4076D-04	695	0.4076D-04
696	0.4081D-04	697	0.4083D-04	698	0.4083D-04	699	0.4083D-04	700	0.4083D-04
701	0.4083D-04	702	0.4083D-04	703	0.4084D-04	704	0.4086D-04	705	0.4088D-04
706	0.4092D-04	707	0.4092D-04	708	0.4261D-04	709	0.4273D-04	710	0.4347D-04
711	0.4347D-04	712	0.4347D-04	713	0.4353D-04	714	0.4355D-04	715	0.4356D-04
716	0.4357D-04	717	0.4358D-04	718	0.4358D-04	719	0.4359D-04	720	0.4363D-04
721	0.4363D-04	722	0.4363D-04	723	0.4363D-04	724	0.4441D-04	725	0.4541D-04
726	0.4546D-04	727	0.4618D-04	728	0.4619D-04	729	0.4619D-04	730	0.4620D-04
731	0.4627D-04	732	0.4627D-04	733	0.4627D-04	734	0.4627D-04	735	0.4631D-04
736	0.4631D-04	737	0.4890D-04	738	0.4890D-04	739	0.4891D-04	740	0.4898D-04
741	0.4898D-04	742	0.4899D-04	743	0.4905D-04	744	0.4907D-04	745	0.5093D-04
746	0.5170D-04	747	0.5173D-04	748	0.5178D-04	749	0.5179D-04	750	0.5433D-04
751	0.5434D-04	752	0.5441D-04	753	0.5442D-04	754	0.5442D-04	755	0.5442D-04
756	0.5445D-04	757	0.5445D-04	758	0.5450D-04	759	0.5450D-04	760	0.5450D-04
761	0.5705D-04	762	0.5706D-04	763	0.5706D-04	764	0.5706D-04	765	0.5708D-04
766	0.5713D-04	767	0.5714D-04	768	0.5722D-04	769	0.5722D-04	770	0.5722D-04
771	0.5905D-04	772	0.5977D-04	773	0.5985D-04	774	0.5989D-04	775	0.5989D-04
776	0.5990D-04	777	0.6248D-04	778	0.6249D-04	779	0.6251D-04	780	0.6251D-04
781	0.6257D-04	782	0.6257D-04	783	0.6258D-04	784	0.6260D-04	785	0.6263D-04
786	0.6265D-04	787	0.6447D-04	788	0.6520D-04	789	0.6521D-04	790	0.6521D-04
791	0.6521D-04	792	0.6521D-04	793	0.6524D-04	794	0.6531D-04	795	0.6537D-04
796	0.6537D-04	797	0.6792D-04	798	0.6792D-04	799	0.6792D-04	800	0.6793D-04
801	0.6794D-04	802	0.6796D-04	803	0.7064D-04	804	0.7064D-04	805	0.7064D-04
806	0.7064D-04	807	0.7066D-04	808	0.7068D-04	809	0.7069D-04	810	0.7070D-04
811	0.7072D-04	812	0.7075D-04	813	0.7075D-04	814	0.7335D-04	815	0.7335D-04
816	0.7335D-04	817	0.7336D-04	818	0.73[illegible]D-04	819	0.7343D-04	820	0.7343D-04
821	0.7350D-04	822	0.7607D-04	823	0.7607D-04	824	0.7607D-04	825	0.7611D-04
826	0.7614D-04	827	0.7617D-04	828	0.7879D-04	829	0.7879D-04	830	0.7887D-04
831	0.8151D-04	832	0.8158D-04	833	0.8159D-04	834	0.8422D-04	835	0.8422D-04
836	0.8422D-04	837	0.8427D-04	838	0.8430D-04	839	0.8431D-04	840	0.8434D-04
841	0.8695D-04	842	0.8702D-04	843	0.8702D-04	844	0.8965D-04	845	0.8965D-04
846	0.8966D-04	847	0.8966D-04	848	0.9237D-04	849	0.9238D-04	850	0.9247D-04
851	0.9252D-04	852	0.9509D-04	853	0.9512D-04	854	0.9513D-04	855	0.9517D-04
856	0.9522D-04	857	0.9524D-04	858	0.9525D-04	859	0.9525D-04	860	0.9695D-04
861	0.9780D-04	862	0.9780D-04	863	0.9780D-04	864	0.9781D-04	865	0.9788D-04
866	0.9791D-04	867	0.1005D-03	868	0.1006D-03	869	0.1006D-03	870	0.1006D-03
871	0.1006D-03	872	0.1032D-03	873	0.1032D-03	874	0.1033D-03	875	0.1033D-03
876	0.1033D-03	877	0.1033D-03	878	0.1033D-03	879	0.1034D-03	880	0.1038D-03
881	0.1060D-03	882	0.1060D-03	883	0.1060D-03	884	0.1060D-03	885	0.1061D-03
886	0.1067D-03	887	0.1088D-03	888	0.1088D-03	889	0.1088D-03	890	0.1114D-03
891	0.1114D-03	892	0.1114D-03	893	0.1114D-03	894	0.1115D-03	895	0.1115D-03

Table 6 (Contd.)

896	0.1133D-03	897	0.1141D-03	898	0.1141D-03	899	0.1142D-03	900	0.1143D-03
901	0.1168D-03	902	0.1168D-03	903	0.1168D-03	904	0.1169D-03	905	0.1169D-03
906	0.1170D-03	907	0.1195D-03	908	0.1196D-03	909	0.1196D-03	910	0.1196D-03
911	0.1223D-03	912	0.1251D-03	913	0.1251D-03	914	0.1268D-03	915	0.1269D-03
916	0.1277D-03	917	0.1278D-03	918	0.1296D-03	919	0.1305D-03	920	0.1332D-03
921	0.1333D-03	922	0.1358D-03	923	0.1358D-03	924	0.1358D-03	925	0.1359D-03
926	0.1359D-03	927	0.1370D-03	928	0.1396D-03	929	0.1396D-03	930	0.1413D-03
931	0.1414D-03	932	0.1467D-03	933	0.1467D-03	934	0.1522D-03	935	0.1522D-03
936	0.1540D-03	937	0.1549D-03	938	0.1576D-03	939	0.1576D-03	940	0.1577D-03
941	0.1603D-03	942	0.1659D-03	943	0.1685D-03	944	0.1685D-03	945	0.1712D-03
946	0.1712D-03	947	0.1739D-03	948	0.1739D-03	949	0.1820D-03	950	0.1820D-03
951	0.1820D-03	952	0.1820D-03	953	0.1822D-03	954	0.1847D-03	955	0.1876D-03
956	0.1903D-03	957	0.1956D-03	958	0.1983D-03	959	0.1983D-03	960	0.2010D-03
961	0.2066D-03	962	0.2066D-03	963	0.2120D-03	964	0.2173D-03	965	0.2201D-03
966	0.2229D-03	967	0.2229D-03	968	0.2229D-03	969	0.2283D-03	970	0.2365D-03
971	0.2391D-03	972	0.2391D-03	973	0.2392D-03	974	0.2445D-03	975	0.2473D-03
976	0.2473D-03	977	0.2554D-03	978	0.2554D-03	979	0.2555D-03	980	0.2663D-03
981	0.2663D-03	982	0.2799D-03	983	0.2954D-03	984	0.2988D-03	985	0.3206D-03
986	0.3613D-03	987	0.3912D-03	988	0.4075D-03	989	0.4076D-03	990	0.4265D-03
991	0.4265D-03	992	0.4320D-03	993	0.4646D-03	994	0.4809D-03	995	0.4864D-03
996	0.4917D-03	997	0.5515D-03	998	0.5516D-03	999	0.7934D-03	1000	0.1182D-02

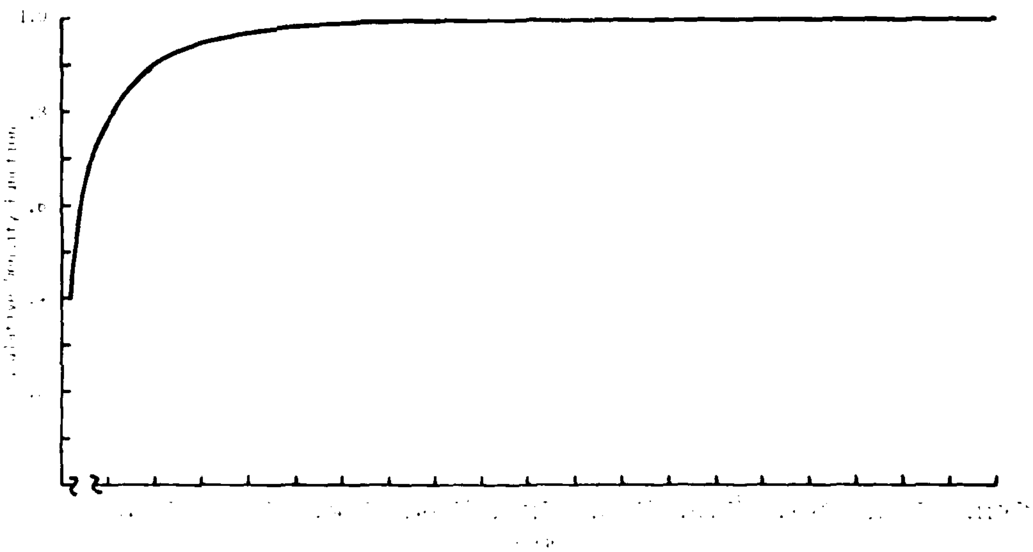


Figure 24 - Cumulative Density Function of Damage Distribution

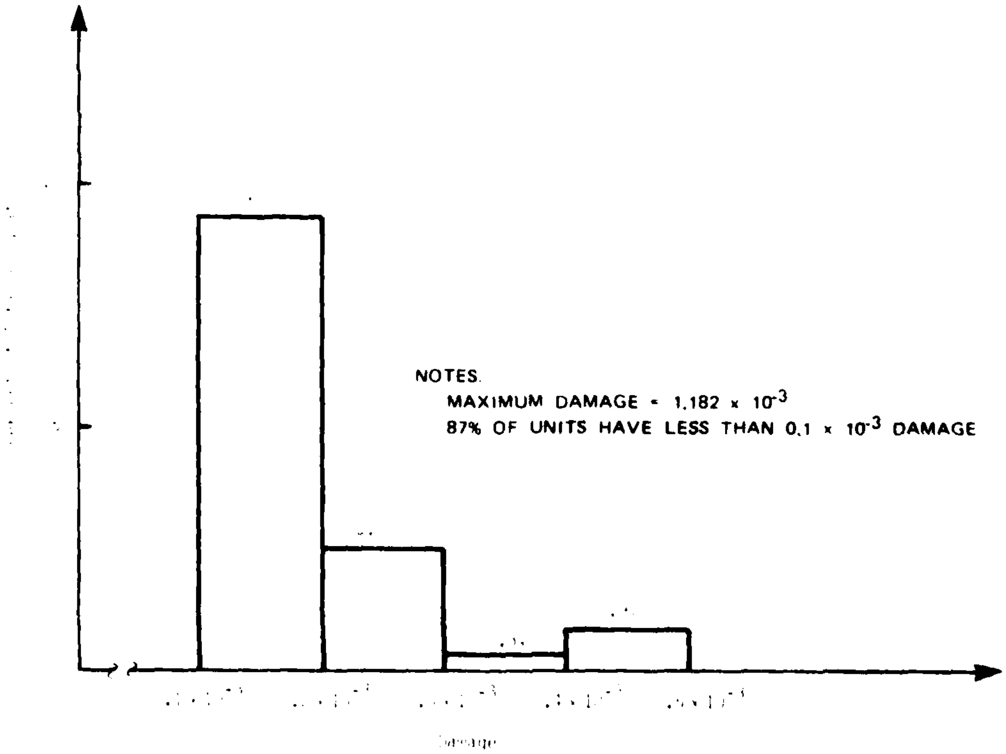


Figure 25 - Distribution of Extreme Damage in 10 Years.

CONCLUSIONS AND RECOMMENDATIONS

In this report a methodology for estimating the time to failure of rockets by using probabilistic analysis has been described and illustrated. In particular, a probabilistic model of the environment to which rockets are exposed in real life situations was developed. As an example, the Sidewinder rocket motors were considered, and the realistic environmental and logistic (i.e., movement of rockets from one location to another) data were probabilistically modeled. The methods used are general and can be applied to other rockets. Several computer programs were written in the course of this project with the final aim being the development of a probabilistic environmental model that predicts the time to failure of rockets. Failure occurs when the rocket has exceeded an allowable level of cumulative damage. In this report the damage was caused by thermal stresses due to external temperature only. There are other causes besides temperature for damage in rockets (such as shock and vibration, chemical aging of the propellant, humidity, and radiation). It is recommended that these other effects be added into the damage calculation.

The Sidewinder damage calculations indicate the need for an extensive analysis of captive flights, which is the single most damaging environment. In spite of the relatively short portion of its life that a rocket spends in captive flight, the damage incurred is more than 100 times the damage that occurs in storage.

For Sidewinder rockets, the Navy storage locations are situated in mild climates. The most damaging location was Tokyo/Atsugi. The damage during storage is minimal. Rockets aboard ships also experience a mild temperature. The surrounding sea acts as a temperature stabilizer and ships do not frequently travel in very cold and icy waters where damage could be higher. Truck and train transportation and air transportation are more damaging than storage but (at least two orders of magnitude) less damaging than captive flight. In captive flight the rocket is directly exposed to the surroundings, and the air temperature at high altitude can be very cold and hence cause a larger amount of damage because the damage rate is large.

These results suggest the need for a follow-on effort for a more extensive analysis of captive flights. For example, careful measurements of rocket skin and propellant temperatures during captive flights are suggested (there are presently a limited number of reports describing temperature variations during captive flights [NWC TP 5365, Part 1 and Part 2][14]). The measured temperatures then can be used to compare with computed transient temperatures through the propellant cross section. Finally, coupling this with a viscoelastic analysis program (possibly Jeter's code "Travis"[22]) allows determination of stresses. From this data, the damage during captive flight can be established as a function of several important parameters such as (1) flight altitude, (2) time to achieve this altitude, (3) flight speed, (4) position of rocket on the plane (there can be differences in skin temperatures of rockets depending on where they are placed on the plane), (5) descent time. This information can be computerized, and by describing the characteristics of every captive flight (i.e., the flight altitude, rise time, etc.), the relative damage for each rocket in a fleet could be monitored. The rockets which accumulate most significant levels of damage should be expended, discarded or used subsequently in "less damaging" flights. The ultimate objective of this proposed scheme is to maximize combat readiness by reducing rocket failures and increasing the overall life expectancy.

An activity should be initiated to establish the location within the propellant most susceptible to damage from fluctuations in the external temperature. The analyses in this report are based on the assumption that the bore is the critical location. Although it is known that this is an important location for occurrence of maximum stresses under some conditions, the assumption that this is the most critical location from the standpoint of damage is not proven. Neither have we shown that "flaw", i.e., a crack, is equivalent to end of useful life.

Installation of the computer codes developed during the course of this project on a China Lake Naval Weapons Center computer system is also planned. These programs can be used for other rockets with different dimensions and properties by simply modifying the input data. The environmental data and logistics (movement of rockets) can be determined for the specific rocket by using these codes in a similar manner as was done for Sidewinder.

REFERENCES

1. Bills, K.W., Jr., and Wiegand, J.H., "The Application of Integrated Structural Analysis to the Prediction of Reliability," Proceedings of the Ninth Reliability and Maintainability Conference, Detroit, Michigan, 20-22 July 1970. (UNCLASSIFIED.)

2. Cost, T.L., and Dagen, J.D., "Probabilistic Service Life Prediction of Solid Rocket Motors Subjected to Thermal Loads Using Computer Simulation," JANNAF Structures and Mechanical Behavior Subcommittee Meeting, CPIA Publication No. 283, February 1977. (UNCLASSIFIED.)

3. Heller, R.A., Thermal Stress: A Narrow Band Random Load in a Solid Propellant Cylinder. U.S. Army Missile Command. May 1976. (Technical Report RK-7T-1, UNCLASSIFIED.)

4. Cost, T.L., "Service Life Predictions for Thermal Loads Using Viscoelastic Finite Element Simulation Models. U.S. Army Missile Research and Development Command, Redstone Arsenal, Alabama, December 1978. (Technical Report T-CR-79-3, UNCLASSIFIED.)

5. Cost, T.L., "Long Range Service Life Prediction for Thermal Cycling," JANNAF Structures and Mechanical Behavior Subcommittee Meeting, Huntsville, Alabama, December 1980. (UNCLASSIFIED.)

6. Cost, T.L., and Weeks, G.E., Probabilistic Methods in Solid Rocket Motor Structural Integrity Analysis." Athena Engineering Company, December 1980. (Technical Report No. RK-CR-80-9, UNCLASSIFIED.)

7. Lee, E.H., "Viscoelasticity," in Handbook of Engineering Mechanics, W. Flugge, Ed., (Chapter 53, General Survey), McGraw-Hill Book Company, New York, 1962.

8. Heller, R.A., "Thermal Stress as a Narrow-Band Random Load," ASCE Journal of the Engineering Mechanics Division, EM5, October 1976.

9. Naval Weapons Center, Environmental Criteria Determination for Air-Launched Tactical Propulsion Systems. Part 1: Stockpile-to-Target Sequence, by H. C. Schafer. China Lake, California, NWC, July 1968. (NWC TP 4464, Part 1, publication UNCLASSIFIED.)

10. Shooman, M. L., Probabilistic Reliability, An Engineering Approach, New York, McGraw-Hill Book Company, 1968.

11. Forsythe, G.E., "RANDK, Random Number Generator," Stanford University, Department of Computer Science, Stanford University Center for Information Technology Computer Program, 1969. (UNCLASSIFIED.)

12. Naval Ordnance Test Station. Storage Temperature of Explosive Hazard Magazines. Part 2: Western Pacific, by I. S. Kurotori and H. C. Schafer. China Lake, CA, NOTS, July 1967. (NOTS TP 4143, Pt. 2, publication UNCLASSIFIED.)

13. Naval Weapons Center. Thermal Exposure of Ammunition on Board Ship. Part 1: Cruisers and Large Destroyers; Part 2: Aircraft Carriers, by S. Matsuda and H. C. Schafer. China Lake, CA, NWC, March 1970, October 1979. (NWC TP 4824, Parts 1 and 2, publications UNCLASSIFIED.)

14. Naval Weapons Center. Measurement of Missile Thermal Response During Captive Flight at High Altitudes," Parts 1 and 2 by H. C. Schafer. China Lake, CA, NWC, March 1973. (NWC TP 5365, publication UNCLASSIFIED.)

15. Naval Weapons Center. Temperature Profiles of Air Transported Material, by H. C. Schafer and R. A. Dickus. China Lake, CA, NWC, October 1970. (NWC TP 4828, publication UNCLASSIFIED.)

16. Naval Weapons Center. Temperature Profiles of Truck Transported Ordnance, by Billy D. Martin and Howard Schafer. China Lake, CA, NWC, June 1970. (NWC TP 4822, publication UNCLASSIFIED.)

17. Carslaw, H.S., and Jaeger, J.C., "Conduction of Heat in Solids," Second Ed., Oxford Press, 1959.

18. Williams, J.L., Blatz, P.J., and Shapery, R., "Fundamental Studies Relating to Systems Analysis of Rocket Propellants," California Institute of Technology, pp.165-166, 1961. (GALCIT, SM61-5, UNCLASSIFIED.)

19. Shapery, R., "A Method of Viscoelastic Stress Analysis Using Elastic Solutions, "Journal of Franklin Institute, 179, No. 4, April 1965.

20. Lee, E.H., and Rogers, T.G., "Solution of Viscoelastic Stress Analysis Problems Using Measured Creep of Relaxation Functions," Journal of Applied Mechanics, 30, pp. 127-133, March 1963.

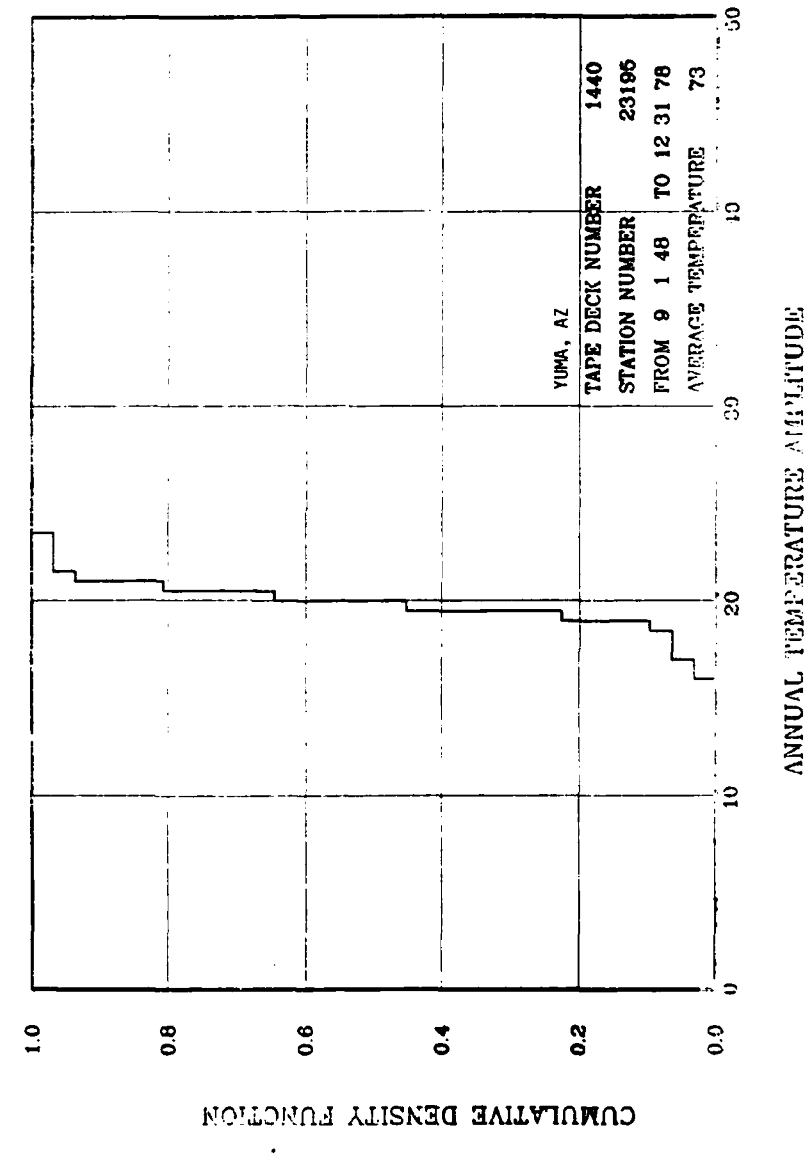
21. Bischel, K.G., and Wiegand, J.H., "Life Prediction of Solid Rocket Propellant Grains in Storage from Tensile and SEC Data," February 1977. (UNCLASSIFIED.)

22. E. L. Jeter. A Thermoviscoelastic Study of a Solid Rocket Propellant Grain. Thesis submitted to Stanford University. March 1970. (Publication UNCLASSIFIED.)

23. Abramowitz, M., and Stegun, I.A., "Handbook of Mathematical Functions," National Bureau of Standards, Applied Mathematics Series, 55, pp. 379-389, Dover 1968.

APPENDIX A

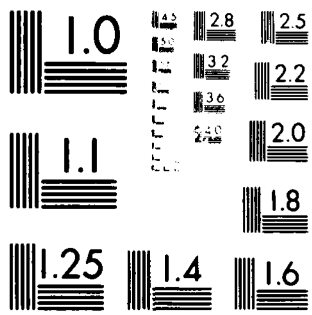
Cumulative Density Functions of Storage Location
Temperature Amplitudes


YUMA, AZ
TAPE DECK NUMBER 1440
STATION NUMBER 23195
FROM 9 1 48 TO 12 31 78
AVERAGE TEMPERATURE 73
ANNUAL TEMPERATURE AMPLITUDE
CUMULATIVE DENSITY FUNCTION
0.0
0.2
0.4
0.6
0.8
1.0
0
10
20
30
40
50

AD-A117 651   FAILURE ANALYSIS ASSOCIATES PALO ALTO CA   F/G 21/8.2
PROBABILISTIC ENVIRONMENTAL MODEL FOR SOLID ROCKET MOTOR LIFE P--ETC(U)
MAR 82   G DERBALIAN, J M THOMAS, P JOHNSTON   N60530-78-C-0127

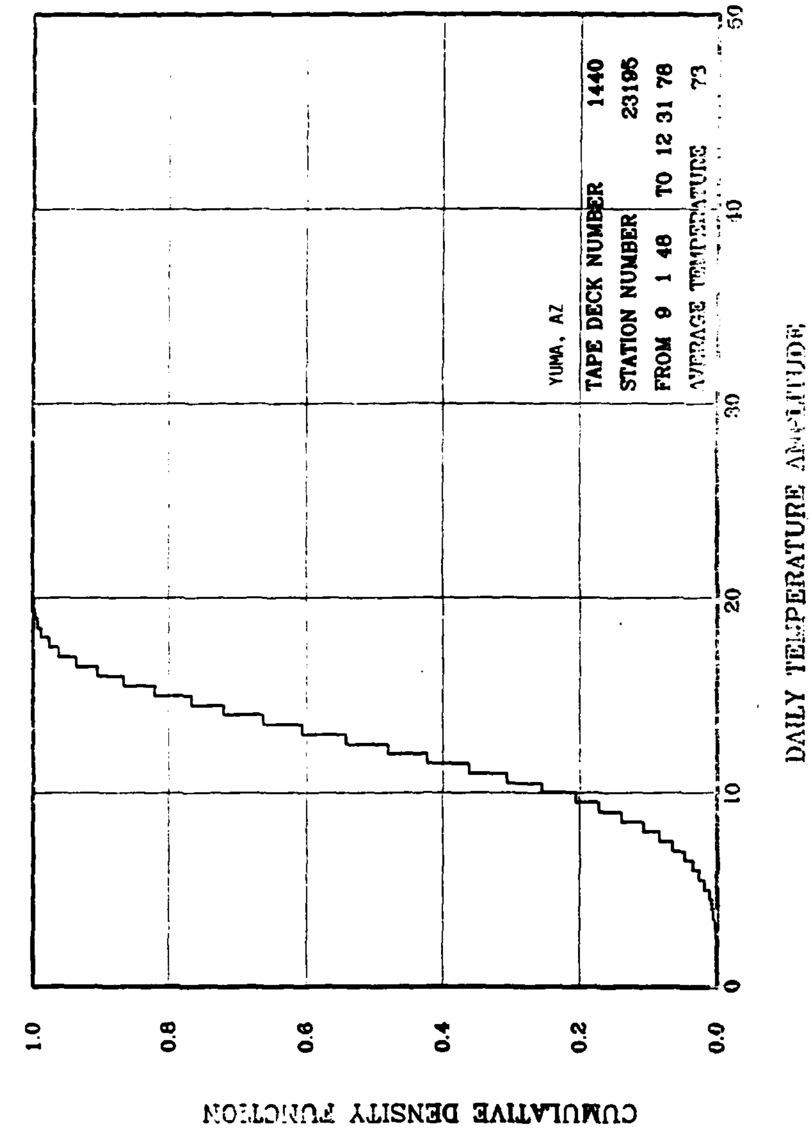
UNCLASSIFIED   NWC-TP-6305   NL

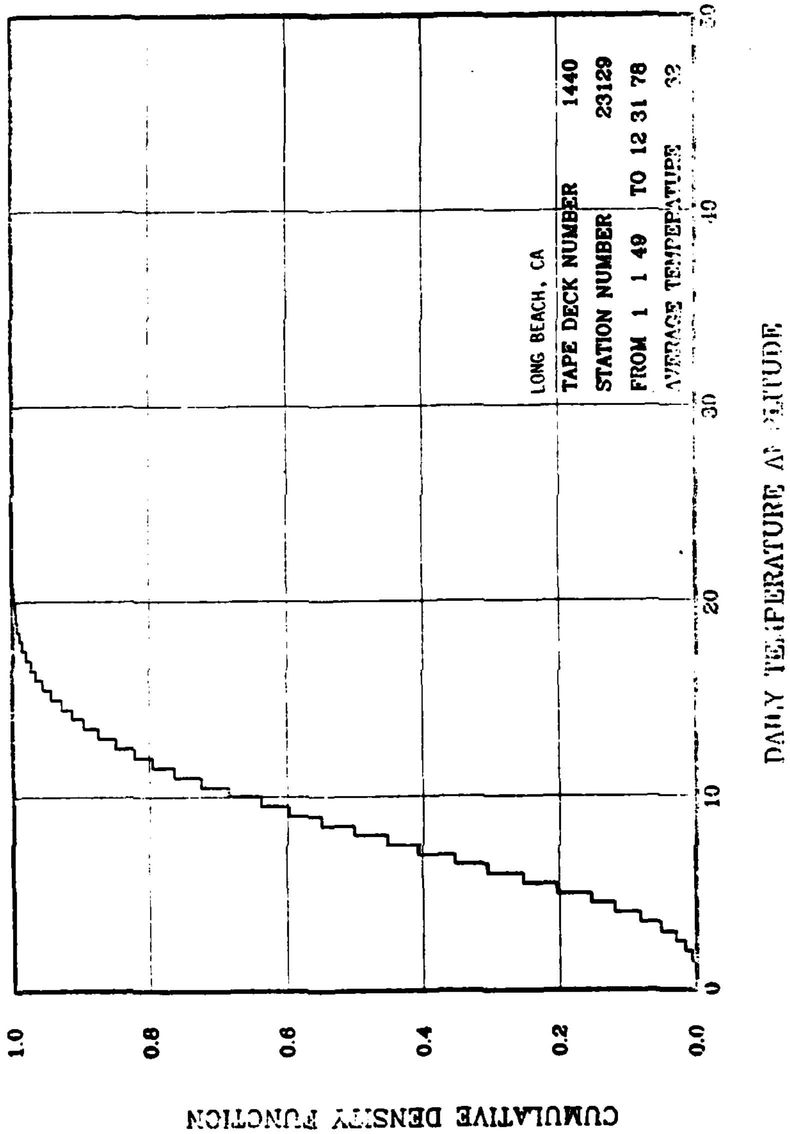
2 of 3

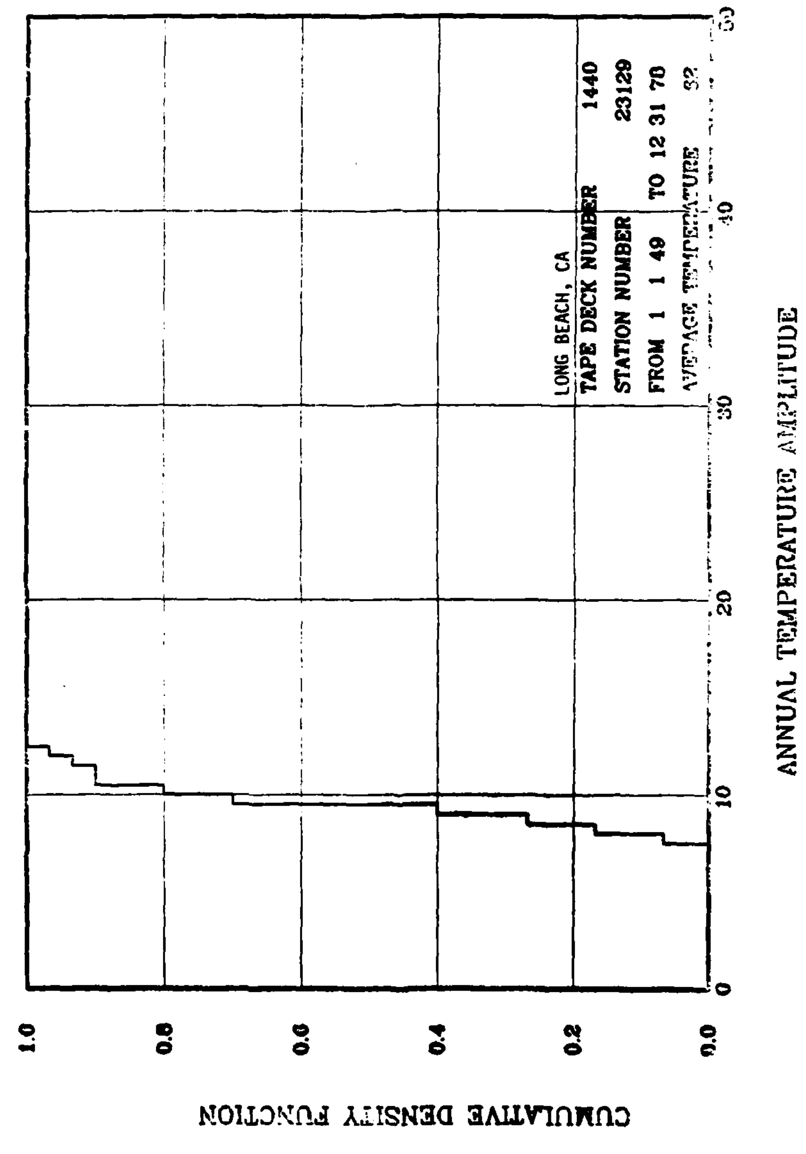


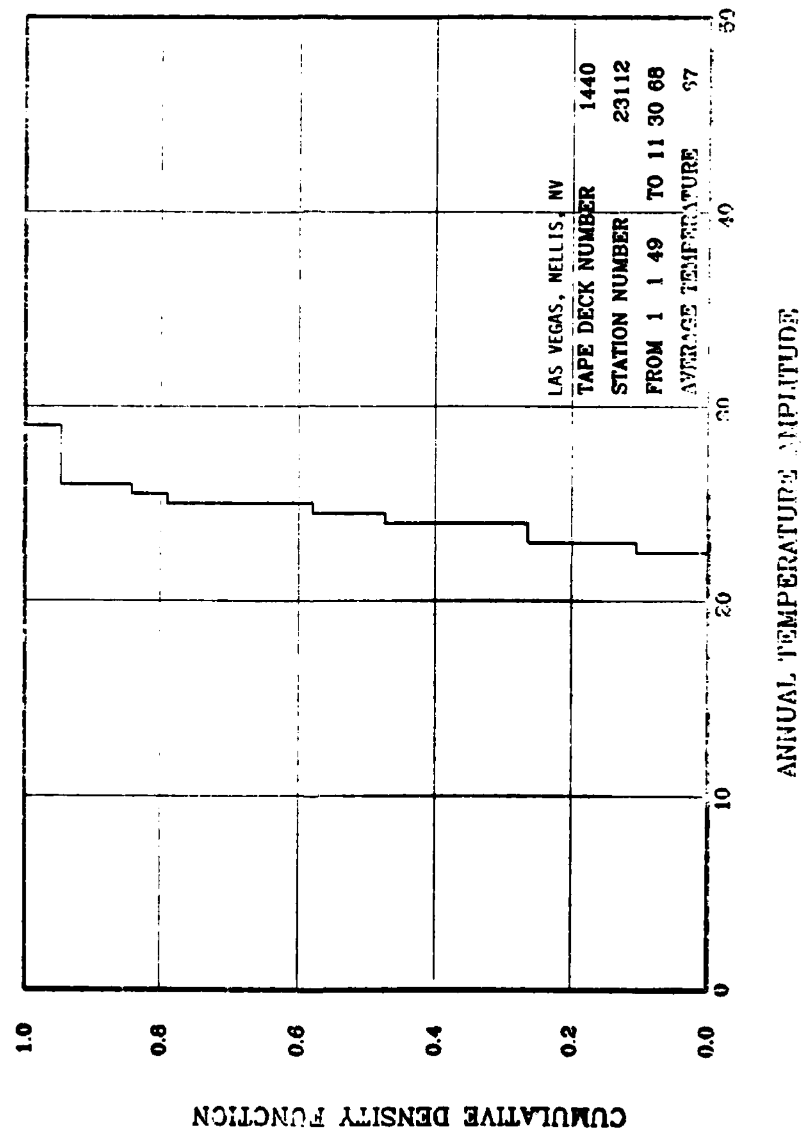
MICROCOPY RESOLUTION TEST CHART

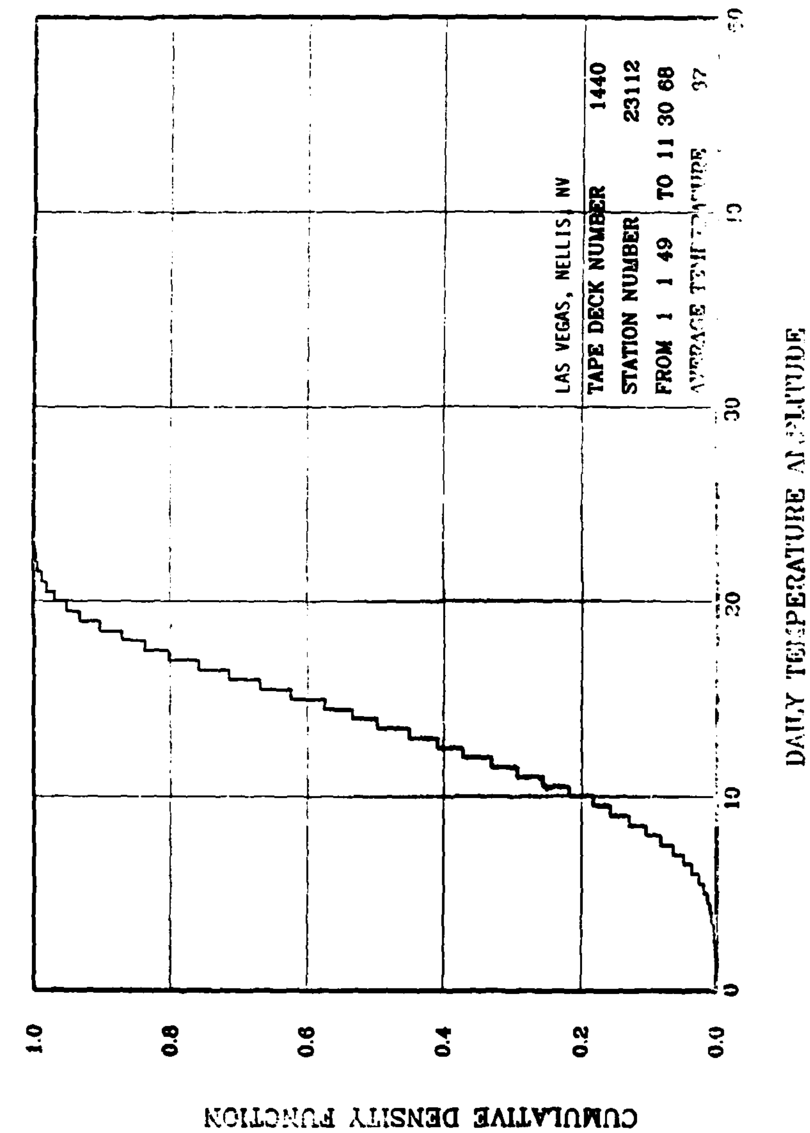
NATIONAL BUREAU OF STANDARDS 1963-A

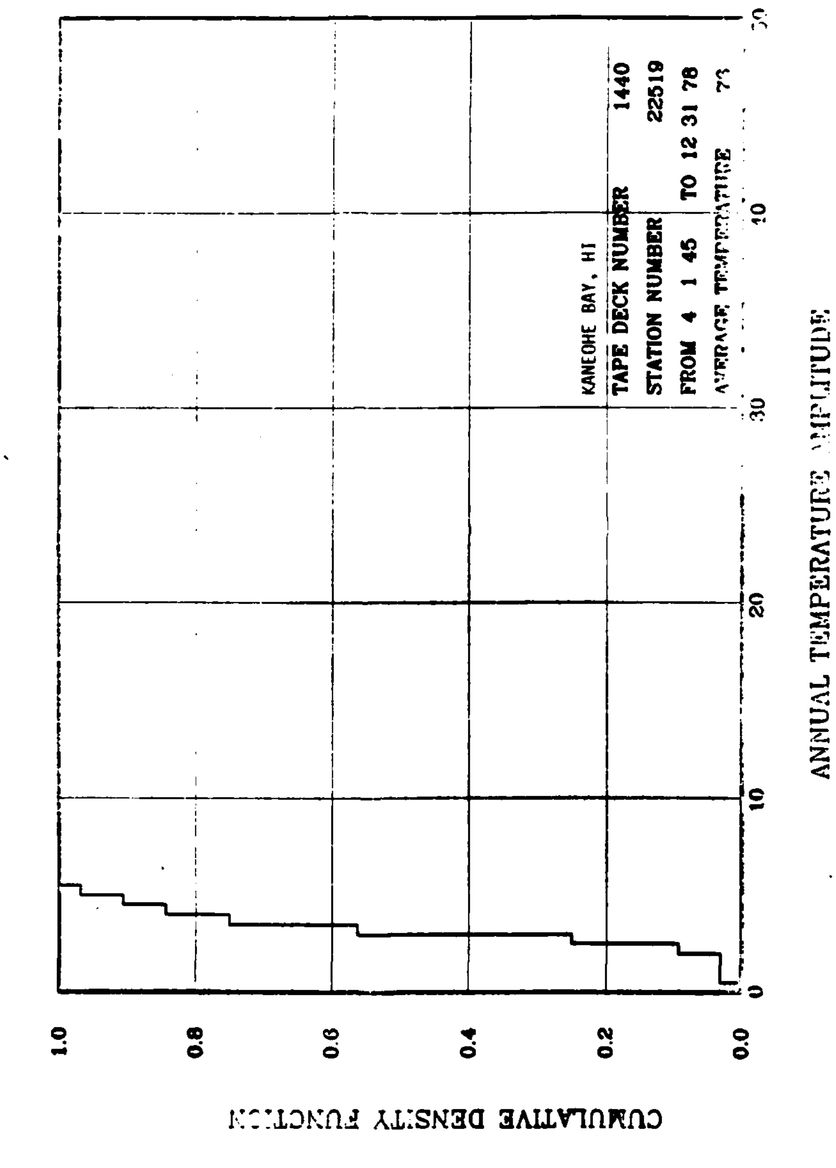

YUMA, AZ
TAPE DECK NUMBER 1440
STATION NUMBER 23195
FROM 9 1 48 TO 12 31 78
AVERAGE TEMPERATURE 73
DAILY TEMPERATURE AMPLITUDE
CUMULATIVE DENSITY FUNCTION
0
10
20
30
40
50
0.0
0.2
0.4
0.6
0.8
1.0

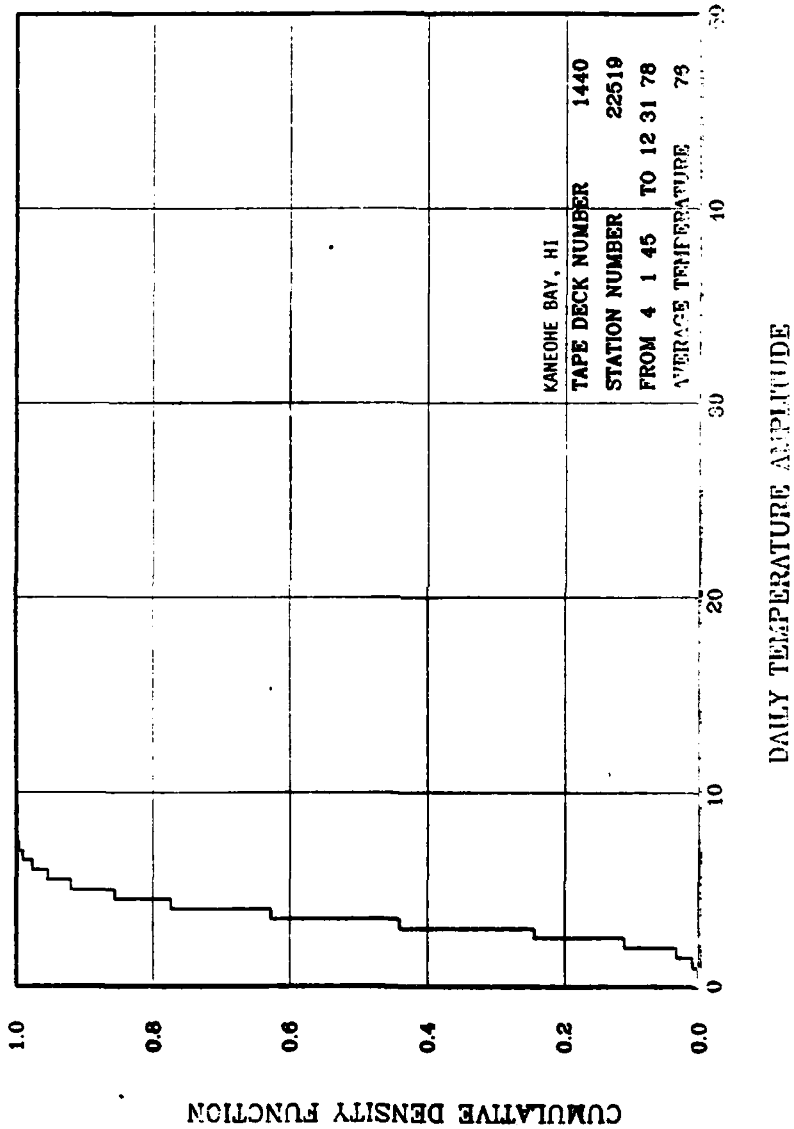

LONG BEACH, CA
TAPE DECK NUMBER 1440
STATION NUMBER 23129
FROM 1 1 49 TO 12 31 78
CUMULATIVE DENSITY FUNCTION
1.0
0.8
0.6
0.4
0.2
0.0
0
10
20
30

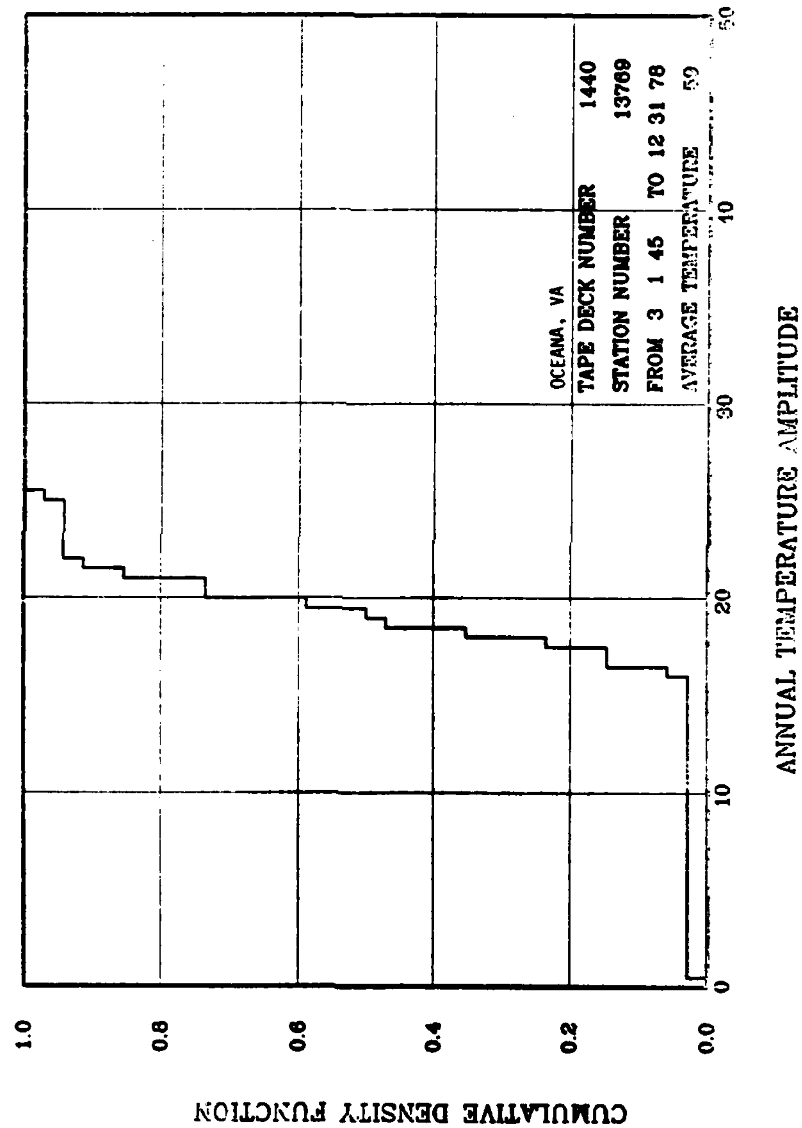

LONG BEACH, CA
TAPE DECK NUMBER 1440
STATION NUMBER 23129
FROM 1 1 49 TO 12 31 78
ANNUAL TEMPERATURE AMPLITUDE
CUMULATIVE DENSITY FUNCTION
0
10
20
30
40
0.0
0.2
0.4
0.8
1.0

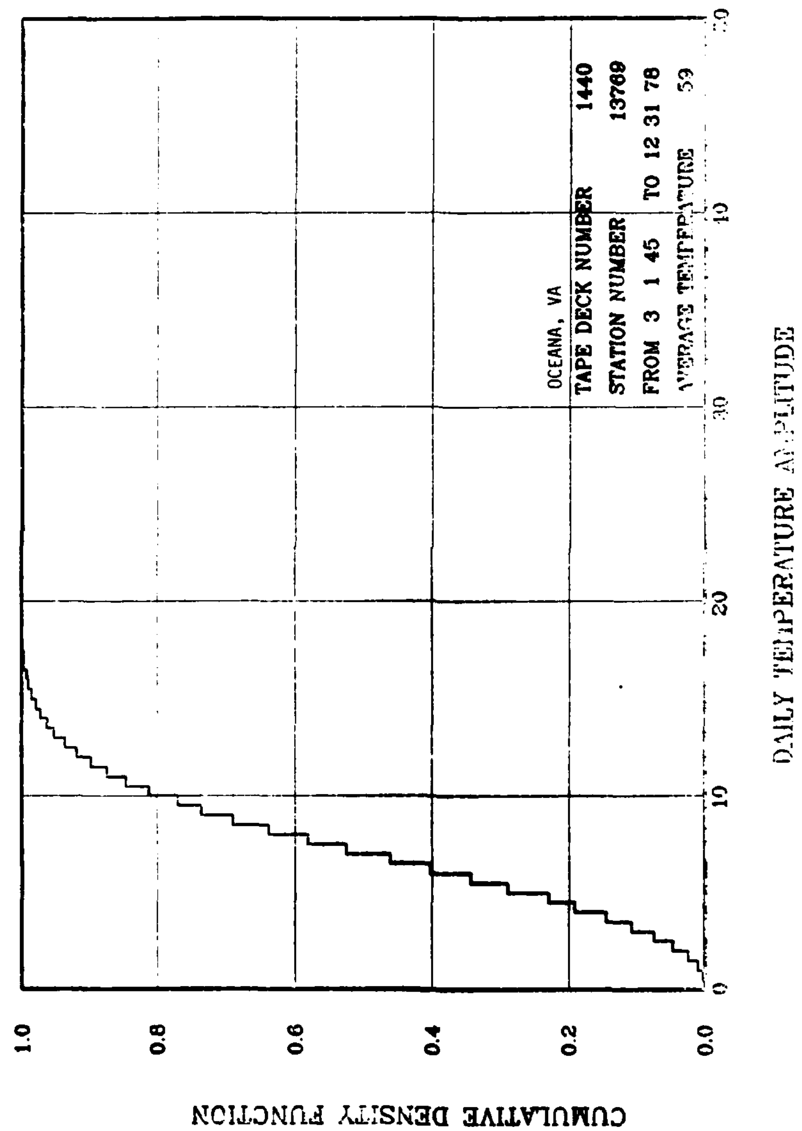

LAS VEGAS, NELLIS, NV
TAPE DECK NUMBER 1440
STATION NUMBER 23112
FROM 1 1 49 TO 11 30 68
AVERAGE TEMPERATURE 97
ANNUAL TEMPERATURE AMPLITUDE
CUMULATIVE DENSITY FUNCTION
0
10
20
30
40
50
0.0
0.2
0.4
0.6
0.8
1.0

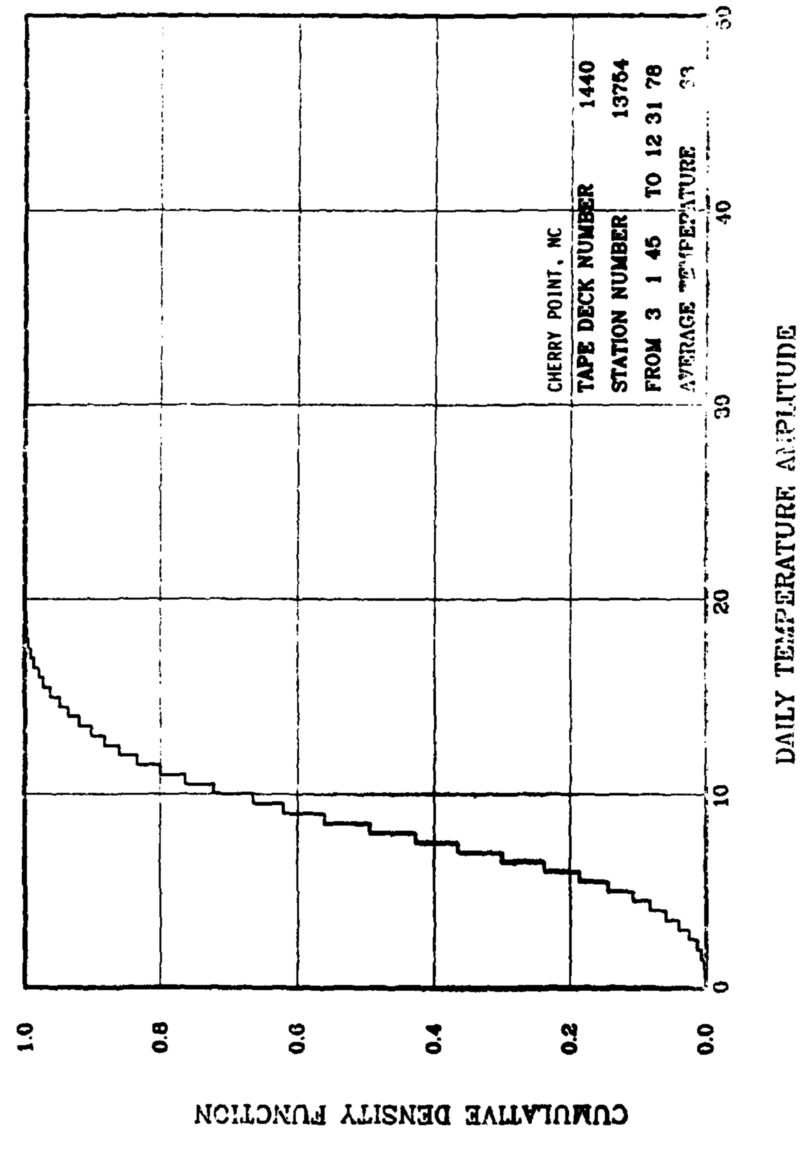

LAS VEGAS, NELLIS, NV
TAPE DECK NUMBER 1440
STATION NUMBER 23112
FROM 1 1 49 TO 11 30 68
CUMULATIVE DENSITY FUNCTION
1.0
0.8
0.6
0.4
0.2
0.0
0
10
20
30

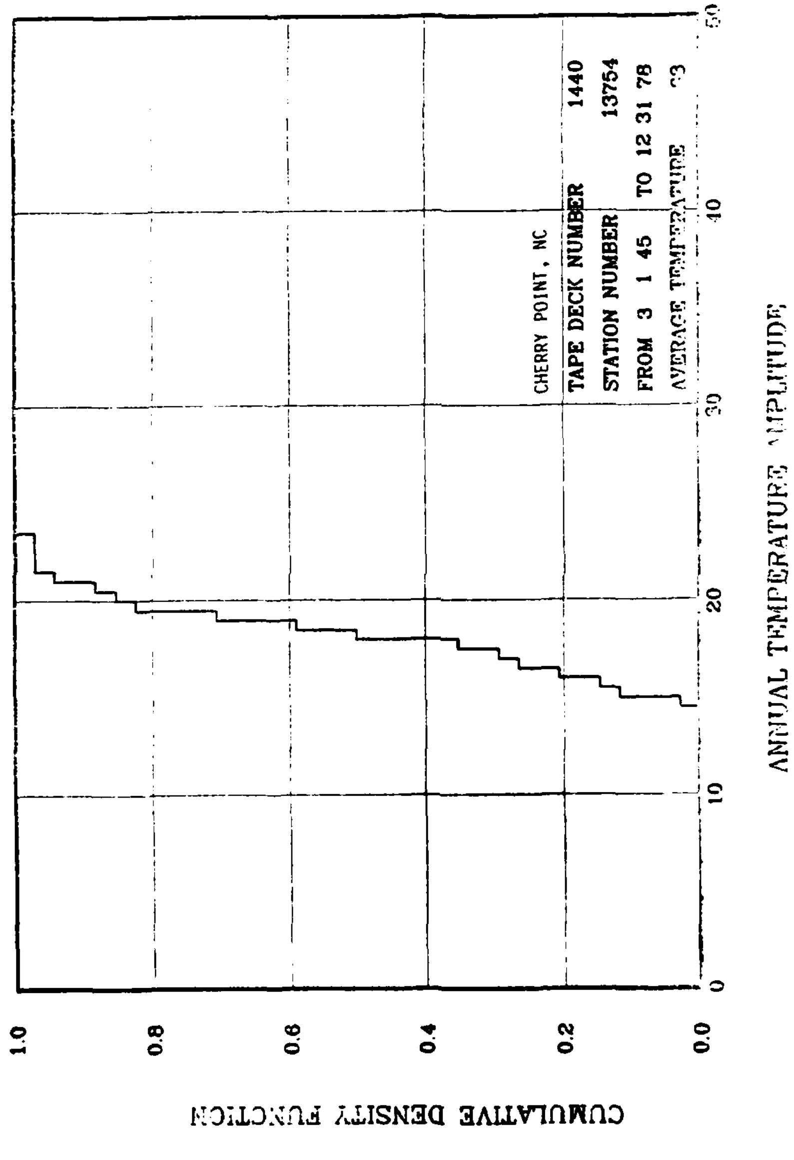

KANEOHE BAY, HI
TAPE DECK NUMBER 1440
STATION NUMBER 22519
FROM 4 1 45 TO 12 31 78
AVERAGE TEMPERATURE
0
10
20
30
40
ANNUAL TEMPERATURE AMPLITUDE
0.0
0.2
0.4
0.6
0.8
1.0
CUMULATIVE DENSITY FUNCTION

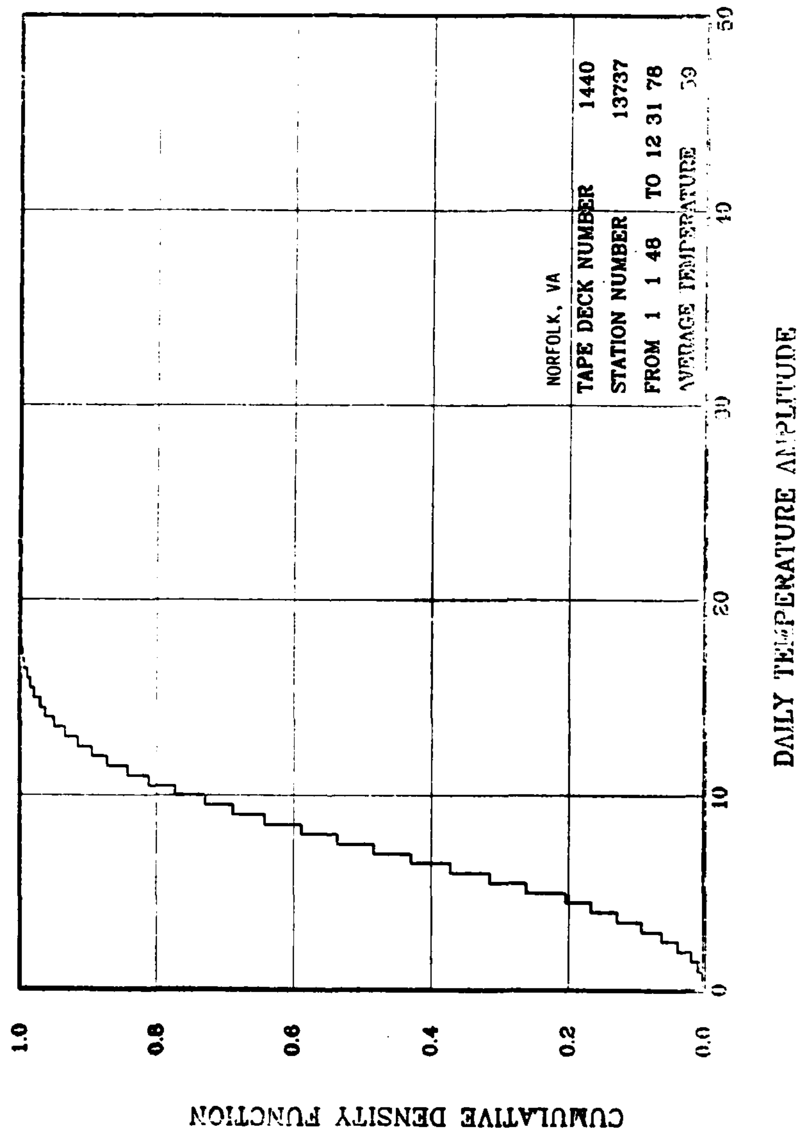

KANEOHE BAY, HI
TAPE DECK NUMBER 1440
STATION NUMBER 22519
FROM 4 1 45 TO 12 31 78
AVERAGE TEMPERATURE 73
DAILY TEMPERATURE AMPLITUDE
CUMULATIVE DENSITY FUNCTION
0
10
20
30
1.0
0.8
0.6
0.4
0.2
0.0

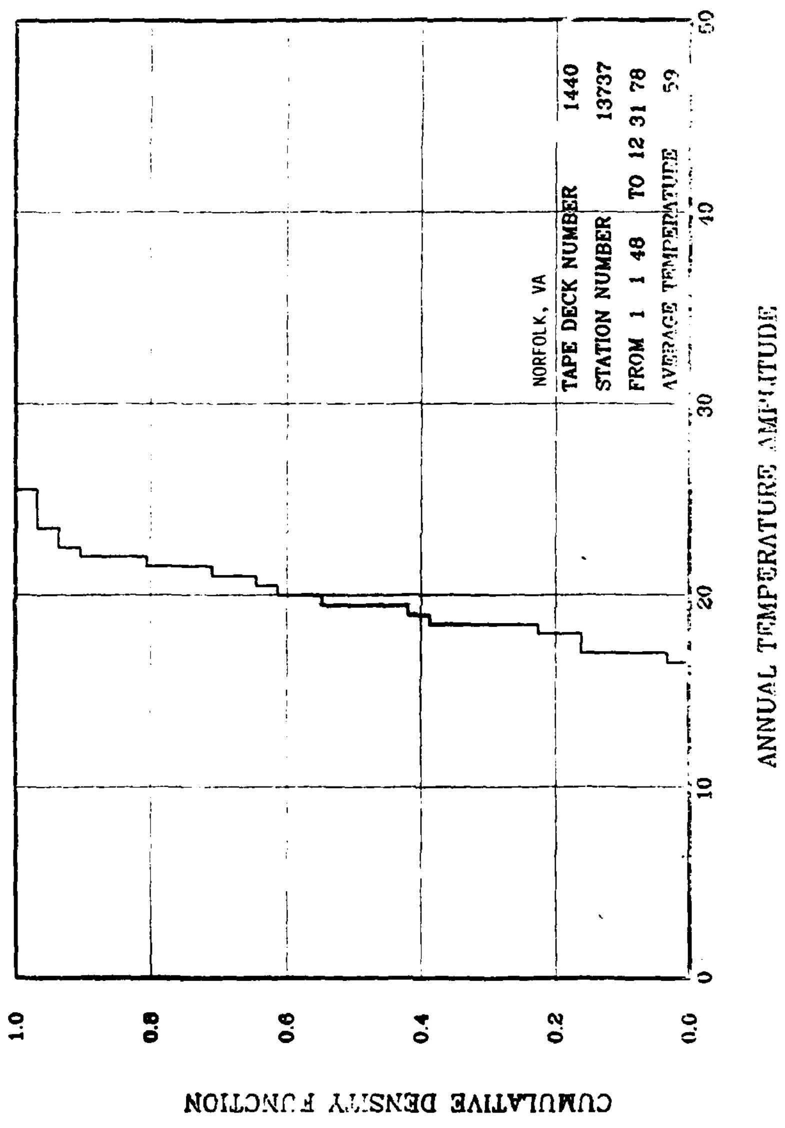

OCEANA, VA
TAPE DECK NUMBER 1440
STATION NUMBER 13769
FROM 3 1 45 TO 12 31 78
AVERAGE TEMPERATURE 59
ANNUAL TEMPERATURE AMPLITUDE
CUMULATIVE DENSITY FUNCTION
0.0
0.2
0.4
0.6
0.8
1.0
0
10
20
30
50

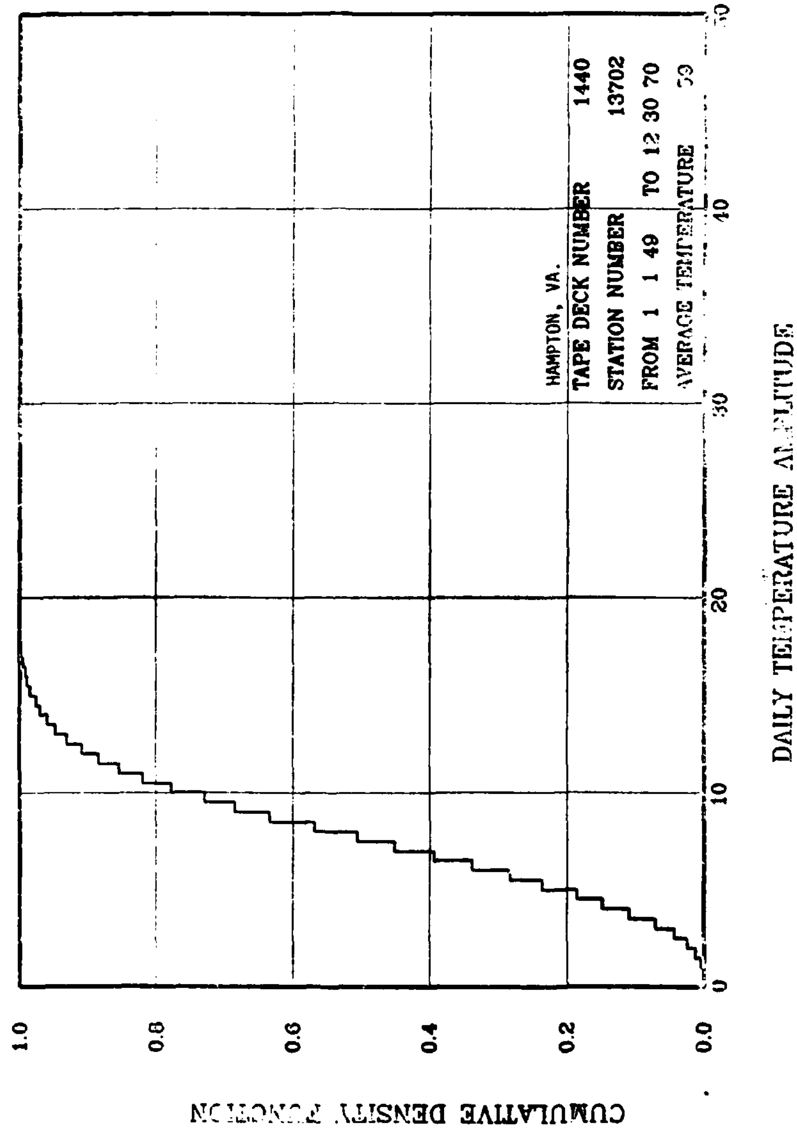

OCEANA, VA
TAPE DECK NUMBER 1440
STATION NUMBER 13769
FROM 3 1 45 TO 12 31 78
AVERAGE TEMPERATURE 59
DAILY TEMPERATURE AMPLITUDE
CUMULATIVE DENSITY FUNCTION
0.0
0.2
0.4
0.6
0.8
1.0
0
10
20
30
40
50

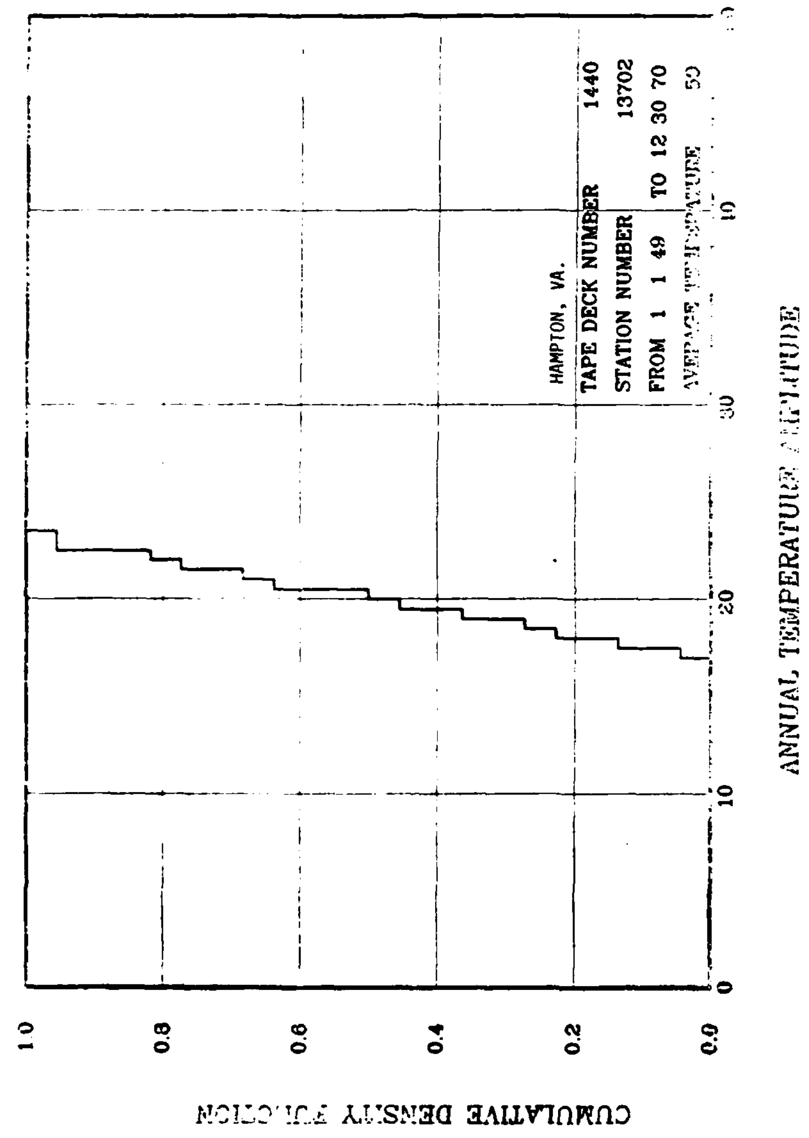

CHERRY POINT, NC
TAPE DECK NUMBER 1440
STATION NUMBER 13754
FROM 3 1 45 TO 12 31 78
AVERAGE TEMPERATURE
CUMULATIVE DENSITY FUNCTION
1.0
0.8
0.6
0.4
0.2
0.0
DAILY TEMPERATURE AMPLITUDE
0
10
20
30
40
50

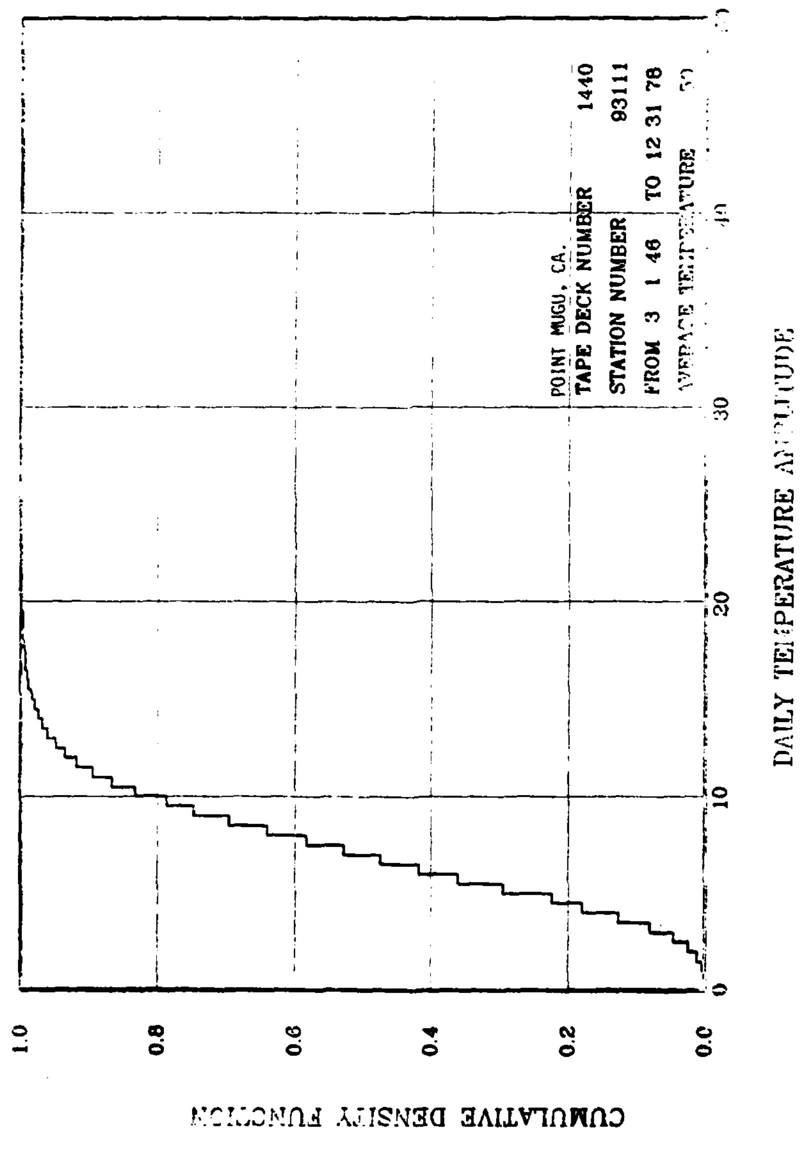

CHERRY POINT, NC
TAPE DECK NUMBER 1440
STATION NUMBER 13754
FROM 3 1 45 TO 12 31 78
ANNUAL TEMPERATURE AMPLITUDE
CUMULATIVE DENSITY FUNCTION
0
10
20
30
40
50
0.0
0.2
0.4
0.6
0.8
1.0

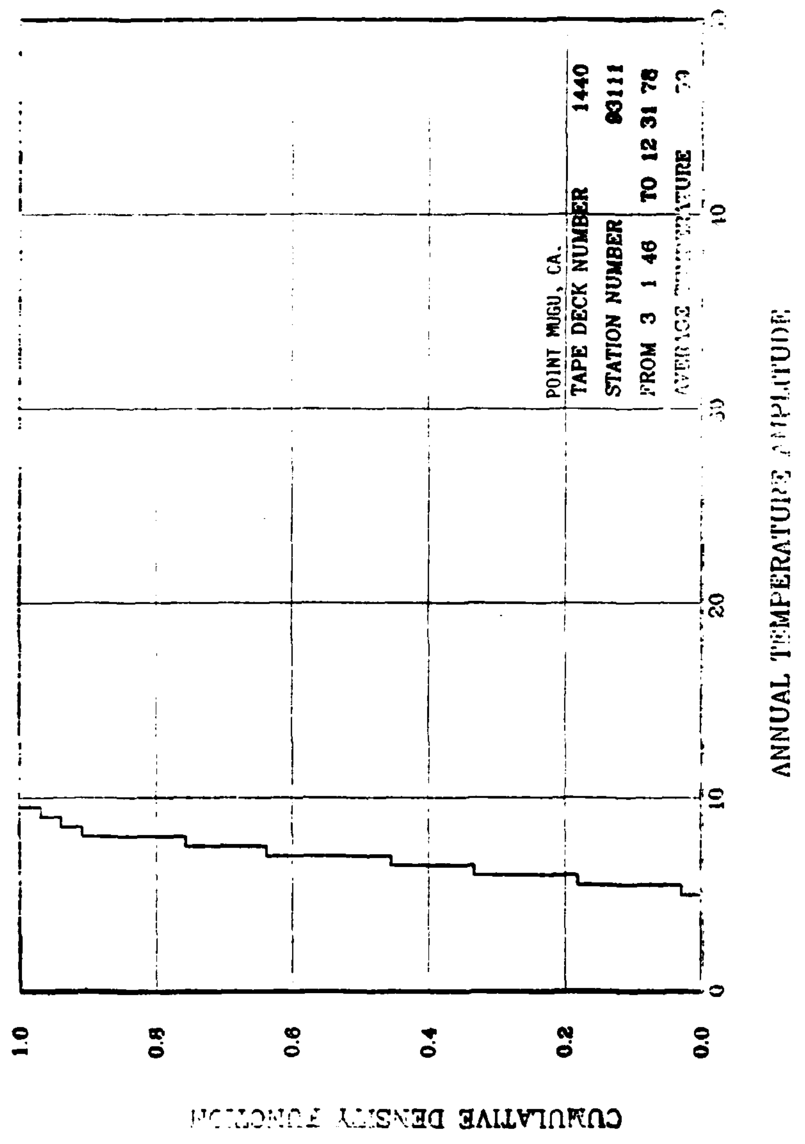

NORFOLK, VA
TAPE DECK NUMBER 1440
STATION NUMBER 13737
FROM 1 1 48 TO 12 31 78
AVERAGE TEMPERATURE
DAILY TEMPERATURE AMPLITUDE
CUMULATIVE DENSITY FUNCTION
0
10
20
30
40
50
0.0
0.2
0.4
0.6
0.8
1.0

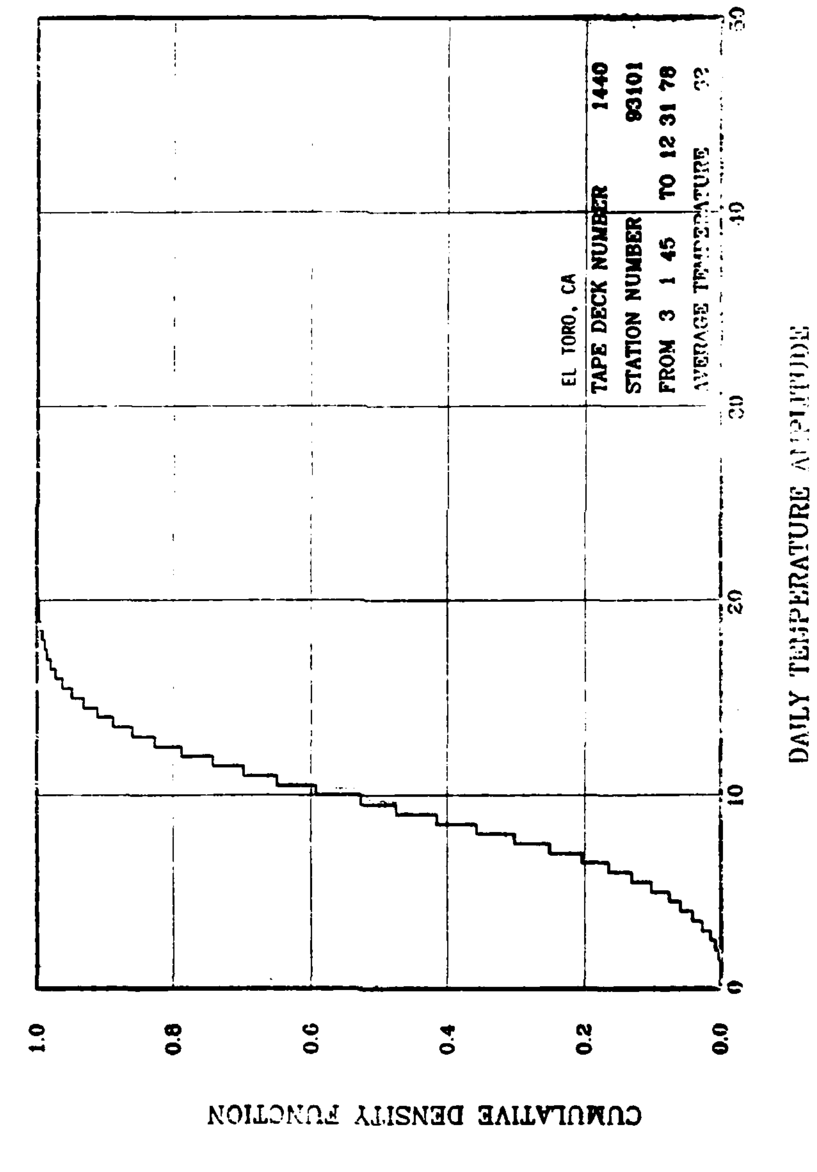

NORFOLK, VA
TAPE DECK NUMBER 1440
STATION NUMBER 13737
FROM 1 1 48 TO 12 31 78
AVERAGE TEMPERATURE 59
0
10
20
30
40
50
ANNUAL TEMPERATURE AMPLITUDE
0.0
0.2
0.4
0.6
0.8
1.0
CUMULATIVE DENSITY FUNCTION

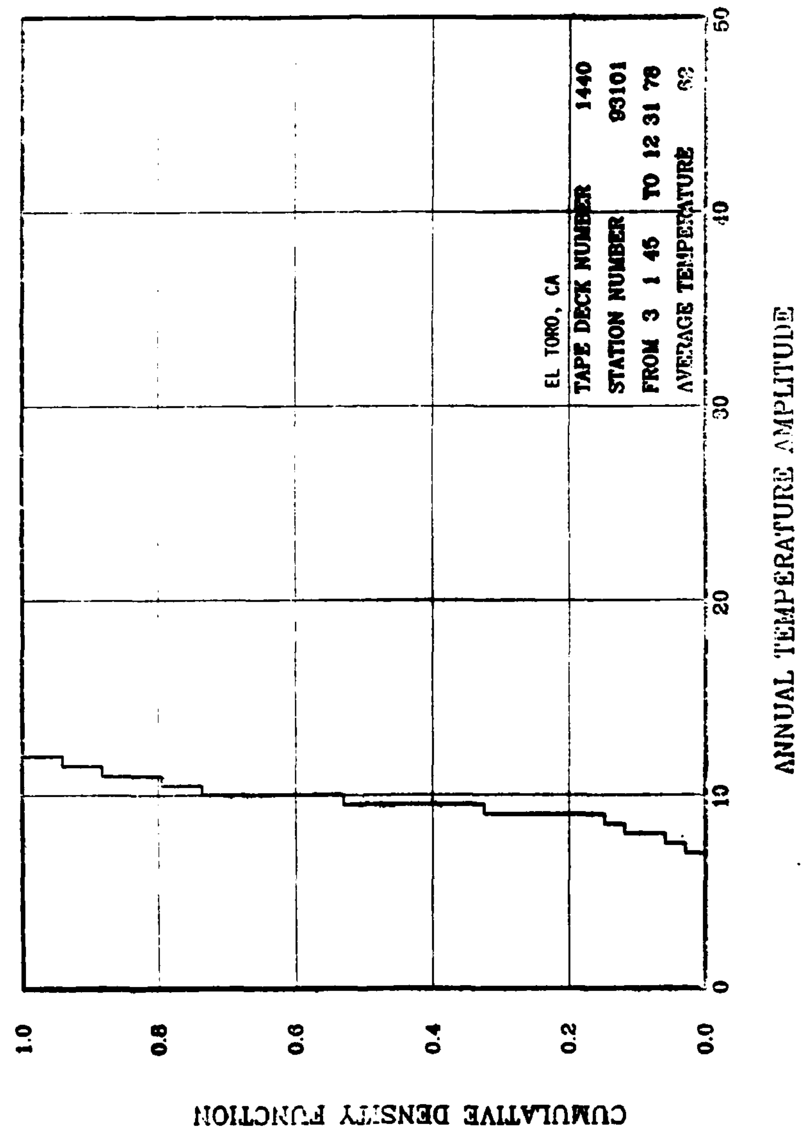

HAMPTON, VA.
TAPE DECK NUMBER 1440
STATION NUMBER 13702
FROM 1 1 49 TO 12 30 70
AVERAGE TEMPERATURE
0
10
20
30
40
1.0
0.8
0.6
0.4
0.2
0.0
CUMULATIVE DENSITY FUNCTION
DAILY TEMPERATURE AMPLITUDE

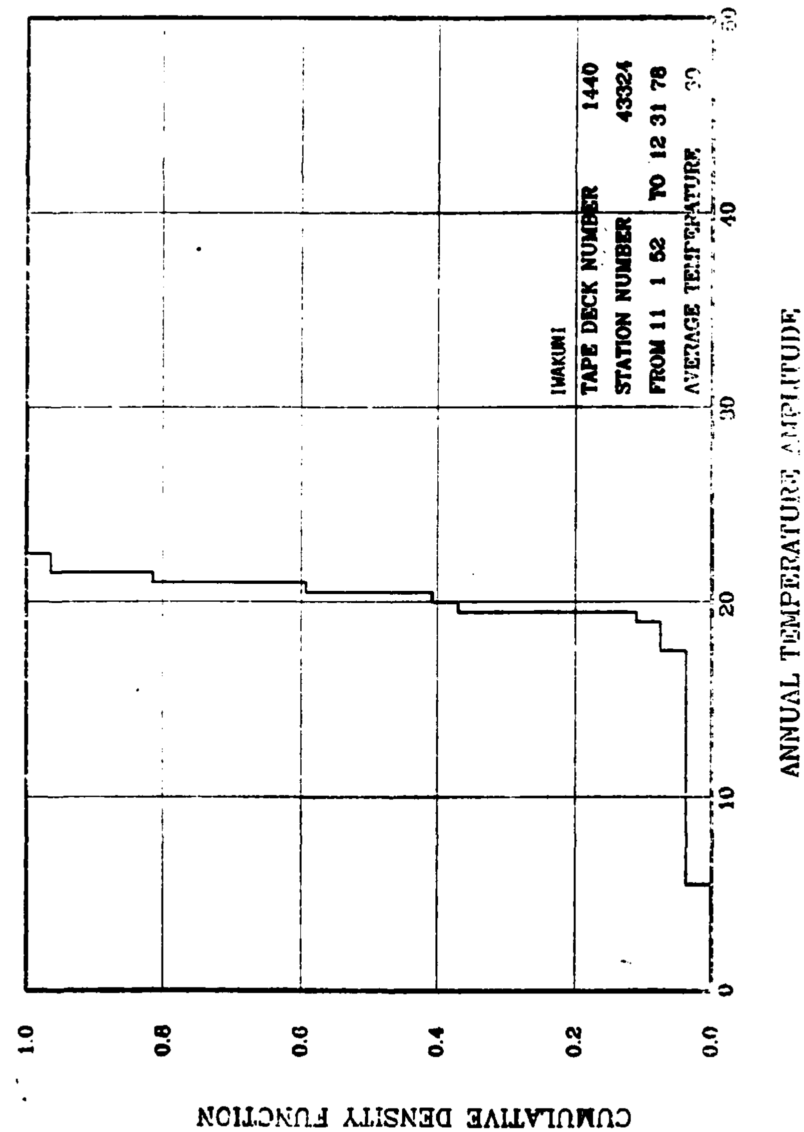

HAMPTON, VA.
TAPE DECK NUMBER 1440
STATION NUMBER 13702
FROM 1 1 49 TO 12 30 70
0
10
20
30
40
ANNUAL TEMPERATURE
1.0
0.8
0.6
0.4
0.2
0.0
CUMULATIVE DENSITY

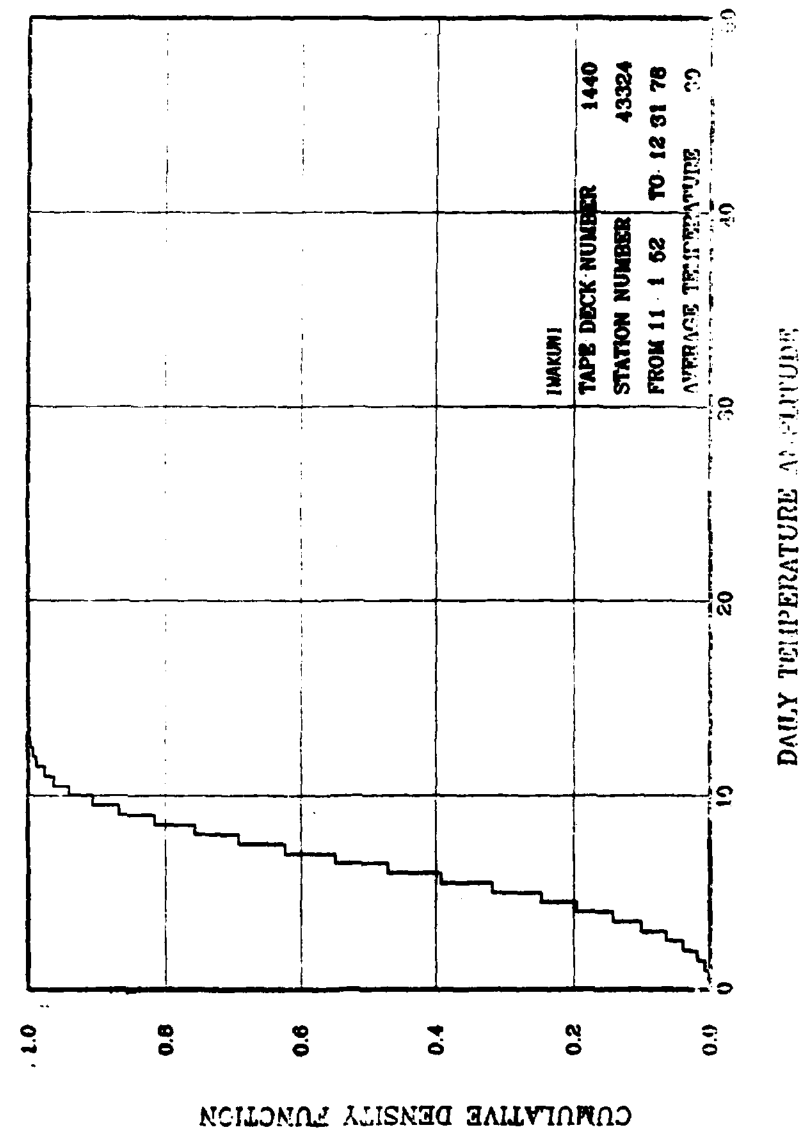

POINT MUGU, CA.
TAPE DECK NUMBER 1440
STATION NUMBER 93111
FROM 3 1 46 TO 12 31 78
DAILY TEMPERATURE AMPLITUDE
CUMULATIVE DENSITY FUNCTION
0
10
20
30
40
0.0
0.2
0.4
0.6
0.8
1.0

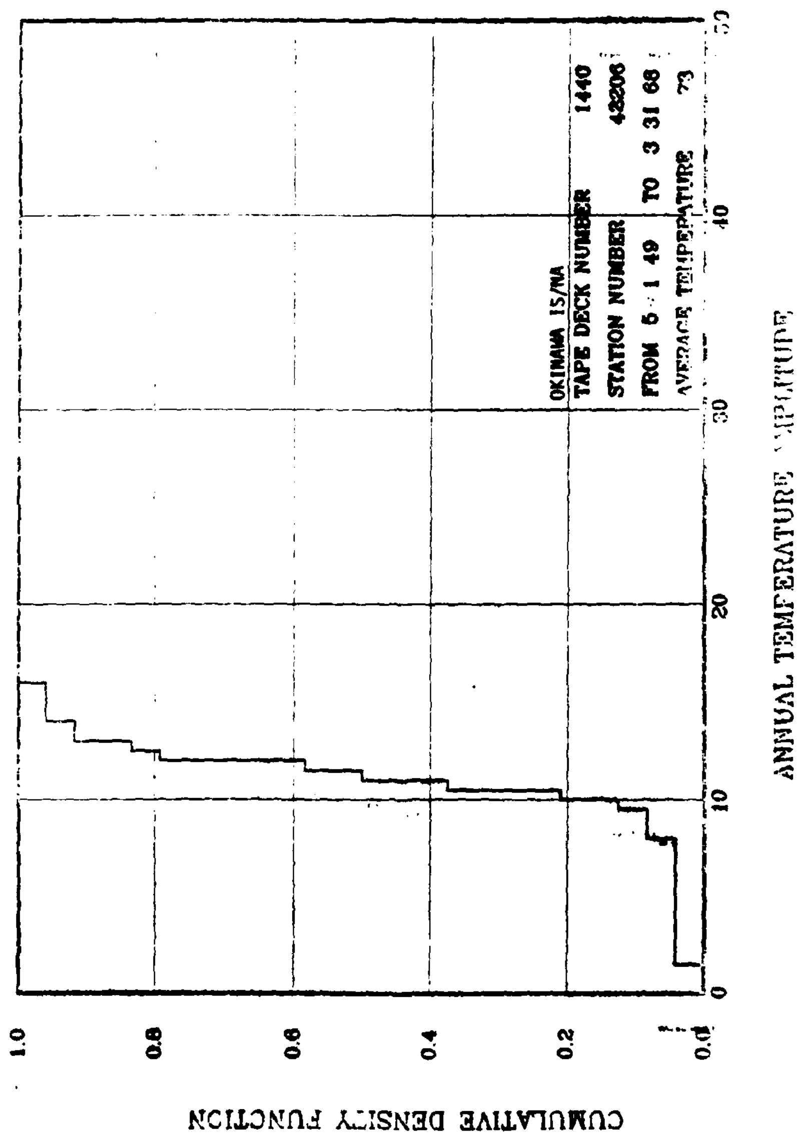

POINT MUGU, CA.
TAPE DECK NUMBER 1440
STATION NUMBER 93111
FROM 3 1 46 TO 12 31 78
ANNUAL TEMPERATURE AMPLITUDE
CUMULATIVE DENSITY FUNCTION
1.0
0.8
0.6
0.4
0.2
0.0
0
10
20

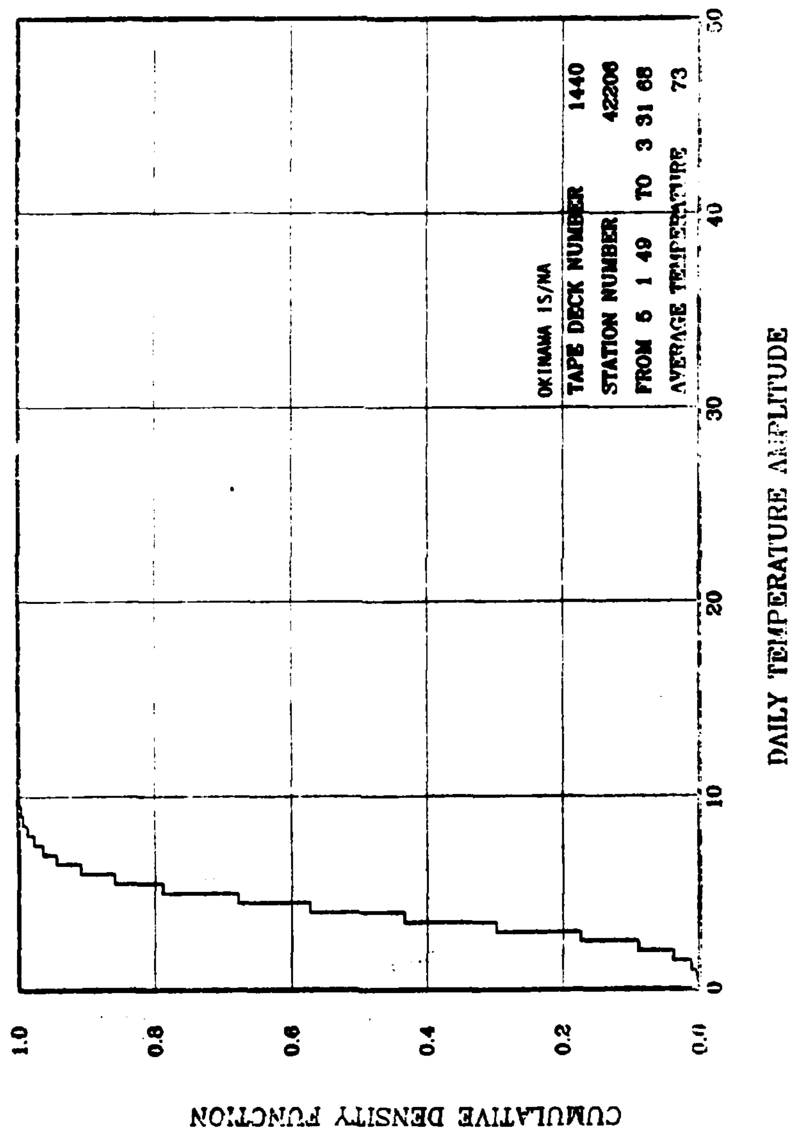

EL TORO, CA
TAPE DECK NUMBER 1440
STATION NUMBER 93101
FROM 3 1 45 TO 12 31 78
AVERAGE TEMPERATURE
DAILY TEMPERATURE AMPLITUDE
CUMULATIVE DENSITY FUNCTION
0.0
0.2
0.4
0.6
0.8
1.0
0
10
20
30
40
50

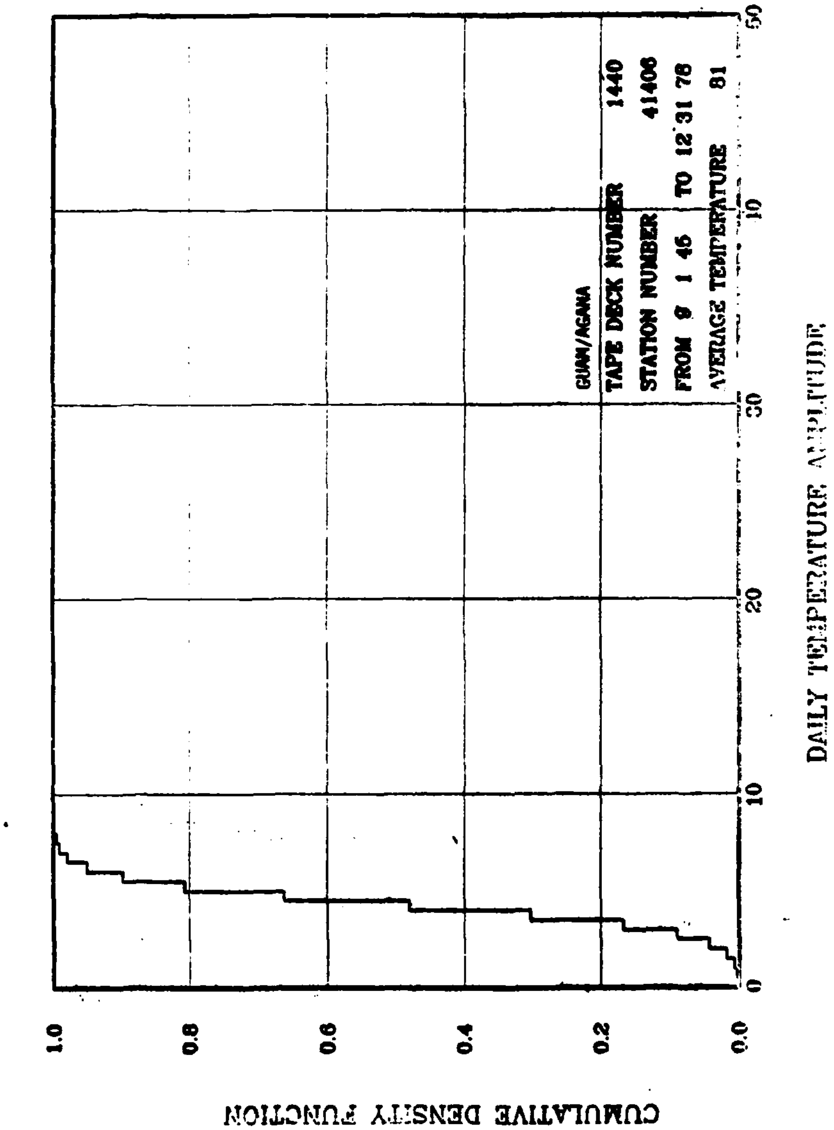

EL TORO, CA
TAPE DECK NUMBER 1440
STATION NUMBER 93101
FROM 3 1 45 TO 12 31 78
AVERAGE TEMPERATURE
ANNUAL TEMPERATURE AMPLITUDE
CUMULATIVE DENSITY FUNCTION
0
10
20
30
40
50
0.0
0.2
0.4
0.6
0.8
1.0

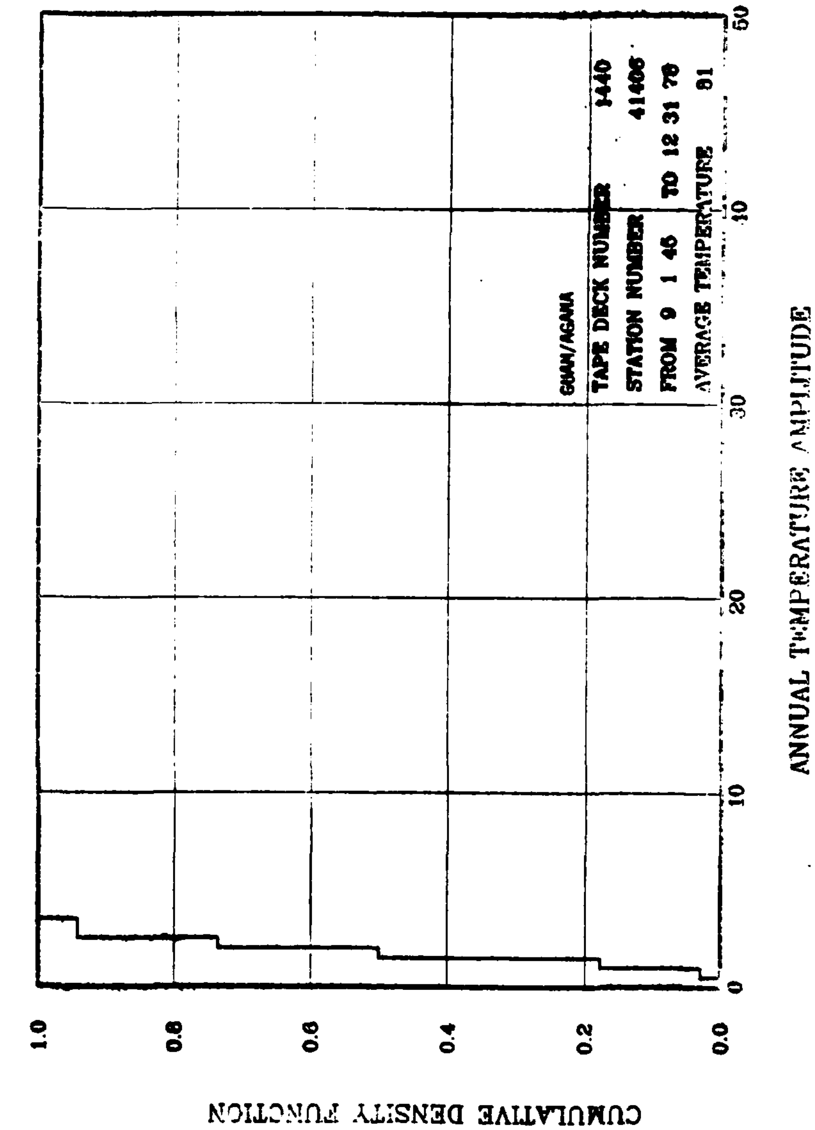

IWAKUNI
TAPE DECK NUMBER 1440
STATION NUMBER 43324
FROM 11 1 52 TO 12 31 78
AVERAGE TEMPERATURE
CUMULATIVE DENSITY FUNCTION
0.0
0.2
0.4
0.6
0.8
1.0
ANNUAL TEMPERATURE AMPLITUDE
0
10
20
30
40
50

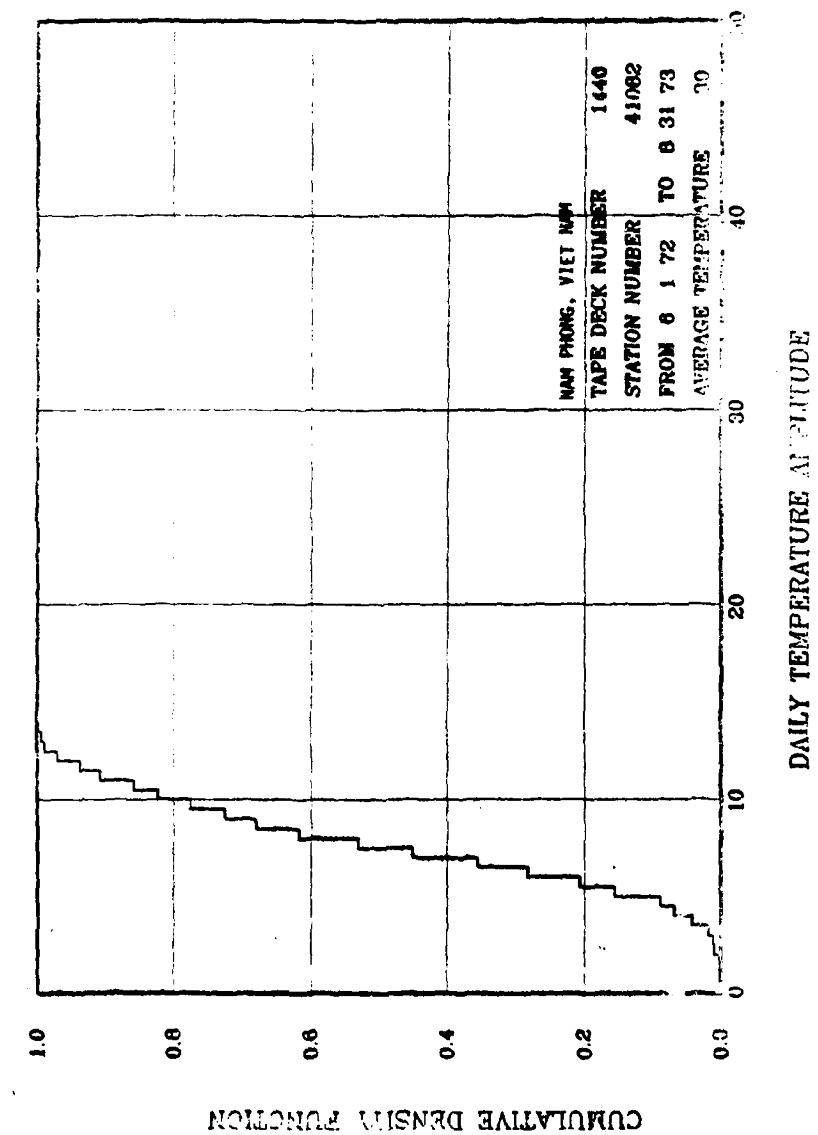

IWAKUNI
TAPE DECK NUMBER 1440
STATION NUMBER 43324
FROM 11 1 52 TO 12 31 78
0
10
20
30
40
DAILY TEMPERATURE
0.0
0.2
0.4
0.6
0.8
1.0
CUMULATIVE DENSITY FUNCTION

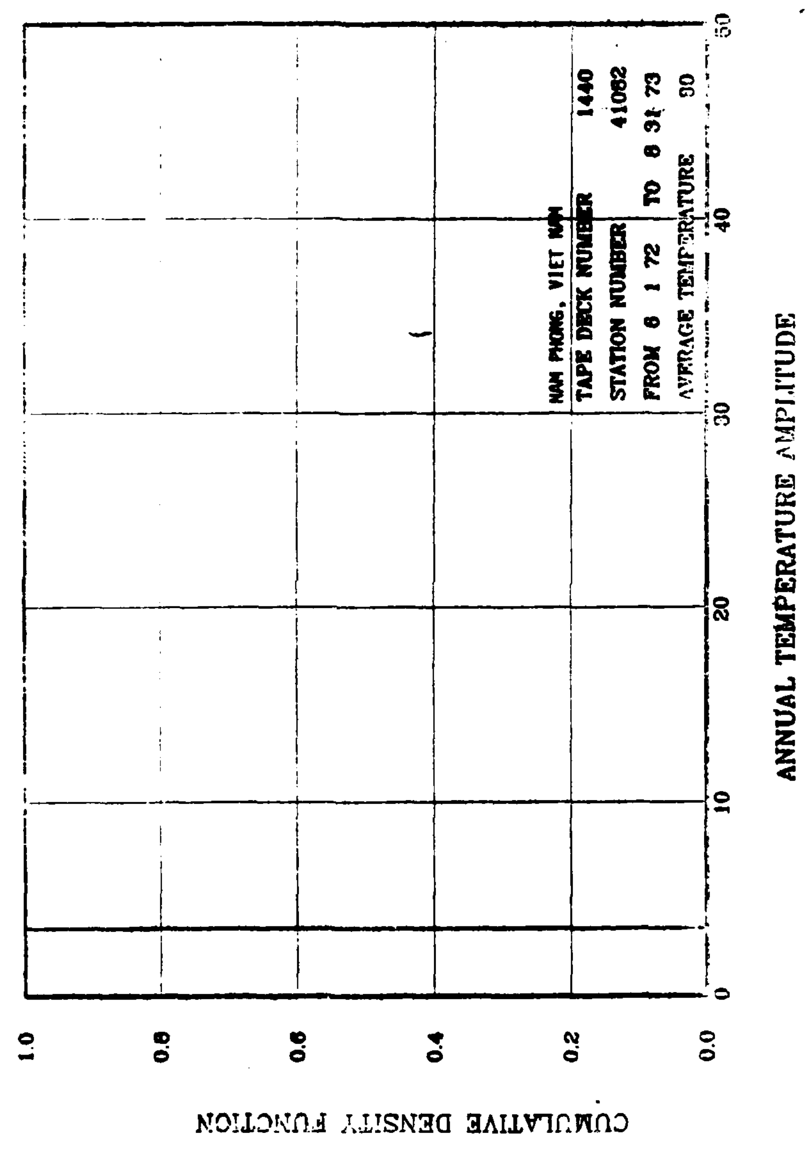

OKINAWA IS/NA
TAPE DECK NUMBER 1440
STATION NUMBER 43206
FROM 5 1 49 TO 3 31 68
AVERAGE TEMPERATURE 73
0
10
20
30
40
50
ANNUAL TEMPERATURE
1.0
0.8
0.6
0.4
0.2
0.0
CUMULATIVE DENSITY FUNCTION

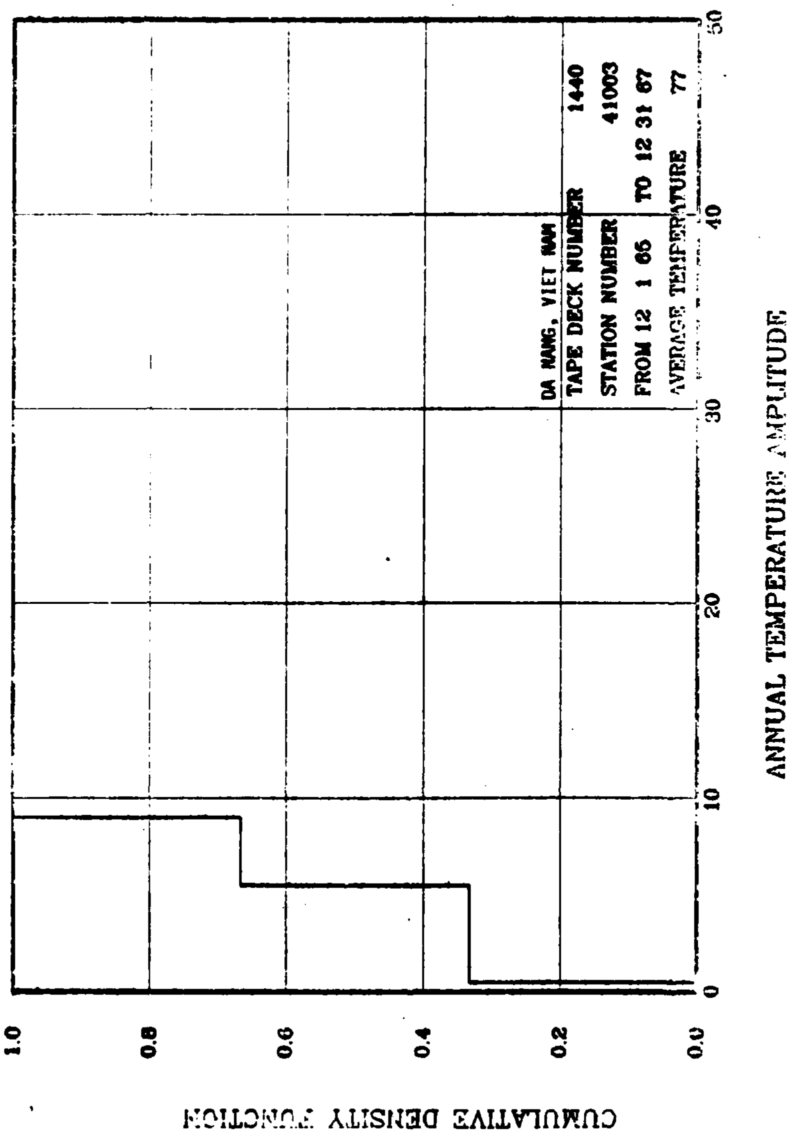

OKINAWA IS/NA
TAPE DECK NUMBER 1440
STATION NUMBER 42206
FROM 5 1 49 TO 3 31 68
AVERAGE TEMPERATURE 73
0
10
20
30
40
50
DAILY TEMPERATURE AMPLITUDE
0.0
0.2
0.4
0.6
0.8
1.0
CUMULATIVE DENSITY FUNCTION

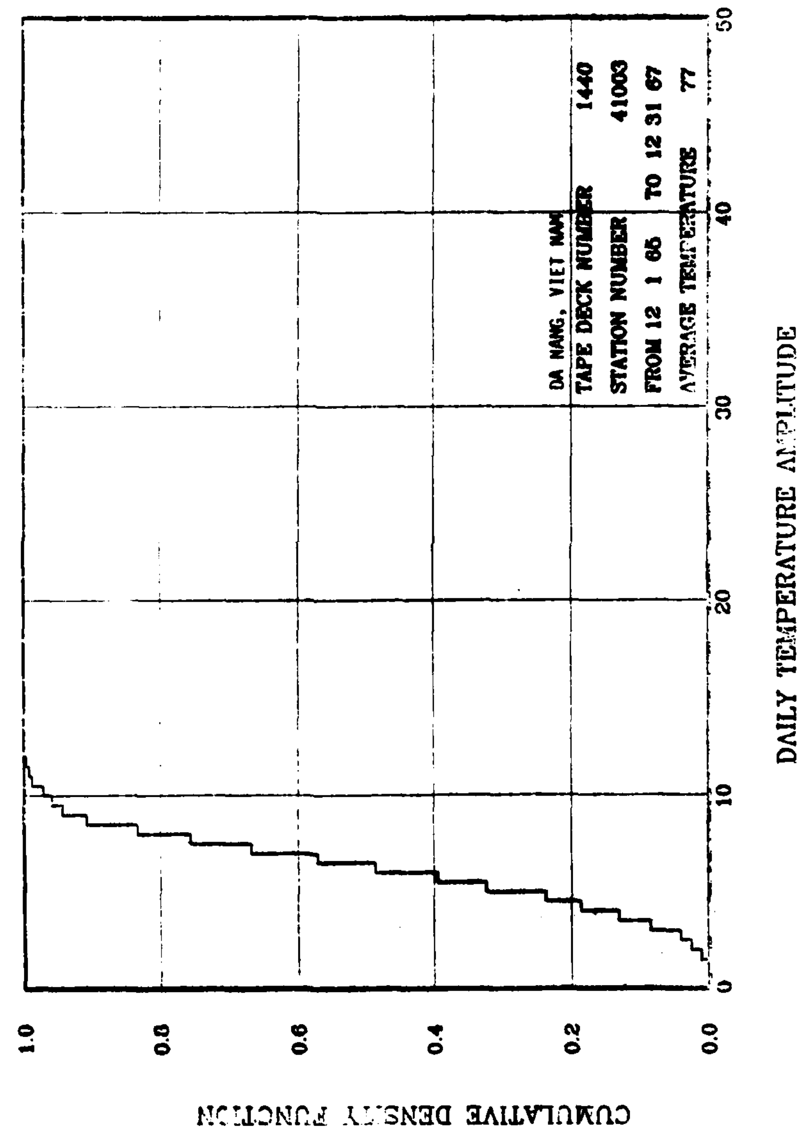

GUAM/AGANA
TAPE DECK NUMBER 1440
STATION NUMBER 41406
FROM 9 1 45 TO 12 31 78
AVERAGE TEMPERATURE 81
0
10
20
30
40
50
DAILY TEMPERATURE AMPLITUDE
0.0
0.2
0.4
0.6
0.8
1.0
CUMULATIVE DENSITY FUNCTION

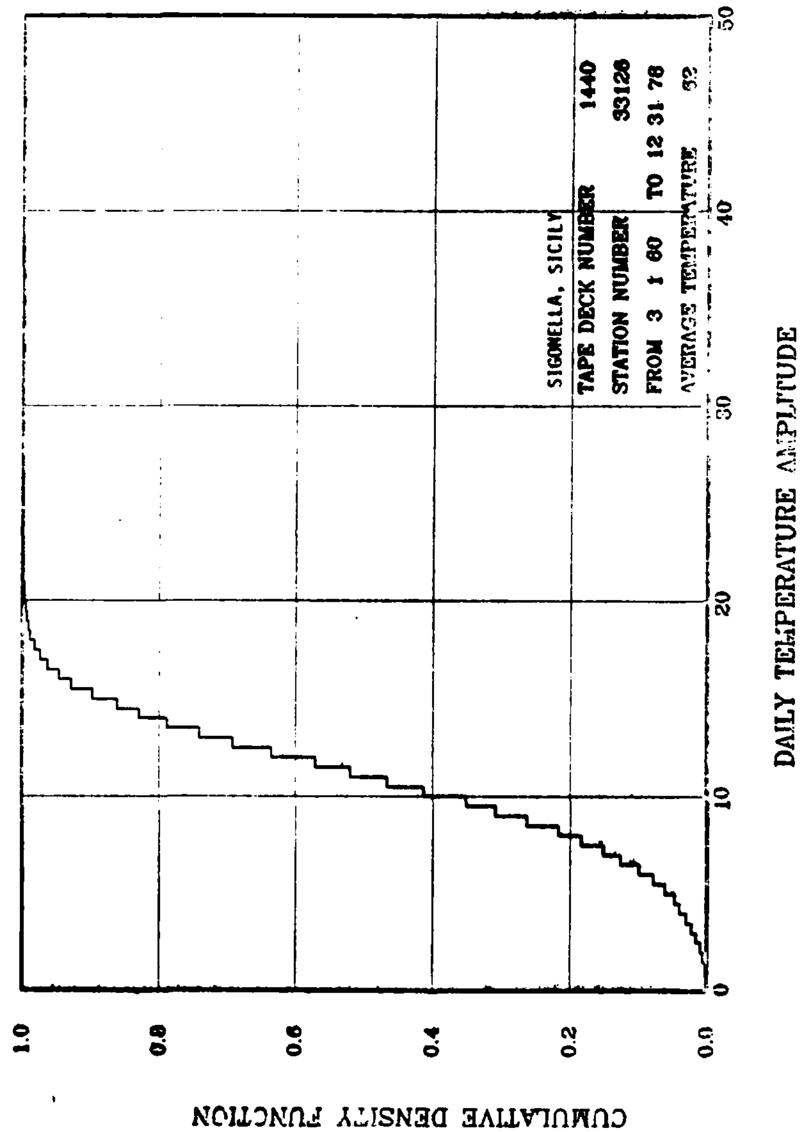

GUAM/AGANA
TAPE DECK NUMBER 1440
STATION NUMBER 41406
FROM 9 1 45 TO 12 31 76
AVERAGE TEMPERATURE 81
0
10
20
30
40
50
ANNUAL TEMPERATURE AMPLITUDE
1.0
0.8
0.6
0.4
0.2
0.0
CUMULATIVE DENSITY FUNCTION

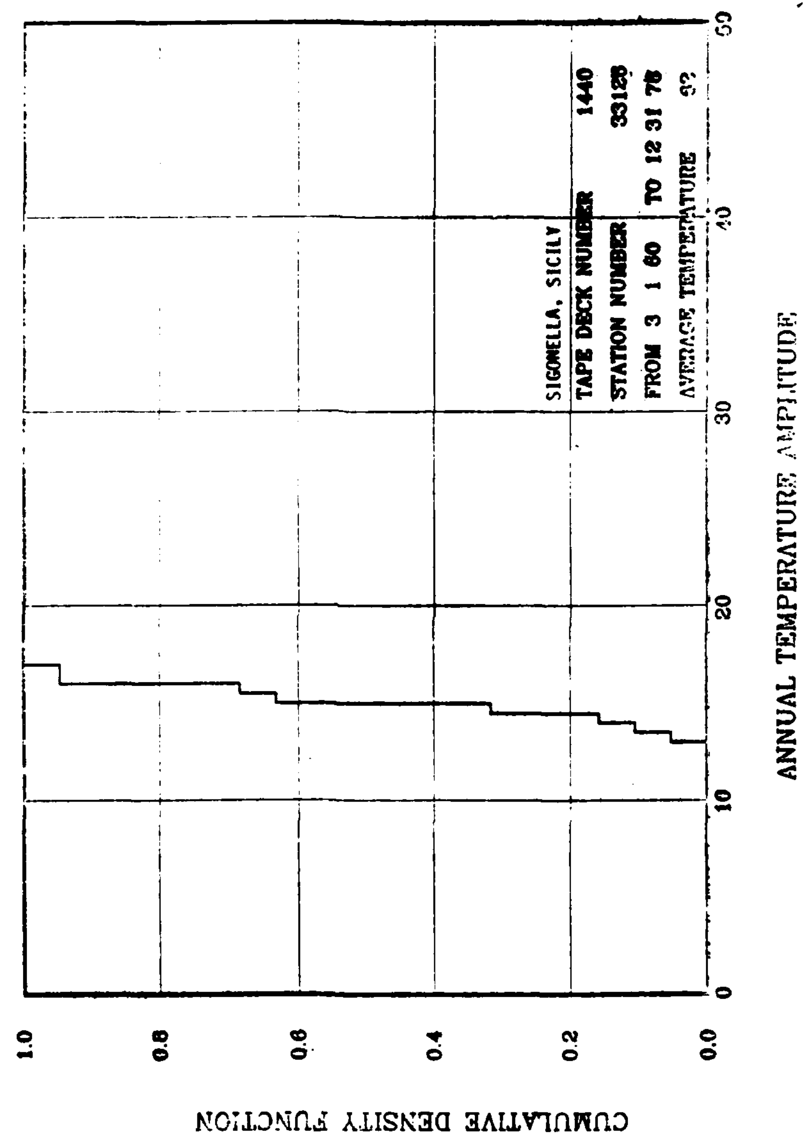

NAM PHONG, VIET NAM
TAPE DECK NUMBER 1440
STATION NUMBER 41062
0
10
20
30
40
0.0
0.2
0.4
0.6
0.8
1.0
DAILY TEMPERATURE

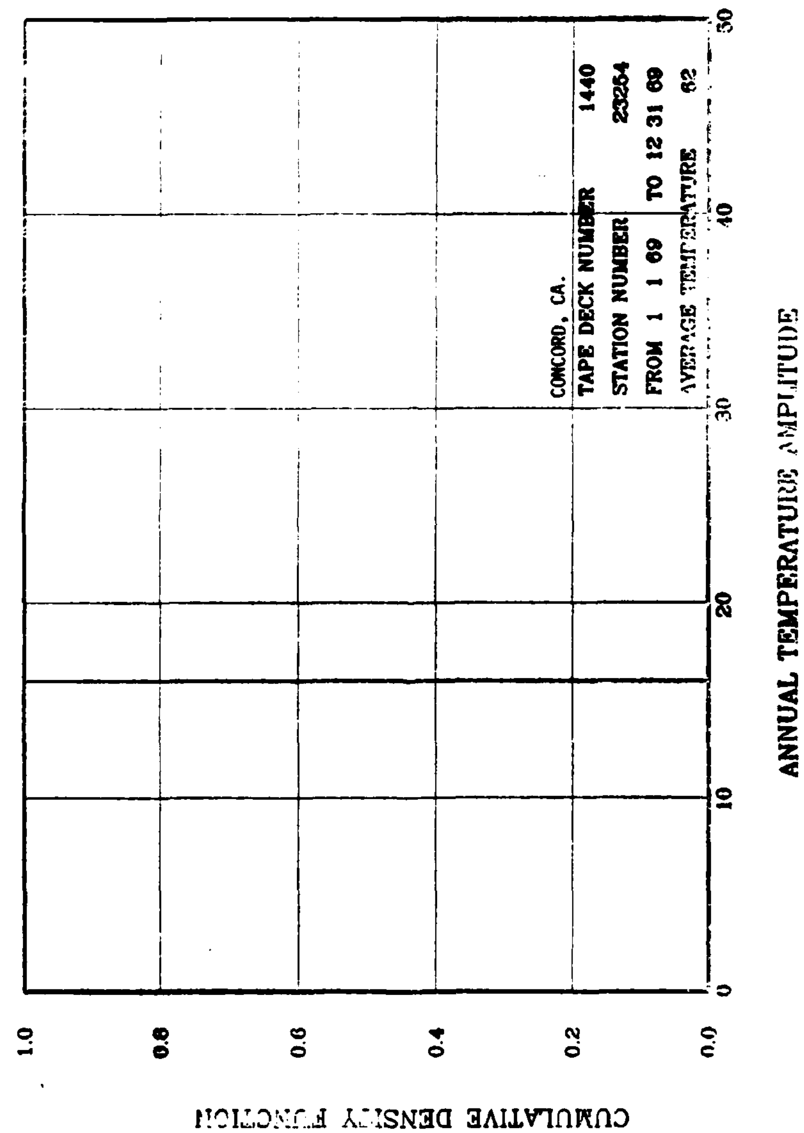

NAM PHONG, VIET NAM
TAPE DECK NUMBER 1440
STATION NUMBER 41082
ANNUAL TEMPERATURE AMPLITUDE
CUMULATIVE DENSITY FUNCTION
0
10
20
30
40
50
0.0
0.2
0.4
0.6
0.8
1.0

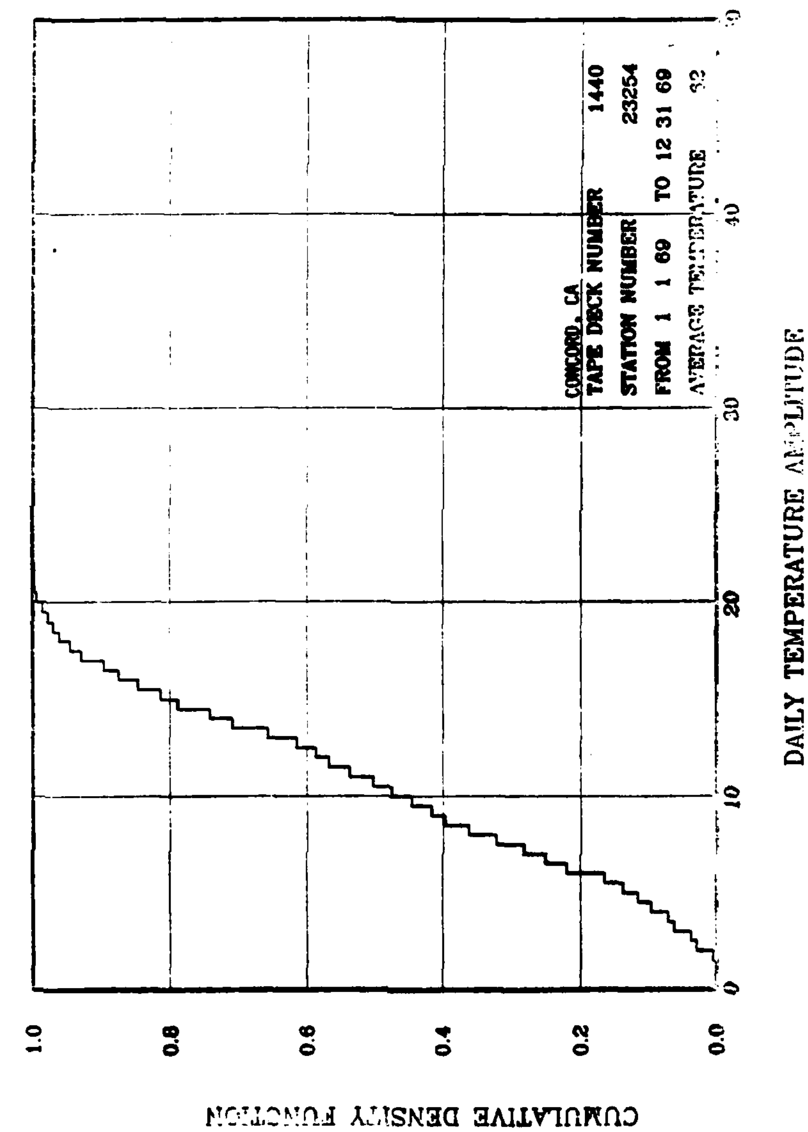

DA NANG, VIET NAM
TAPE DECK NUMBER 1440
STATION NUMBER 41003
FROM 12 1 65 TO 12 31 67
AVERAGE TEMPERATURE 77
ANNUAL TEMPERATURE AMPLITUDE
CUMULATIVE DENSITY FUNCTION
0
10
20
30
40
50
0.0
0.2
0.4
0.6
0.8
1.0


DA NANG, VIET NAM
TAPE DECK NUMBER 1440
STATION NUMBER 41003
FROM 12 1 65 TO 12 31 67
AVERAGE TEMPERATURE 77
DAILY TEMPERATURE AMPLITUDE
CUMULATIVE DENSITY FUNCTION
0 10 20 30 40 50
0.0 0.2 0.4 0.6 0.8 1.0


SIGONELLA, SICILY
TAPE DECK NUMBER 1440
STATION NUMBER 33126
FROM 3 1 60 TO 12 31 78
AVERAGE TEMPERATURE 32
DAILY TEMPERATURE AMPLITUDE
CUMULATIVE DENSITY FUNCTION
0
10
20
30
40
50
0.0
0.2
0.4
0.6
0.8
1.0


SIGONELLA, SICILY
TAPE DECK NUMBER 1440
STATION NUMBER 33128
FROM 3 1 60 TO 12 31 78
AVERAGE TEMPERATURE
ANNUAL TEMPERATURE AMPLITUDE
CUMULATIVE DENSITY FUNCTION
0 10 20 30 40 50
0.0 0.2 0.4 0.6 0.8 1.0


CONCORD, CA.
TAPE DECK NUMBER 1440
STATION NUMBER 23254
FROM 1 1 69 TO 12 31 69
AVERAGE TEMPERATURE 62
0
10
20
30
40
ANNUAL TEMPERATURE AMPLITUDE
0.0
0.2
0.4
0.6
0.8
1.0
CUMULATIVE DENSITY FUNCTION


CONCORD, CA
TAPE DECK NUMBER 1440
STATION NUMBER 23254
FROM 1 1 69 TO 12 31 69
AVERAGE TEMPERATURE 32
0
10
20
30
40
DAILY TEMPERATURE AMPLITUDE
0.0
0.2
0.4
0.6
0.8
1.0
CUMULATIVE DENSITY FUNCTION

APPENDIX B

Parameters Used for the Solution
of the Heat Equation

$$E(r) = \mathrm{ber}\ p_2 r + c\ \mathrm{ker}\ p_2 r - d\ \mathrm{kei}\ p_2 r$$

$$F(r) = \mathrm{bei}\ p_2 r + c\ \mathrm{kei}\ p_2 r + d\ \mathrm{ker}\ p_2 r$$

where

$p_2^2 = \omega/k_2$

ω = frequency (annual or diurnal)

k_2 = thermal diffusivity of propellant

ber, bei, ker, dei are Kelvin functions of order zero (Ref. 23)

$c = \mathrm{Re}\ (\rho)$

$d = \mathrm{Im}\ (\rho)$

$$\rho = \frac{I_1(\sqrt{i}\ p_2 a)\ I_o(\sqrt{i}\ p_1 a) - \dfrac{C_1 P_1}{C_2 P_2} I_1(\sqrt{i}\ p_1 a)\ I_o(\sqrt{i}\ p_2 a)}{K_1(\sqrt{i}\ p_2 a)\ I_o(\sqrt{i}\ p_1 a) + \dfrac{C_1 P_1}{C_2 P_2} I_1(\sqrt{i}\ p_1 a)\ K_o(\sqrt{i}\ p_2 a)}$$

$p_1^2 = \omega/k_1$ (k_1 is the thermal diffusivity of air)

$i = \sqrt{-1}$

I_o, I_1, K_o, K_1 are modified Bessel functions (Ref. 23)

C_1 and C_2 are thermal conductivities of air and propellant respectively.

APPENDIX C

Location Codes
(Duration of Stay in Each Location
Transition Markov Matrix)

ROCKET LO- ION CODES

1	W5A
2	P5N
3	ISRAE
4	WED
5	KITT
6	GU M
7	[illegible] JE
8	CORA
9	JFK
10	MIDW
11	CONS
12	SUBIC
13	CONC
14	YUMA
15	SEAL
16	MIRA
17	DALL
18	RANG
19	KANE
20	ENTE
21	CRIS
22	SHAS
23	ATSUG
24	SDAR
25	SURI
26	CCEA
27	ELT
28	SARA
29	BUTT
30	MIMI
31	FORR
32	ROOS
33	NAHA
34	KADE
35	FLIN
36	HULL
37	NMC
38	NAFS
39	BAKE
40	BEAU
41	AOR5
42	NCR
43	SINGP
44	AMER
45	FOR
46	DANG
47	DET
48	VF-17
49	NITR
50	NELL
51	VF-43
52	ROTA
53	VF-10
54	VF-11
55	CHER
56	EISEN
57	KEY
58	CANI
59	IWAK
60	CAMD
61	HALE
62	GA1
63	KISK
64	D59
65	NPG
66	CVSG-
67	GF1
68	FSRPA

NUMBER OI IS SPENT IN EACH LOCATION

1	0.4775E+06
2	0.1227E+06
3	0.2304E+05
4	0.9034E+05
5	0.8318E+04
6	0.1393E+02
7	0.1360E+05
8	0.1514E+05
9	0.5076E+05
10	0.3494E+05
11	0.1086E+05
12	0.1001E+05
13	0.6297E+05
14	0.9695E+04
15	0.4734E+04
16	0.2961E+05
17	0.4470E+03
18	0.4092E+04
19	0.3687E+04
20	0.1862E+05
21	0.5590E+04
22	0.9790E+03
23	0.8090E+03
24	0.1975E+04
25	0.4119E+03
26	0.1177E+05
27	0.2107E+04
28	0.1903E+05
29	0.7508E+04
30	0.8113E+04
31	0.2417E+04
32	0.6877E+04
33	0.3092E+04
34	0.9550E+03
35	0.1200E+02
36	0.1950E+03
37	0.2383E+04
38	0.2980E+03
39	0.2296E+04
40	0.4537E+04
41	0.1400E+02
42	0.1800E+02
43	0.1735E+04
44	0.2439E+05
45	0.1898E+04
46	-0.2542E+00
47	0.2975E+04
48	0.7000E+03
49	0.1192E+04
50	0.1230E+03
51	0.1880E+04
52	0.1215E+04
53	0.9110E+03
54	0.7000E+03
55	0.2953E+04
56	0.2402E+04
57	0.8589E+03
58	0.0
59	0.5029E+04
60	0.[illegible]0E+03
61	0.2170E+03
62	0.1166E+04
63	0.3500E+03
64	0.2300E+03
65	0.5220E+04
66	0.4000E+01
67	0.4946E+01
68	0.2740E+03
69	0.3071E+03

ABSORBING STATE 19
ABSORBING STATE 43
ABSORBING STATE 48
ABSORBING STATE 51
ABSORBING STATE 52
ABSORBING STATE 54

RELATIVE [illegible] SPENT IN EACH LOCATION

1	0.1430E+04		
2	0.1496E+04	60	0.[illegible]0E+02
3	0.2560E+04	61	0.1085E+03
4	0.6022E+03	62	0.1665E+03
5	0.5941E+03	63	0.1267E+03
6	0.1393E+02	64	0.2300E+03
7	0.7157E+03	65	0.2610E+04
8	0.5824E+03	66	0.2000E+01
9	0.7356E+03	67	0.4946E+01
10	0.5294E+03	68	0.2740E+03
11	0.8356E+03	69	0.7546E-01
12	0.3708E+03		
13	0.7409E+03		
14	0.4848E+04		
15	0.5260E+03		
16	0.1974E+04		
17	0.4470E+03		
18	0.5946E+03		
19	0.3687E+04		
20	0.7163E+03		
21	0.2942E+03		
22	0.1088E+03		
23	0.2022E+03		
24	0.3950E+03		
25	0.1030E+03		
26	0.3677E+03		
27	0.1053E+04		
28	0.3660E+03		
29	0.3003E+03		
30	0.3527E+03		
31	0.2417E+04		
32	0.1146E+04		
33	0.7731E+03		
34	0.9550E+03		
35	0.3000E+01		
36	0.6500E+02		
37	0.1192E+04		
38	0.7450E+02		
39	0.2551E+03		
40	0.5671E+03		
41	0.7000E+01		
42	0.1800E+02		
43	0.1735E+04		
44	0.9756E+03		
45	0.9489E+03		
46	-0.8472E-01		
47	0.4958E+03		
48	0.7000E+03		
49	0.1490E+03		
50	0.1230E+03		
51	0.1880E+04		
52	0.1215E+04		
53	0.9110E+03		
54	0.7000E+03		
55	0.1477E+04		
56	0.1201E+04		
57	0.4295E+03		
58	0.0		
59	0.4191E+03		

THE PROBABILITY MATRIX

ROW NUMBER= 1

1 0.0	2 0.004301	3 0.032258	4 0.008602	5 0.0
6 0.002151	7 0.023656	8 0.0	9 0.156989	10 0.017204
11 0.008602	12 0.0	13 0.010753	14 0.006452	15 0.004301
16 0.025806	17 0.0	18 0.004301	19 0.004301	20 0.006452
21 0.0	22 0.0	23 0.0	24 0.010753	25 0.008602
26 0.062366	27 0.0	28 0.101075	29 0.049462	30 0.030108
31 0.002151	32 0.017204	33 0.002151	34 0.0	35 0.0
36 0.0	37 0.0	38 0.002151	39 0.010753	40 0.017204
41 0.002151	42 0.0	43 0.002151	44 0.047312	45 0.0
46 0.0	47 0.010753	48 0.0	49 0.015054	50 0.0
51 0.0	52 0.0	53 0.0	54 0.0	55 0.004301
56 0.0	57 0.004301	58 0.0	59 0.002151	60 0.0
61 0.0	62 0.0	63 0.0	64 0.0	65 0.0
66 0.0	67 0.0	68 0.0	69 0.281720	

ROW NUMBER= 2

1 0.118812	2 0.0	3 0.009901	4 0.059406	5 0.009901
6 0.0	7 0.0	8 0.029703	9 0.0	10 0.168317
11 0.0	12 0.039604	13 0.198020	14 0.0	15 0.029703
16 0.009901	17 0.0	18 0.009901	19 0.0	20 0.0
21 0.009901	22 0.009901	23 0.0	24 0.0	25 0.0
26 0.009901	27 0.009901	28 0.0	29 0.0	30 0.009901
31 0.0	32 0.009901	33 0.009901	34 0.009901	35 0.029703
36 0.0	37 0.0	38 0.0	39 0.0	40 0.0
41 0.0	42 0.0	43 0.0	44 0.009901	45 0.0
46 0.0	47 0.0	48 0.0	49 0.0	50 0.0
51 0.0	52 0.0	53 0.0	54 0.0	55 0.0
56 0.0	57 0.0	58 0.0	59 0.009901	60 0.0
61 0.0	62 0.0	63 0.0	64 0.0	65 0.0
66 0.0	67 0.0	68 0.0	69 0.188119	

ROW NUMBER= 3

1 0.700000	2 0.0	3 0.0	4 0.0	5 0.0
6 0.0	7 0.0	8 0.100000	9 0.0	10 0.0
11 0.0	12 0.0	13 0.0	14 0.0	15 0.0
16 0.100000	17 0.0	18 0.0	19 0.0	20 0.0
21 0.0	22 0.0	23 0.0	24 0.0	25 0.0
26 0.0	27 0.0	28 0.0	29 0.0	30 0.0
31 0.0	32 0.0	33 0.0	34 0.0	35 0.0
36 0.0	37 0.0	38 0.0	39 0.0	40 0.0
41 0.0	42 0.0	43 0.0	44 0.0	45 0.0
46 0.0	47 0.0	48 0.0	49 0.0	50 0.0
51 0.0	52 0.0	53 0.0	54 0.0	55 0.0
56 0.0	57 0.0	58 0.0	59 0.0	60 0.0
61 0.0	62 0.0	63 0.0	64 0.0	65 0.0
66 0.0	67 0.0	68 0.0	69 0.100000	

ROW NUMBER= 4

1 0.027650	2 0.225806	3 0.004608	4 0.0	5 0.036866
6 0.0	7 0.004608	8 0.041475	9 0.004608	10 0.105991
11 0.027650	12 0.0	13 0.064516	14 0.0	15 0.009217
16 0.050691	17 0.0	18 0.0	19 0.004608	20 0.018433
21 0.018433	22 0.0	23 0.0	24 0.0	25 0.0
26 0.0	27 0.004608	28 0.0	29 0.0	30 0.0
31 0.0	32 0.0	33 0.0	34 0.0	35 0.0
36 0.0	37 0.004608	38 0.004608	39 0.0	40 0.0
41 0.0	42 0.0	43 0.0	44 0.009217	45 0.0

```
46 0.0         47 0.0         48 0.0         49 0.0         50 0.0
51 0.0         52 0.0         53 0.0         54 0.0         55 0.0
56 0.0         57 0.0         58 0.0         59 0.004608    60 0.0
61 0.0         62 0.0         63 0.0         64 0.0         65 0.013825
66 0.0         67 0.0         68 0.004608    69 0.308756

ROW NUMBER=  5
 1 0.007916     2 0.007916     3 0.0          4 0.0          5 0.0
 6 0.0          7 0.0          8 0.0          9 0.0         10 0.002639
11 0.0         12 0.0         13 0.002639    14 0.0         15 0.0
16 0.0         17 0.002639    18 0.002639    19 0.0         20 0.002639
21 0.0         22 0.0         23 0.0         24 0.0         25 0.0
26 0.0         27 0.0         28 0.0         29 0.0         30 0.002639
31 0.002639    32 0.0         33 0.0         34 0.0         35 0.0
36 0.0         37 0.0         38 0.002639    39 0.0         40 0.0
41 0.0         42 0.0         43 0.0         44 0.0         45 0.0
46 0.0         47 0.0         48 0.0         49 0.0         50 0.0
51 0.0         52 0.0         53 0.0         54 0.0         55 0.0
56 0.0         57 0.0         58 0.0         59 0.0         60 0.0
61 0.0         62 0.0         63 0.0         64 0.0         65 0.0
66 0.0         67 0.0         68 0.0         69 0.963061

ROW NUMBER=  6
 1 0.0          2 0.0          3 0.0          4 0.0          5 0.0
 6 0.0          7 0.500000     8 0.0          9 0.0         10 0.0
11 0.0         12 0.0         13 0.0         14 0.0         15 0.0
16 0.0         17 0.0         18 0.0         19 0.0         20 0.0
21 0.0         22 0.0         23 0.0         24 0.0         25 0.0
26 0.0         27 0.0         28 0.0         29 0.0         30 0.0
31 0.0         32 0.0         33 0.0         34 0.0         35 0.0
36 0.0         37 0.0         38 0.0         39 0.0         40 0.0
41 0.0         42 0.0         43 0.0         44 0.0         45 0.0
46 0.0         47 0.0         48 0.0         49 0.0         50 0.0
51 0.0         52 0.0         53 0.0         54 0.0         55 0.0
56 0.0         57 0.0         58 0.0         59 0.0         60 0.0
61 0.0         62 0.0         63 0.0         64 0.0         65 0.0
66 0.0         67 0.0         68 0.0         69 0.500000

ROW NUMBER=  7
 1 0.085890     2 0.0          3 0.0          4 0.006135     5 0.0
 6 0.0          7 0.0          8 0.006135     9 0.0         10 0.0
11 0.0         12 0.0         13 0.0         14 0.0         15 0.0
16 0.006135    17 0.0         18 0.0         19 0.0         20 0.006135
21 0.0         22 0.0         23 0.0         24 0.0         25 0.0
26 0.0         27 0.0         28 0.006135    29 0.0         30 0.0
31 0.0         32 0.0         33 0.0         34 0.0         35 0.0
36 0.0         37 0.0         38 0.0         39 0.0         40 0.0
41 0.0         42 0.0         43 0.0         44 0.0         45 0.0
46 0.0         47 0.0         48 0.0         49 0.0         50 0.0
51 0.0         52 0.0         53 0.0         54 0.0         55 0.0
56 0.0         57 0.0         58 0.0         59 0.0         60 0.0
61 0.0         62 0.0         63 0.0         64 0.0         65 0.0
66 0.0         67 0.0         68 0.0         69 0.883436

ROW NUMBER=  8
 1 0.013825     2 0.0          3 0.004608     4 0.009217     5 0.009217
 6 0.0          7 0.0          8 0.0          9 0.004608    10 0.009217
11 0.004608    12 0.0         13 0.032258    14 0.004608    15 0.0
16 0.009217    17 0.004608    18 0.0         19 0.0         20 0.0
21 0.0         22 0.0         23 0.0         24 0.0         25 0.0
```

```
26 0.0          27 0.0          28 0.0          29 0.0          30 0.0
31 0.0          32 0.0          33 0.0          34 0.0          35 0.004608
36 0.0          37 0.0          38 0.0          39 0.0          40 0.0
41 0.0          42 0.0          43 0.0          44 0.004608     45 0.0
46 0.0          47 0.0          48 0.0          49 0.0          50 0.0
51 0.0          52 0.0          53 0.0          54 0.0          55 0.0
56 0.0          57 0.0          58 0.0          59 0.004608     60 0.0
61 0.0          62 0.0          63 0.0          64 0.0          65 0.0
66 0.0          67 0.0          68 0.0          69 0.880184

ROW NUMBER=  9
 1 0.195286      2 0.0           3 0.0           4 0.0           5 0.0
 6 0.0           7 0.0           8 0.0           9 0.0          10 0.003367
11 0.003367     12 0.0          13 0.013468     14 0.0          15 0.0
16 0.0          17 0.0          18 0.0          19 0.0          20 0.006734
21 0.0          22 0.0          23 0.0          24 0.0          25 0.0
26 0.003367     27 0.0          28 0.0          29 0.0          30 0.0
31 0.0          32 0.0          33 0.0          34 0.0          35 0.0
36 0.0          37 0.0          38 0.0          39 0.003367     40 0.0
41 0.0          42 0.0          43 0.0          44 0.0          45 0.0
46 0.0          47 0.0          48 0.0          49 0.0          50 0.0
51 0.0          52 0.003367     53 0.0          54 0.0          55 0.0
56 0.0          57 0.0          58 0.0          59 0.0          60 0.0
61 0.0          62 0.0          63 0.0          64 0.0          65 0.0
66 0.0          67 0.0          68 0.0          69 0.767677

ROW NUMBER= 10
 1 0.004637      2 0.005410      3 0.0           4 0.000773      5 0.0
 6 0.0           7 0.0           8 0.002318      9 0.001546     10 0.0
11 0.0          12 0.013138     13 0.003091     14 0.001546     15 0.0
16 0.0          17 0.0          18 0.000773     19 0.0          20 0.001546
21 0.000773     22 0.0          23 0.002318     24 0.0          25 0.0
26 0.0          27 0.0          28 0.000773     29 0.0          30 0.0
31 0.0          32 0.0          33 0.000773     34 0.0          35 0.0
36 0.0          37 0.0          38 0.0          39 0.0          40 0.000773
41 0.0          42 0.000773     43 0.0          44 0.003091     45 0.0
46 0.002318     47 0.0          48 0.0          49 0.000773     50 0.0
51 0.0          52 0.0          53 0.0          54 0.0          55 0.0
56 0.0          57 0.0          58 0.0          59 0.0          60 0.0
61 0.000773     62 0.0          63 0.002318     64 0.0          65 0.000773
66 0.0          67 0.0          68 0.0          69 0.948995

ROW NUMBER= 11
 1 0.009390      2 0.004695      3 0.0           4 0.0           5 0.0
 6 0.0           7 0.004695      8 0.004695      9 0.004695     10 0.0
11 0.0          12 0.004695     13 0.004695     14 0.0          15 0.^
16 0.009390     17 0.0          18 0.0          19 0.0          20 0.009390
21 0.0          22 0.0          23 0.0          24 0.0          25 0.0
26 0.0          27 0.0          28 0.0          29 0.0          30 0.0
31 0.0          32 0.0          33 0.0          34 0.0          35 0.0
36 0.0          37 0.004695     38 0.0          39 0.0          40 0.0
41 0.0          42 0.0          43 0.0          44 0.0          45 0.0
46 0.0          47 0.0          48 0.0          49 0.0          50 0.0
51 0.0          52 0.0          53 0.0          54 0.0          55 0.0
56 0.0          57 0.0          58 0.0          59 0.0          60 0.0
61 0.0          62 0.0          63 0.0          64 0.0          65 0.0
66 0.0          67 0.0          68 0.0          69 0.938967
```

ROW NUMBER= 12

1 0.031250	2 0.250000	3 0.0	4 0.0	5 0.0
6 0.0	7 0.0	8 0.0	9 0.0	10 0.187500
11 0.0	12 0.0	13 0.250000	14 0.0	15 0.0
16 0.031250	17 0.0	18 0.0	19 0.0	20 0.0
21 0.0	22 0.0	23 0.0	24 0.0	25 0.0
26 0.0	27 0.0	28 0.031250	29 0.0	30 0.031250
31 0.0	32 0.0	33 0.0	34 0.0	35 0.0
36 0.031250	37 0.0	38 0.0	39 0.0	40 0.0
41 0.0	42 0.0	43 0.0	44 0.0	45 0.0
46 0.0	47 0.0	48 0.0	49 0.0	50 0.0
51 0.0	52 0.0	53 0.0	54 0.0	55 0.0
56 0.0	57 0.0	58 0.0	59 0.0	60 0.0
61 0.0	62 0.0	63 0.0	64 0.0	65 0.0
66 0.0	67 0.0	68 0.0	69 0.156250	

ROW NUMBER= 13

1 0.080000	2 0.100000	3 0.010000	4 0.030000	5 0.010000
6 0.0	7 0.0	8 0.040000	9 0.020000	10 0.020000
11 0.010000	12 0.020000	13 0.0	14 0.030000	15 0.0
16 0.030000	17 0.0	18 0.020000	19 0.010000	20 0.140000
21 0.140000	22 0.010000	23 0.0	24 0.0	25 0.0
26 0.020000	27 0.010000	28 0.0	29 0.0	30 0.010000
31 0.0	32 0.010000	33 0.020000	34 0.0	35 0.0
36 0.0	37 0.0	38 0.010000	39 0.0	40 0.0
41 0.0	42 0.0	43 0.0	44 0.0	45 0.0
46 0.0	47 0.0	48 0.0	49 0.0	50 0.0
51 0.010000	52 0.0	53 0.0	54 0.0	55 0.0
56 0.0	57 0.0	58 0.0	59 0.030000	60 0.0
61 0.010000	62 0.0	63 0.0	64 0.0	65 0.0
66 0.0	67 0.0	68 0.0	69 0.150000	

ROW NUMBER= 14

1 0.0	2 0.005495	3 0.0	4 0.0	5 0.0
6 0.0	7 0.0	8 0.0	9 0.0	10 0.0
11 0.0	12 0.0	13 0.0	14 0.0	15 0.0
16 0.0	17 0.0	18 0.0	19 0.0	20 0.0
21 0.0	22 0.0	23 0.005495	24 0.0	25 0.0
26 0.0	27 0.0	28 0.0	29 0.0	30 0.0
31 0.0	32 0.0	33 0.0	34 0.0	35 0.0
36 0.0	37 0.0	38 0.0	39 0.0	40 0.0
41 0.0	42 0.0	43 0.0	44 0.0	45 0.0
46 0.0	47 0.0	48 0.0	49 0.0	50 0.0
51 0.0	52 0.0	53 0.0	54 0.0	55 0.0
56 0.0	57 0.0	58 0.0	59 0.0	60 0.0
61 0.0	62 0.0	63 0.0	64 0.0	65 0.0
66 0.0	67 0.0	68 0.0	69 0.989011	

ROW NUMBER= 15

1 0.090909	2 0.090909	3 0.0	4 0.090909	5 0.090909
6 0.0	7 0.0	8 0.0	9 0.090909	10 0.090909
11 0.090909	12 0.0	13 0.090909	14 0.0	15 0.0
16 0.0	17 0.0	18 0.0	19 0.0	20 0.0
21 0.0	22 0.0	23 0.0	24 0.0	25 0.0
26 0.0	27 0.0	28 0.0	29 0.0	30 0.0
31 0.0	32 0.0	33 0.0	34 0.0	35 0.0
36 0.0	37 0.0	38 0.0	39 0.0	40 0.0
41 0.0	42 0.0	43 0.0	44 0.0	45 0.0
46 0.0	47 0.090909	48 0.0	49 0.0	50 0.0
51 0.0	52 0.0	53 0.0	54 0.0	55 0.0
56 0.0	57 0.0	58 0.0	59 0.0	60 0.0
61 0.0	62 0.0	63 0.0	64 0.0	65 0.0
66 0.0	67 0.0	68 0.0	69 0.181818	

ROW NUMBER= 16

1 0.080000	2 0.040000	3 0.0	4 0.0	5 0.0
6 0.0	7 0.0	8 0.020000	9 0.0	10 0.020000
11 0.020000	12 0.020000	13 0.020000	14 0.020000	15 0.020000
16 0.0	17 0.0	18 0.0	19 0.0	20 0.0
21 0.0	22 0.0	23 0.0	24 0.0	25 0.0
26 0.020000	27 0.0	28 0.0	29 0.0	30 0.0
31 0.0	32 0.0	33 0.0	34 0.0	35 0.0
36 0.0	37 0.0	38 0.0	39 0.0	40 0.0
41 0.0	42 0.0	43 0.0	44 0.0	45 0.020000
46 0.0	47 0.0	48 0.0	49 0.0	50 0.0
51 0.0	52 0.0	53 0.0	54 0.0	55 0.0
56 0.0	57 0.0	58 0.0	59 0.0	60 0.0
61 0.0	62 0.0	63 0.0	64 0.0	65 0.0
66 0.0	67 0.0	68 0.0	69 0.700000	

ROW NUMBER= 17

1 0.0	2 0.0	3 0.0	4 0.0	5 0.0
6 0.0	7 0.0	8 0.0	9 0.0	10 0.0
11 0.0	12 0.0	13 0.0	14 0.0	15 0.0
16 0.0	17 0.0	18 0.0	19 0.0	20 0.0
21 0.0	22 0.0	23 0.0	24 0.0	25 0.0
26 0.0	27 0.0	28 0.0	29 0.0	30 0.0
31 0.0	32 0.0	33 0.0	34 0.0	35 0.0
36 0.0	37 0.0	38 0.0	39 0.0	40 0.0
41 0.0	42 0.0	43 0.0	44 0.0	45 0.0
46 0.0	47 0.0	48 0.0	49 0.0	50 0.0
51 0.0	52 0.0	53 0.0	54 0.0	55 0.0
56 0.0	57 0.0	58 0.0	59 0.500000	60 0.0
61 0.0	62 0.0	63 0.0	64 0.0	65 0.0
66 0.0	67 0.0	68 0.0	69 0.500000	

ROW NUMBER= 18

1 0.125000	2 0.062500	3 0.0	4 0.0	5 0.0
6 0.0	7 0.0	8 0.062500	9 0.0	10 0.0
11 0.0	12 0.0	13 0.0	14 0.0	15 0.0
16 0.0	17 0.0	18 0.0	19 0.0	20 0.0
21 0.0	22 0.0	23 0.0	24 0.0	25 0.0
26 0.0	27 0.0	28 0.062500	29 0.0	30 0.0
31 0.0	32 0.0	33 0.0	34 0.0	35 0.0
36 0.0	37 0.0	38 0.0	39 0.0	40 0.0
41 0.0	42 0.0	43 0.0	44 0.0	45 0.0
46 0.0	47 0.0	48 0.0	49 0.0	50 0.0
51 0.0	52 0.0	53 0.0	54 0.0	55 0.0
56 0.0	57 0.0	58 0.0	59 0.062500	60 0.0
61 0.0	62 0.062500	63 0.0	64 0.0	65 0.0
66 0.0	67 0.0	68 0.0	69 0.562500	

ROW NUMBER= 19

1 0.0	2 0.0	3 0.0	4 0.0	5 0.0
6 0.0	7 0.0	8 0.0	9 0.0	10 0.0
11 0.0	12 0.0	13 0.0	14 0.0	15 0.0
16 0.0	17 0.0	18 0.0	19 1.000000	20 0.0
21 0.0	22 0.0	23 0.0	24 0.0	25 0.0
26 0.0	27 0.0	28 0.0	29 0.0	30 0.0
31 0.0	32 0.0	33 0.0	34 0.0	35 0.0
36 0.0	37 0.0	38 0.0	39 0.0	40 0.0
41 0.0	42 0.0	43 0.0	44 0.0	45 0.0

```
46 0.0      47 0.0      48 0.0      49 0.0      50 0.0
51 0.0      52 0.0      53 0.0      54 0.0      55 0.0
56 0.0      57 0.0      58 0.0      59 0.0      60 0.0
61 0.0      62 0.0      63 0.0      64 0.0      65 0.0
66 0.0      67 0.0      68 0.0      6910.000000

ROW NUMBER= 20
 1 0.007937  2 0.003968  3 0.0       4 0.001984  5 0.001984
 6 0.0       7 0.001984  8 0.001984  9 0.001984 10 0.001984
11 0.001984 12 0.001984 13 0.009921 14 0.0      15 0.001984
16 0.0      17 0.0      18 0.0      19 0.0      20 0.0
21 0.0      22 0.0      23 0.0      24 0.0      25 0.0
26 0.0      27 0.0      28 0.0      29 0.0      30 0.0
31 0.0      32 0.0      33 0.0      34 0.0      35 0.0
36 0.0      37 0.0      38 0.0      39 0.0      40 0.0
41 0.0      42 0.0      43 0.0      44 0.001984 45 0.0
46 0.0      47 0.0      48 0.0      49 0.0      50 0.0
51 0.0      52 0.0      53 0.0      54 0.0      55 0.0
56 0.0      57 0.0      58 0.0      59 0.0      60 0.009921
61 0.0      62 0.0      63 0.0      64 0.0      65 0.0
66 0.0      67 0.0      68 0.0      69 0.948413

ROW NUMBER= 21
 1 0.0       2 0.044118  3 0.0       4 0.0       5 0.0
 6 0.0       7 0.0       8 0.0       9 0.014706 10 0.0
11 0.0      12 0.0      13 0.132353 14 0.0      15 0.0
16 0.0      17 0.0      18 0.0      19 0.0      20 0.0
21 0.0      22 0.088235 23 0.0      24 0.0      25 0.0
26 0.0      27 0.0      28 0.0      29 0.0      30 0.0
31 0.0      32 0.0      33 0.0      34 0.0      35 0.0
36 0.0      37 0.0      38 0.0      39 0.0      40 0.0
41 0.0      42 0.0      43 0.0      44 0.0      45 0.0
46 0.0      47 0.0      48 0.0      49 0.0      50 0.0
51 0.0      52 0.0      53 0.0      54 0.0      55 0.0
56 0.0      57 0.0      58 0.0      59 0.0      60 0.0
61 0.0      62 0.0      63 0.0      64 0.0      65 0.0
66 0.0      67 0.0      68 0.0      69 0.720588

ROW NUMBER= 22
 1 0.0       2 0.555556  3 0.0       4 0.0       5 0.0
 6 0.0       7 0.0       8 0.0       9 0.0      10 0.0
11 0.0      12 0.0      13 0.444444 14 0.0      15 0.0
16 0.0      17 0.0      18 0.0      19 0.0      20 0.0
21 0.0      22 0.0      23 0.0      24 0.0      25 0.0
26 0.0      27 0.0      28 0.0      29 0.0      30 0.0
31 0.0      32 0.0      33 0.0      34 0.0      35 0.0
36 0.0      37 0.0      38 0.0      39 0.0      40 0.0
41 0.0      42 0.0      43 0.0      44 0.0      45 0.0
46 0.0      47 0.0      48 0.0      49 0.0      50 0.0
51 0.0      52 0.0      53 0.0      54 0.0      55 0.0
56 0.0      57 0.0      58 0.0      59 0.0      60 0.0
61 0.0      62 0.0      63 0.0      64 0.0      65 0.0
66 0.0      67 0.0      68 0.0      69 0.0

ROW NUMBER= 23
 1 0.0       2 0.750000  3 0.0       4 0.0       5 0.0
 6 0.0       7 0.0       8 0.0       9 0.0      10 0.0
11 0.0      12 0.0      13 0.0      14 0.0      15 0.0
16 0.0      17 0.0      18 0.0      19 0.0      20 0.0
21 0.0      22 0.0      23 0.0      24 0.0      25 0.0
```

```
26 0.0        27 0.0        28 0.0        29 0.0        30 0.0
31 0.0        32 0.0        33 0.0        34 0.0        35 0.0
36 0.0        37 0.250000   38 0.0        39 0.0        40 0.0
41 0.0        42 0.0        43 0.0        44 0.0        45 0.0
46 0.0        47 0.0        48 0.0        49 0.0        50 0.0
51 0.0        52 0.0        53 0.0        54 0.0        55 0.0
56 0.0        57 0.0        58 0.0        59 0.0        60 0.0
61 0.0        62 0.0        63 0.0        64 0.0        65 0.0
66 0.0        67 0.0        68 0.0        69 0.0

ROW NUMBER= 24
 1 0.428571    2 0.0         3 0.0         4 0.0         5 0.0
 6 0.0         7 0.0         8 0.0         9 0.0        10 0.0
11 0.0        12 0.0        13 0.0        14 0.0        15 0.0
16 0.142857   17 0.0        18 0.0        19 0.0        20 0.0
21 0.0        22 0.0        23 0.0        24 0.0        25 0.0
26 0.0        27 0.0        28 0.0        29 0.0        30 0.0
31 0.0        32 0.0        33 0.0        34 0.0        35 0.0
36 0.0        37 0.0        38 0.0        39 0.0        40 0.142857
41 0.0        42 0.0        43 0.0        44 0.0        45 0.0
46 0.0        47 0.0        48 0.0        49 0.0        50 0.0
51 0.0        52 0.0        53 0.0        54 0.0        55 0.0
56 0.0        57 0.0        58 0.0        59 0.0        60 0.0
61 0.0        62 0.0        63 0.0        64 0.0        65 0.0
66 0.0        67 0.0        68 0.0        69 0.285714

ROW NUMBER= 25
 1 0.400000    2 0.0         3 0.0         4 0.0         5 0.0
 6 0.0         7 0.0         8 0.0         9 0.0        10 0.200000
11 0.0        12 0.0        13 0.0        14 0.0        15 0.0
16 0.0        17 0.0        18 0.200000   19 0.0        20 0.0
21 0.0        22 0.0        23 0.0        24 0.0        25 0.0
26 0.0        27 0.0        28 0.0        29 0.0        30 0.0
31 0.0        32 0.0        33 0.0        34 0.0        35 0.0
36 0.0        37 0.0        38 0.0        39 0.0        40 0.0
41 0.0        42 0.0        43 0.0        44 0.0        45 0.0
46 0.0        47 0.0        48 0.0        49 0.0        50 0.0
51 0.0        52 0.0        53 0.0        54 0.0        55 0.0
56 0.0        57 0.0        58 0.0        59 0.0        60 0.0
61 0.0        62 0.0        63 0.0        64 0.0        65 0.0
66 0.0        67 0.0        68 0.0        69 0.200000

ROW NUMBER= 26
 1 0.100000    2 0.0         3 0.0         4 0.0         5 0.0
 6 0.0         7 0.0         8 0.0         9 0.020000   10 0.0
11 0.0        12 0.0        13 0.020000   14 0.0        15 0.0
16 0.0        17 0.0        18 0.0        19 0.0        20 0.0
21 0.0        22 0.0        23 0.020000   24 0.0        25 0.0
26 0.0        27 0.020000   28 0.020000   29 0.0        30 0.060000
31 0.0        32 0.040000   33 0.0        34 0.0        35 0.0
36 0.0        37 0.020000   38 0.0        39 0.0        40 0.0
41 0.0        42 0.0        43 0.0        44 0.040000   45 0.0
46 0.0        47 0.0        48 0.040000   49 0.0        50 0.0
51 0.020000   52 0.0        53 0.040000   54 0.020000   55 0.0
56 0.100000   57 0.0        58 0.020000   59 0.0        60 0.0
61 0.0        62 0.0        63 0.0        64 0.0        65 0.0
66 0.040000   67 0.0        68 0.0        69 0.360000

ROW NUMBER= 27
 1 0.0         2 0.0         3 0.0         4 0.250000    5 0.0
```

6 0.0	7 0.0	8 0.0	9 0.0	10 0.0
11 0.0	12 0.0	13 0.0	14 0.0	15 0.250000
16 0.0	17 0.0	18 0.0	19 0.0	20 0.0
21 0.0	22 0.0	23 0.0	24 0.0	25 0.0
26 0.0	27 0.0	28 0.0	29 0.0	30 0.0
31 0.0	32 0.0	33 0.0	34 0.0	35 0.0
36 0.0	37 0.0	38 0.0	39 0.0	40 0.0
41 0.0	42 0.0	43 0.0	44 0.0	45 0.0
46 0.0	47 0.0	48 0.0	49 0.0	50 0.0
51 0.0	52 0.0	53 0.0	54 0.0	55 0.0
56 0.0	57 0.0	58 0.0	59 0.0	60 0.0
61 0.0	62 0.0	63 0.0	64 0.0	65 0.0
66 0.0	67 0.0	68 0.0	69 0.500000	

ROW NUMBER= 28

1 0.307143	2 0.0	3 0.0	4 0.007143	5 0.0
6 0.0	7 0.014286	8 0.0	9 0.007143	10 0.0
11 0.0	12 0.0	13 0.007143	14 0.0	15 0.0
16 0.007143	17 0.0	18 0.0	19 0.0	20 0.007143
21 0.0	22 0.0	23 0.0	24 0.0	25 0.0
26 0.0	27 0.0	28 0.0	29 0.014286	30 0.0
31 0.0	32 0.0	33 0.0	34 0.0	35 0.0
36 0.0	37 0.0	38 0.0	39 0.0	40 0.0
41 0.0	42 0.0	43 0.0	44 0.0	45 0.0
46 0.0	47 0.0	48 0.0	49 0.0	50 0.0
51 0.0	52 0.0	53 0.0	54 0.0	55 0.0
56 0.0	57 0.0	58 0.0	59 0.0	60 0.0
61 0.0	62 0.0	63 0.0	64 0.0	65 0.0
66 0.0	67 0.0	68 0.0	69 0.628571	

ROW NUMBER= 29

1 0.653846	2 0.0	3 0.0	4 0.0	5 0.0
6 0.0	7 0.0	8 0.0	9 0.0	10 0.0
11 0.0	12 0.0	13 0.0	14 0.0	15 0.0
16 0.0	17 0.0	18 0.0	19 0.0	20 0.038462
21 0.0	22 0.0	23 0.0	24 0.0	25 0.0
26 0.0	27 0.0	28 0.0	29 0.0	30 0.230769
31 0.0	32 0.0	33 0.0	34 0.0	35 0.0
36 0.0	37 0.0	38 0.0	39 0.0	40 0.0
41 0.038462	42 0.0	43 0.0	44 0.0	45 0.0
46 0.0	47 0.0	48 0.0	49 0.0	50 0.0
51 0.0	52 0.0	53 0.0	54 0.0	55 0.0
56 0.0	57 0.0	58 0.0	59 0.0	60 0.0
61 0.0	62 0.0	63 0.0	64 0.0	65 0.0
66 0.0	67 0.0	68 0.0	69 0.038462	

ROW NUMBER= 30

1 0.363636	2 0.0	3 0.0	4 0.0	5 0.0
6 0.0	7 0.0	8 0.0	9 0.0	10 0.0
11 0.0	12 0.0	13 0.0	14 0.0	15 0.0
16 0.0	17 0.0	18 0.0	19 0.0	20 0.0
21 0.0	22 0.0	23 0.0	24 0.0	25 0.0
26 0.018182	27 0.0	28 0.0	29 0.0	30 0.0
31 0.0	32 0.0	33 0.0	34 0.0	35 0.0
36 0.0	37 0.0	38 0.0	39 0.0	40 0.0
41 0.0	42 0.0	43 0.0	44 0.036364	45 0.0
46 0.0	47 0.0	48 0.0	49 0.0	50 0.0
51 0.0	52 0.0	53 0.0	54 0.0	55 0.0
56 0.0	57 0.0	58 0.0	59 0.0	60 0.0
61 0.0	62 0.0	63 0.0	64 0.0	65 0.0
66 0.0	67 0.0	68 0.0	69 0.581818	

ROW NUMBER= 31

1	0.0	2	0.0	3	0.0	4	0.0	5	0.0
6	0.0	7	0.0	8	0.0	9	0.0	10	0.0
11	0.0	12	0.0	13	0.0	14	0.0	15	0.0
16	0.0	17	0.0	18	0.0	19	0.0	20	0.0
21	0.0	22	0.0	23	0.0	24	0.0	25	0.0
26	0.083333	27	0.0	28	0.0	29	0.0	30	0.0
31	0.0	32	0.0	33	0.0	34	0.0	35	0.0
36	0.0	37	0.0	38	0.0	39	0.0	40	0.0
41	0.0	42	0.0	43	0.0	44	0.0	45	0.0
46	0.0	47	0.0	48	0.0	49	0.0	50	0.0
51	0.0	52	0.0	53	0.0	54	0.0	55	0.0
56	0.0	57	0.0	58	0.0	59	0.0	60	0.0
61	0.0	62	0.0	63	0.0	64	0.0	65	0.0
66	0.0	67	0.0	68	0.0	69	0.916667		

ROW NUMBER= 32

1	0.157895	2	0.0	3	0.0	4	0.0	5	0.0
6	0.0	7	0.0	8	0.0	9	0.0	10	0.0
11	0.0	12	0.0	13	0.0	14	0.0	15	0.0
16	0.0	17	0.0	18	0.0	19	0.0	20	0.0
21	0.0	22	0.052632	23	0.0	24	0.0	25	0.0
26	0.105263	27	0.0	28	0.0	29	0.0	30	0.0
31	0.0	32	0.0	33	0.0	34	0.0	35	0.0
36	0.0	37	0.0	38	0.0	39	0.0	40	0.0
41	0.0	42	0.0	43	0.0	44	0.0	45	0.0
46	0.0	47	0.0	48	0.0	49	0.0	50	0.0
51	0.0	52	0.0	53	0.0	54	0.0	55	0.0
56	0.0	57	0.0	58	0.0	59	0.0	60	0.0
61	0.0	62	0.0	63	0.0	64	0.0	65	0.0
66	0.0	67	0.0	68	0.0	69	0.684210		

ROW NUMBER= 33

1	0.083333	2	0.0	3	0.0	4	0.0	5	0.0
6	0.0	7	0.0	8	0.0	9	0.0	10	0.0
11	0.0	12	0.0	13	0.083333	14	0.0	15	0.0
16	0.0	17	0.0	18	0.0	19	0.0	20	0.0
21	0.0	22	0.0	23	0.0	24	0.0	25	0.0
26	0.083333	27	0.0	28	0.0	29	0.0	30	0.0
31	0.0	32	0.0	33	0.0	34	0.083333	35	0.0
36	0.0	37	0.0	38	0.0	39	0.0	40	0.0
41	0.0	42	0.0	43	0.0	44	0.0	45	0.0
46	0.0	47	0.0	48	0.0	49	0.0	50	0.0
51	0.0	52	0.0	53	0.0	54	0.0	55	0.0
56	0.0	57	0.0	58	0.0	59	0.0	60	0.0
61	0.0	62	0.0	63	0.0	64	0.0	65	0.0
66	0.0	67	0.0	68	0.0	69	0.666667		

ROW NUMBER= 34

1	0.0	2	0.0	3	0.0	4	0.0	5	0.0
6	0.0	7	0.0	8	0.0	9	0.0	10	0.0
11	0.0	12	0.0	13	1.000000	14	0.0	15	0.0
16	0.0	17	0.0	18	0.0	19	0.0	20	0.0
21	0.0	22	0.0	23	0.0	24	0.0	25	0.0
26	0.0	27	0.0	28	0.0	29	0.0	30	0.0
31	0.0	32	0.0	33	0.0	34	0.0	35	0.0
36	0.0	37	0.0	38	0.0	39	0.0	40	0.0
41	0.0	42	0.0	43	0.0	44	0.0	45	0.0

```
46 0.0          47 0.0          48 0.0          49 0.0          50 0.0
51 0.0          52 0.0          53 0.0          54 0.0          55 0.0
56 0.0          57 0.0          58 0.0          59 0.0          60 0.0
61 0.0          62 0.0          63 0.0          64 0.0          65 0.0
66 0.0          67 0.0          68 0.0          69 0.0

ROW NUMBER= 35
 1 0.0           2 0.0           3 0.0           4 0.0           5 0.0
 6 0.0           7 0.0           8 0.0           9 0.0          10 0.250000
11 0.0          12 0.250000     13 0.0          14 0.0          15 0.0
16 0.0          17 0.0          18 0.0          19 0.0          20 0.0
21 0.0          22 0.0          23 0.0          24 0.0          25 0.0
26 0.0          27 0.0          28 0.0          29 0.0          30 0.0
31 0.0          32 0.0          33 0.0          34 0.0          35 0.0
36 0.500000     37 0.0          38 0.0          39 0.0          40 0.0
41 0.0          42 0.0          43 0.0          44 0.0          45 0.0
46 0.0          47 0.0          48 0.0          49 0.0          50 0.0
51 0.0          52 0.0          53 0.0          54 0.0          55 0.0
56 0.0          57 0.0          58 0.0          59 0.0          60 0.0
61 0.0          62 0.0          63 0.0          64 0.0          65 0.0
66 0.0          67 0.0          68 0.0          69 0.0

ROW NUMBER= 36
 1 0.0           2 0.333333      3 0.0           4 0.0           5 0.0
 6 0.0           7 0.0           8 0.0           9 0.0          10 0.0
11 0.0          12 0.0          13 0.666667     14 0.0          15 0.0
16 0.0          17 0.0          18 0.0          19 0.0          20 0.0
21 0.0          22 0.0          23 0.0          24 0.0          25 0.0
26 0.0          27 0.0          28 0.0          29 0.0          30 0.0
31 0.0          32 0.0          33 0.0          34 0.0          35 0.0
36 0.0          37 0.0          38 0.0          39 0.0          40 0.0
41 0.0          42 0.0          43 0.0          44 0.0          45 0.0
46 0.0          47 0.0          48 0.0          49 0.0          50 0.0
51 0.0          52 0.0          53 0.0          54 0.0          55 0.0
56 0.0          57 0.0          58 0.0          59 0.0          60 0.0
61 0.0          62 0.0          63 0.0          64 0.0          65 0.0
66 0.0          67 0.0          68 0.0          69 0.0

ROW NUMBER= 37
 1 0.0           2 0.0           3 0.0           4 0.0           5 0.0
 6 0.0           7 0.0           8 0.0           9 0.0          10 0.0
11 0.0          12 0.0          13 0.0          14 0.0          15 0.0
16 0.500000     17 0.0          18 0.0          19 0.0          20 0.0
21 0.0          22 0.0          23 0.0          24 0.0          25 0.0
26 0.500000     27 0.0          28 0.0          29 0.0          30 0.0
31 0.0          32 0.0          33 0.0          34 0.0          35 0.0
36 0.0          37 0.0          38 0.0          39 0.0          40 0.0
41 0.0          42 0.0          43 0.0          44 0.0          45 0.0
46 0.0          47 0.0          48 0.0          49 0.0          50 0.0
51 0.0          52 0.0          53 0.0          54 0.0          55 0.0
56 0.0          57 0.0          58 0.0          59 0.0          60 0.0
61 0.0          62 0.0          63 0.0          64 0.0          65 0.0
66 0.0          67 0.0          68 0.0          69 0.0

ROW NUMBER= 38
 1 0.500000      2 0.0           3 0.0           4 0.250000      5 0.0
 6 0.0           7 0.0           8 0.0           9 0.0          10 0.0
11 0.0          12 0.0          13 0.0          14 0.0          15 0.0
16 0.0          17 0.0          18 0.0          19 0.0          20 0.0
21 0.0          22 0.0          23 0.0          24 0.0          25 0.0
```

26 0.0	27 0.0	28 0.0	29 0.0	30 0.0
31 0.0	32 0.0	33 0.0	34 0.0	35 0.0
36 0.0	37 0.0	38 0.0	39 0.250000	40 0.0
41 0.0	42 0.0	43 0.0	44 0.0	45 0.0
46 0.0	47 0.0	48 0.0	49 0.0	50 0.0
51 0.0	52 0.0	53 0.0	54 0.0	55 0.0
56 0.0	57 0.0	58 0.0	59 0.0	60 0.0
61 0.0	62 0.0	63 0.0	64 0.0	65 0.0
66 0.0	67 0.0	68 0.0	69 0.0	

ROW NUMBER= 39

1 0.800000	2 0.0	3 0.0	4 0.0	5 0.0
6 0.0	7 0.0	8 0.0	9 0.0	10 0.0
11 0.0	12 0.0	13 0.0	14 0.0	15 0.0
16 0.0	17 0.0	18 0.0	19 0.0	20 0.0
21 0.0	22 0.0	23 0.0	24 0.0	25 0.0
26 0.0	27 0.0	28 0.100000	29 0.0	30 0.0
31 0.0	32 0.0	33 0.0	34 0.0	35 0.0
36 0.0	37 0.0	38 0.0	39 0.0	40 0.0
41 0.0	42 0.0	43 0.0	44 0.0	45 0.0
46 0.0	47 0.0	48 0.0	49 0.0	50 0.0
51 0.0	52 0.0	53 0.0	54 0.0	55 0.0
56 0.0	57 0.0	58 0.0	59 0.0	60 0.0
61 0.0	62 0.0	63 0.0	64 0.0	65 0.0
66 0.0	67 0.0	68 0.0	69 0.100000	

ROW NUMBER= 40

1 0.146341	2 0.0	3 0.0	4 0.0	5 0.0
6 0.0	7 0.0	8 0.0	9 0.0	10 0.0
11 0.0	12 0.024390	13 0.0	14 0.0	15 0.0
16 0.0	17 0.0	18 0.0	19 0.0	20 0.0
21 0.0	22 0.0	23 0.0	24 0.0	25 0.0
26 0.0	27 0.0	28 0.0	29 0.0	30 0.0
31 0.0	32 0.0	33 0.0	34 0.0	35 0.0
36 0.0	37 0.0	38 0.0	39 0.0	40 0.0
41 0.0	42 0.0	43 0.0	44 0.0	45 0.0
46 0.0	47 0.0	48 0.0	49 0.0	50 0.0
51 0.0	52 0.0	53 0.0	54 0.0	55 0.024390
56 0.0	57 0.0	58 0.0	59 0.0	60 0.0
61 0.0	62 0.0	63 0.0	64 0.0	65 0.0
66 0.0	67 0.0	68 0.0	69 0.804878	

ROW NUMBER= 41

1 1.000000	2 0.0	3 0.0	4 0.0	5 0.0
6 0.0	7 0.0	8 0.0	9 0.0	10 0.0
11 0.0	12 0.0	13 0.0	14 0.0	15 0.0
16 0.0	17 0.0	18 0.0	19 0.0	20 0.0
21 0.0	22 0.0	23 0.0	24 0.0	25 0.0
26 0.0	27 0.0	28 0.0	29 0.0	30 0.0
31 0.0	32 0.0	33 0.0	34 0.0	35 0.0
36 0.0	37 0.0	38 0.0	39 0.0	40 0.0
41 0.0	42 0.0	43 0.0	44 0.0	45 0.0
46 0.0	47 0.0	48 0.0	49 0.0	50 0.0
51 0.0	52 0.0	53 0.0	54 0.0	55 0.0
56 0.0	57 0.0	58 0.0	59 0.0	6. 0.0
61 0.0	62 0.0	63 0.0	64 0.0	65 0.0
66 0.0	67 0.0	68 0.0	69 0.0	

ROW NUMBER= 42

1 0.0	2 0.0	3 1.000000	4 0.0	5 0.0
6 0.0	7 0.0	8 0.0	9 0.0	10 0.0
11 0.0	12 0.0	13 0.0	14 0.0	15 0.0
16 0.0	17 0.0	18 0.0	19 0.0	20 0.0
21 0.0	22 0.0	23 0.0	24 0.0	25 0.0
26 0.0	27 0.0	28 0.0	29 0.0	30 0.0
31 0.0	32 0.0	33 0.0	34 0.0	35 0.0
36 0.0	37 0.0	38 0.0	39 0.0	40 0.0
41 0.0	42 0.0	43 0.0	44 0.0	45 0.0
46 0.0	47 0.0	48 0.0	49 0.0	50 0.0
51 0.0	52 0.0	53 0.0	54 0.0	55 0.0
56 0.0	57 0.0	58 0.0	59 0.0	60 0.0
61 0.0	62 0.0	63 0.0	64 0.0	65 0.0
66 0.0	67 0.0	68 0.0	69 0.0	

ROW NUMBER= 43

1 0.0	2 0.0	3 0.0	4 0.0	5 0.0
6 0.0	7 0.0	8 0.0	9 0.0	10 0.0
11 0.0	12 0.0	13 0.0	14 0.0	15 0.0
16 0.0	17 0.0	18 0.0	19 0.0	20 0.0
21 0.0	22 0.0	23 0.0	24 0.0	25 0.0
26 0.0	27 0.0	28 0.0	29 0.0	30 0.0
31 0.0	32 0.0	33 0.0	34 0.0	35 0.0
36 0.0	37 0.0	38 0.0	39 0.0	40 0.0
41 0.0	42 0.0	43 1.000000	44 0.0	45 0.0
46 0.0	47 0.0	48 0.0	49 0.0	50 0.0
51 0.0	52 0.0	53 0.0	54 0.0	55 0.0
56 0.0	57 0.0	58 0.0	59 0.0	60 0.0
61 0.0	62 0.0	63 0.0	64 0.0	65 0.0
66 0.0	67 0.0	68 0.0	69 0.0	

ROW NUMBER= 44

1 0.034483	2 0.002463	3 0.0	4 0.0	5 0.002463
6 0.0	7 0.0	8 0.002463	9 0.0	10 0.002463
11 0.0	12 0.0	13 0.0	14 0.0	15 0.0
16 0.0	17 0.0	18 0.0	19 0.0	20 0.002463
21 0.0	22 0.0	23 0.0	24 0.0	25 0.0
26 0.0	27 0.0	28 0.002463	29 0.0	30 0.0
31 0.0	32 0.0	33 0.0	34 0.0	35 0.0
36 0.0	37 0.0	38 0.0	39 0.004926	40 0.002463
41 0.0	42 0.0	43 0.0	44 0.0	45 0.002463
46 0.0	47 0.002463	48 0.0	49 0.0	50 0.0
51 0.0	52 0.0	53 0.0	54 0.0	55 0.0
56 0.0	57 0.0	58 0.0	59 0.0	60 0.0
61 0.0	62 0.0	63 0.0	64 0.0	65 0.0
66 0.0	67 0.0	68 0.0	69 0.938424	

ROW NUMBER= 45

1 0.0	2 0.0	3 0.0	4 0.0	5 0.0
6 0.0	7 0.0	8 0.0	9 0.0	10 0.0
11 0.0	12 0.0	13 0.052632	14 0.0	15 0.0
16 0.0	17 0.0	18 0.0	19 0.0	20 0.0
21 0.0	22 0.0	23 0.0	24 0.0	25 0.0
26 0.0	27 0.0	28 0.0	29 0.0	30 0.0
31 0.0	32 0.0	33 0.0	34 0.0	35 0.0
36 0.0	37 0.0	38 0.0	39 0.0	40 0.0
41 0.0	42 0.0	43 0.0	44 0.052632	45 0.0
46 0.0	47 0.0	48 0.0	49 0.0	50 0.0
51 0.0	52 0.0	53 0.0	54 0.0	55 0.0
56 0.0	57 0.0	58 0.0	59 0.0	60 0.0
61 0.0	62 0.0	63 0.0	64 0.0	65 0.0
66 0.0	67 0.0	68 0.0	69 0.894737	

ROW NUMBER= 46

1 0.0	2 0.0	3 0.0	4 0.0	5 0.0
6 0.0	7 0.0	8 0.0	9 0.0	10 0.428571
11 0.0	12 0.0	13 0.0	14 0.0	15 0.0
16 0.0	17 0.0	18 0.0	19 0.0	20 0.0
21 0.0	22 0.0	23 0.0	24 0.0	25 0.0
26 0.0	27 0.0	28 0.0	29 0.0	30 0.0
31 0.0	32 0.0	33 0.0	34 0.0	35 0.0
36 0.0	37 0.0	38 0.0	39 0.0	40 0.0
41 0.0	42 0.0	43 0.0	44 0.0	45 0.0
46 0.0	47 0.0	48 0.0	49 0.0	50 0.0
51 0.0	52 0.0	53 0.0	54 0.0	55 0.0
56 0.0	57 0.0	58 0.0	59 0.0	60 0.0
61 0.0	62 0.0	63 0.0	64 0.0	65 0.0
66 0.0	67 0.0	68 0.0	69 0.571429	

ROW NUMBER= 47

1 0.375000	2 0.0	3 0.0	4 0.0	5 0.0
6 0.0	7 0.125000	8 0.0	9 0.0	10 0.0
11 0.0	12 0.0	13 0.0	14 0.0	15 0.0
16 0.0	17 0.0	18 0.0	19 0.0	20 0.0
21 0.0	22 0.0	23 0.0	24 0.0	25 0.0
26 0.0	27 0.0	28 0.125000	29 0.0	30 0.0
31 0.0	32 0.0	33 0.0	34 0.0	35 0.0
36 0.0	37 0.0	38 0.0	39 0.0	40 0.0
41 0.0	42 0.0	43 0.0	44 0.125000	45 0.0
46 0.0	47 0.0	48 0.0	49 0.0	50 0.0
51 0.0	52 0.0	53 0.0	54 0.0	55 0.0
56 0.0	57 0.0	58 0.0	59 0.0	60 0.0
61 0.0	62 0.0	63 0.0	64 0.0	65 0.0
66 0.0	67 0.0	68 0.0	69 0.250000	

ROW NUMBER= 48

1 0.0	2 0.0	3 0.0	4 0.0	5 0.0
6 0.0	7 0.0	8 0.0	9 0.0	10 0.0
11 0.0	12 0.0	13 0.0	14 0.0	15 0.0
16 0.0	17 0.0	18 0.0	19 0.0	20 0.0
21 0.0	22 0.0	23 0.0	24 0.0	25 0.0
26 0.0	27 0.0	28 0.0	29 0.0	30 0.0
31 0.0	32 0.0	33 0.0	34 0.0	35 0.0
36 0.0	37 0.0	38 0.0	39 0.0	40 0.0
41 0.0	42 0.0	43 0.0	44 0.0	45 0.0
46 0.0	47 0.0	48 1.000000	49 0.0	50 0.0
51 0.0	52 0.0	53 0.0	54 0.0	55 0.0
56 0.0	57 0.0	58 0.0	59 0.0	60 0.0
61 0.0	62 0.0	63 0.0	64 0.0	65 0.0
66 0.0	67 0.0	68 0.0	69 0.0	

ROW NUMBER= 49

1 0.444444	2 0.0	3 0.0	4 0.0	5 0.0
6 0.0	7 0.0	8 0.0	9 0.0	10 0.0
11 0.0	12 0.0	13 0.0	14 0.0	15 0.0
16 0.0	17 0.0	18 0.0	19 0.0	20 0.0
21 0.0	22 0.0	23 0.0	24 0.0	25 0.0
26 0.0	27 0.0	28 0.111111	29 0.0	30 0.111111
31 0.0	32 0.111111	33 0.0	34 0.0	35 0.0
36 0.0	37 0.0	38 0.0	39 0.0	40 0.0
41 0.0	42 0.0	43 0.0	44 0.0	45 0.0

46 0.0	47 0.0	48 0.0	49 0.0	50 0.111111
51 0.0	52 0.0	53 0.0	54 0.0	55 0.0
56 0.0	57 0.0	58 0.0	59 0.0	60 0.0
61 0.0	62 0.0	63 0.0	64 0.0	65 0.0
66 0.0	67 0.0	68 0.0	69 0.111111	

ROW NUMBER= 50

1 0.500000	2 0.0	3 0.0	4 0.0	5 0.0
6 0.0	7 0.0	8 0.0	9 0.0	10 0.0
11 0.0	12 0.0	13 0.0	14 0.0	15 0.0
16 0.0	17 0.0	18 0.0	19 0.0	20 0.0
21 0.0	22 0.0	23 0.0	24 0.0	25 0.0
26 0.0	27 0.0	28 0.0	29 0.0	30 0.0
31 0.0	32 0.0	33 0.0	34 0.0	35 0.0
36 0.0	37 0.0	38 0.0	39 0.0	40 0.0
41 0.0	42 0.0	43 0.0	44 0.0	45 0.0
46 0.0	47 0.0	48 0.0	49 0.0	50 0.0
51 0.0	52 0.0	53 0.0	54 0.0	55 0.0
56 0.0	57 0.0	58 0.0	59 0.0	60 0.0
61 0.0	62 0.0	63 0.0	64 0.0	65 0.0
66 0.0	67 0.0	68 0.0	69 0.500000	

ROW NUMBER= 51

1 0.0	2 0.0	3 0.0	4 0.0	5 0.0
6 0.0	7 0.0	8 0.0	9 0.0	10 0.0
11 0.0	12 0.0	13 0.0	14 0.0	15 0.0
16 0.0	17 0.0	18 0.0	19 0.0	20 0.0
21 0.0	22 0.0	23 0.0	24 0.0	25 0.0
26 0.0	27 0.0	28 0.0	29 0.0	30 0.0
31 0.0	32 0.0	33 0.0	34 0.0	35 0.0
36 0.0	37 0.0	38 0.0	39 0.0	40 0.0
41 0.0	42 0.0	43 0.0	44 0.0	45 0.0
46 0.0	47 0.0	48 0.0	49 0.0	50 0.0
51 1.000000	52 0.0	53 0.0	54 0.0	55 0.0
56 0.0	57 0.0	58 0.0	59 0.0	60 0.0
61 0.0	62 0.0	63 0.0	64 0.0	65 0.0
66 0.0	67 0.0	68 0.0	69 0.0	

ROW NUMBER= 52

1 0.0	2 0.0	3 0.0	4 0.0	5 0.0
6 0.0	7 0.0	8 0.0	9 0.0	10 0.0
11 0.0	12 0.0	13 0.0	14 0.0	15 0.0
16 0.0	17 0.0	18 0.0	19 0.0	20 0.0
21 0.0	22 0.0	23 0.0	24 0.0	25 0.0
26 0.0	27 0.0	28 0.0	29 0.0	30 0.0
31 0.0	32 0.0	33 0.0	34 0.0	35 0.0
36 0.0	37 0.0	38 0.0	39 0.0	40 0.0
41 0.0	42 0.0	43 0.0	44 0.0	45 0.0
46 0.0	47 0.0	48 0.0	49 0.0	50 0.0
51 0.0	52 1.000000	53 0.0	54 0.0	55 0.0
56 0.0	57 0.0	58 0.0	59 0.0	60 0.0
61 0.0	62 0.0	63 0.0	64 0.0	65 0.0
66 0.0	67 0.0	68 0.0	69 0.0	

ROW NUMBER= 53

1 0.0	2 0.0	3 0.0	4 0.0	5 0.0
6 0.0	7 0.0	8 0.0	9 0.0	10 0.0
11 0.0	12 0.0	13 0.0	14 0.0	15 0.0
16 0.0	17 0.0	18 0.0	19 0.0	20 0.0
21 0.0	22 0.0	23 0.0	24 0.0	25 0.0

26 0.500000	27 0.0	28 0.0	29 0.0	30 0.0
31 0.0	32 0.0	33 0.0	34 0.0	35 0.0
36 0.0	37 0.0	38 0.0	39 0.0	40 0.0
41 0.0	42 0.0	43 0.0	44 0.0	45 0.0
46 0.0	47 0.0	48 0.0	49 0.0	50 0.0
51 0.0	52 0.0	53 0.0	54 0.0	55 0.0
56 0.0	57 0.0	58 0.0	59 0.0	60 0.0
61 0.0	62 0.0	63 0.0	64 0.0	65 0.0
66 0.0	67 0.0	68 0.0	69 0.500000	

ROW NUMBER= 54

1 0.0	2 0.0	3 0.0	4 0.0	5 0.0
6 0.0	7 0.0	8 0.0	9 0.0	10 0.0
11 0.0	12 0.0	13 0.0	14 0.0	15 0.0
16 0.0	17 0.0	18 0.0	19 0.0	20 0.0
21 0.0	22 0.0	23 0.0	24 0.0	25 0.0
26 0.0	27 0.0	28 0.0	29 0.0	30 0.0
31 0.0	32 0.0	33 0.0	34 0.0	35 0.0
36 0.0	37 0.0	38 0.0	39 0.0	40 0.0
41 0.0	42 0.0	43 0.0	44 0.0	45 0.0
46 0.0	47 0.0	48 0.0	49 0.0	50 0.0
51 0.0	52 0.0	53 0.0	54 1.000000	55 0.0
56 0.0	57 0.0	58 0.0	59 0.0	60 0.0
61 0.0	62 0.0	63 0.0	64 0.0	65 0.0
66 0.0	67 0.0	68 0.0	69 0.0	

ROW NUMBER= 55

1 0.500000	2 0.0	3 0.0	4 0.0	5 0.0
6 0.0	7 0.0	8 0.0	9 0.0	10 0.0
11 0.0	12 0.0	13 0.0	14 0.500000	15 0.0
16 0.0	17 0.0	18 0.0	19 0.0	20 0.0
21 0.0	22 0.0	23 0.0	24 0.0	25 0.0
26 0.0	27 0.0	28 0.0	29 0.0	30 0.0
31 0.0	32 0.0	33 0.0	34 0.0	35 0.0
36 0.0	37 0.0	38 0.0	39 0.0	40 0.0
41 0.0	42 0.0	43 0.0	44 0.0	45 0.0
46 0.0	47 0.0	48 0.0	49 0.0	50 0.0
51 0.0	52 0.0	53 0.0	54 0.0	55 0.0
56 0.0	57 0.0	58 0.0	59 0.0	60 0.0
61 0.0	62 0.0	63 0.0	64 0.0	65 0.0
66 0.0	67 0.0	68 0.0	69 0.0	

ROW NUMBER= 56

1 0.111111	2 0.0	3 0.0	4 0.0	5 0.0
6 0.0	7 0.0	8 0.0	9 0.0	10 0.0
11 0.0	12 0.0	13 0.0	14 0.0	15 0.0
16 0.0	17 0.0	18 0.0	19 0.0	20 0.0
21 0.0	22 0.0	23 0.0	24 0.0	25 0.0
26 0.111111	27 0.0	28 0.0	29 0.0	30 0.0
31 0.0	32 0.0	33 0.0	34 0.0	35 0.0
36 0.0	37 0.0	38 0.0	39 0.0	40 0.0
41 0.0	42 0.0	43 0.0	44 0.0	45 0.0
46 0.0	47 0.0	48 0.0	49 0.0	50 0.0
51 0.0	52 0.0	53 0.0	54 0.0	55 0.0
56 0.0	57 0.0	58 0.0	59 0.0	60 0.0
61 0.0	62 0.0	63 0.0	64 0.0	65 0.0
66 0.0	67 0.0	68 0.0	69 0.777778	

```
ROW NUMBER= 57
  1 0.0         2 0.034483    3 0.0          4 0.0         5 0.0
  6 0.0         7 0.0         8 0.0          9 0.0        10 0.034483
 11 0.0        12 0.0        13 0.0         14 0.0        15 0.0
 16 0.0        17 0.0        18 0.0         19 0.0        20 0.0
 21 0.0        22 0.0        23 0.0         24 0.0        25 0.0
 26 0.0        27 0.0        28 0.0         29 0.0        30 0.0
 31 0.0        32 0.0        33 0.0         34 0.0        35 0.0
 36 0.0        37 0.0        38 0.0         39 0.0        40 0.0
 41 0.0        42 0.0        43 0.0         44 0.0        45 0.0
 46 0.0        47 0.0        48 0.0         49 0.0        50 0.0
 51 0.0        52 0.0        53 0.0         54 0.0        55 0.0
 56 0.0        57 0.0        58 0.0         59 0.0        60 0.0
 61 0.0        62 0.0        63 0.0         64 0.0        65 0.0
 66 0.0        67 0.0        68 0.0         69 0.931034

ROW NUMBER= 58
  1 0.0         2 0.0         3 0.0          4 0.0         5 0.0
  6 0.0         7 0.0         8 0.0          9 0.0        10 0.0
 11 0.0        12 0.0        13 0.0         14 0.0        15 0.0
 16 0.0        17 0.0        18 0.0         19 0.0        20 0.0
 21 0.0        22 0.0        23 0.0         24 0.0        25 0.0
 26 0.0        27 0.0        28 0.0         29 0.0        30 0.0
 31 0.0        32 0.0        33 0.0         34 0.0        35 0.0
 36 0.0        37 0.0        38 0.0         39 0.0        40 0.0
 41 0.0        42 0.0        43 0.0         44 0.0        45 0.0
 46 0.0        47 0.0        48 0.0         49 0.0        50 0.0
 51 0.0        52 0.0        53 0.0         54 0.0        55 0.0
 56 1.000000   57 0.0        58 0.0         59 0.0        60 0.0
 61 0.0        62 0.0        63 0.0         64 0.0        65 0.0
 66 0.0        67 0.0        68 0.0         69 0.0

ROW NUMBER= 59
  1 0.0         2 0.0         3 0.0          4 0.0         5 0.0
  6 0.0         7 0.0         8 0.0          9 0.0        10 0.031250
 11 0.0        12 0.125000   13 0.093750    14 0.0        15 0.0
 16 0.0        17 0.0        18 0.031250    19 0.0        20 0.0
 21 0.0        22 0.0        23 0.0         24 0.0        25 0.0
 26 0.0        27 0.0        28 0.0         29 0.0        30 0.0
 31 0.0        32 0.0        33 0.0         34 0.0        35 0.0
 36 0.0        37 0.0        38 0.0         39 0.0        40 0.0
 41 0.0        42 0.0        43 0.0         44 0.0        45 0.0
 46 0.0        47 0.0        48 0.0         49 0.0        50 0.0
 51 0.0        52 0.0        53 0.0         54 0.0        55 0.0
 56 0.0        57 0.0        58 0.0         59 0.0        60 0.0
 61 0.0        62 0.093750   63 0.0         64 0.0        65 0.0
 66 0.0        67 0.0        68 0.0         69 0.625000

ROW NUMBER= 60
  1 0.0         2 0.0         3 0.0          4 0.400000    5 0.0
  6 0.0         7 0.0         8 0.0          9 0.0        10 0.0
 11 0.0        12 0.0        13 0.0         14 0.0        15 0.600000
 16 0.0        17 0.0        18 0.0         19 0.0        20 0.0
 21 0.0        22 0.0        23 0.0         24 0.0        25 0.0
 26 0.0        27 0.0        28 0.0         29 0.0        30 0.0
 31 0.0        32 0.0        33 0.0         34 0.0        35 0.0
 36 0.0        37 0.0        38 0.0         39 0.0        40 0.0
 41 0.0        42 0.0        43 0.0         44 0.0        45 0.0
 46 0.0        47 0.0        48 0.0         49 0.0        50 0.0
 51 0.0        52 0.0        53 0.0         54 0.0        55 0.0
 56 0.0        57 0.0        58 0.0         59 0.0        60 0.0
 61 0.0        62 0.0        63 0.0         64 0.0        65 0.0
 66 0.0        67 0.0        68 0.0         69 0.0
```

```
ROW NUMBER= 61
  1 0.0         2 0.0         3 0.0         4 0.0         5 0.0
  6 0.0         7 0.0         8 0.0         9 0.0        10 0.500000
 11 0.0        12 0.0        13 0.500000   14 0.0        15 0.0
 16 0.0        17 0.0        18 0.0        19 0.0        20 0.0
 21 0.0        22 0.0        23 0.0        24 0.0        25 0.0
 26 0.0        27 0.0        28 0.0        29 0.0        30 0.0
 31 0.0        32 0.0        33 0.0        34 0.0        35 0.0
 36 0.0        37 0.0        38 0.0        39 0.0        40 0.0
 41 0.0        42 0.0        43 0.0        44 0.0        45 0.0
 46 0.0        47 0.0        48 0.0        49 0.0        50 0.0
 51 0.0        52 0.0        53 0.0        54 0.0        55 0.0
 56 0.0        57 0.0        58 0.0        59 0.0        60 0.0
 61 0.0        62 0.0        63 0.0        64 0.0        65 0.0
 66 0.0        67 0.0        68 0.0        69 0.0

ROW NUMBER= 62
  1 0.0         2 0.0         3 0.0         4 0.0         5 0.0
  6 0.0         7 0.0         8 0.0         9 0.0        10 0.0
 11 0.0        12 0.083333   13 0.083333   14 0.0        15 0.0
 16 0.0        17 0.0        18 0.0        19 0.0        20 0.0
 21 0.0        22 0.0        23 0.0        24 0.0        25 0.0
 26 0.0        27 0.0        28 0.0        29 0.0        30 0.0
 31 0.0        32 0.0        33 0.083333   34 0.0        35 0.0
 36 0.0        37 0.0        38 0.0        39 0.0        40 0.0
 41 0.0        42 0.0        43 0.0        44 0.0        45 0.0
 46 0.0        47 0.0        48 0.0        49 0.0        50 0.0
 51 0.0        52 0.0        53 0.0        54 0.0        55 0.0
 56 0.0        57 0.0        58 0.0        59 0.333333   60 0.0
 61 0.0        62 0.0        63 0.0        64 0.0        65 0.0
 66 0.0        67 0.0        68 0.0        69 0.416667

ROW NUMBER= 63
  1 0.0         2 0.0         3 0.0         4 0.0         5 0.0
  6 0.0         7 0.0         8 0.0         9 0.0        10 0.0
 11 0.0        12 0.666667   13 0.333333   14 0.0        15 0.0
 16 0.0        17 0.0        18 0.0        19 0.0        20 0.0
 21 0.0        22 0.0        23 0.0        24 0.0        25 0.0
 26 0.0        27 0.0        28 0.0        29 0.0        30 0.0
 31 0.0        32 0.0        33 0.0        34 0.0        35 0.0
 36 0.0        37 0.0        38 0.0        39 0.0        40 0.0
 41 0.0        42 0.0        43 0.0        44 0.0        45 0.0
 46 0.0        47 0.0        48 0.0        49 0.0        50 0.0
 51 0.0        52 0.0        53 0.0        54 0.0        55 0.0
 56 0.0        57 0.0        58 0.0        59 0.0        60 0.0
 61 0.0        62 0.0        63 0.0        64 0.0        65 0.0
 66 0.0        67 0.0        68 0.0        69 0.0

ROW NUMBER= 64
  1 0.0         2 0.0         3 0.0         4 0.0         5 0.0
  6 0.0         7 0.0         8 1.000000    9 0.0        10 0.0
 11 0.0        12 0.0        13 0.0        14 0.0        15 0.0
 16 0.0        17 0.0        18 0.0        19 0.0        20 0.0
 21 0.0        22 0.0        23 0.0        24 0.0        25 0.0
 26 0.0        27 0.0        28 0.0        29 0.0        30 0.0
 31 0.0        32 0.0        33 0.0        34 0.0        35 0.0
 36 0.0        37 0.0        38 0.0        39 0.0        40 0.0
 41 0.0        42 0.0        43 0.0        44 0.0        45 0.0
```

46 0.0	47 0.0	48 0.0	49 0.0	50 0.0
51 0.0	52 0.0	53 0.0	54 0.0	55 0.0
56 0.0	57 0.0	58 0.0	59 0.0	60 0.0
61 0.0	62 0.0	63 0.0	64 0.0	65 0.0
66 0.0	67 0.0	68 0.0	69 0.0	

ROW NUMBER= 65

1 0.0	2 0.0	3 0.0	4 0.0	5 0.0
6 0.0	7 0.0	8 0.0	9 0.0	10 0.055556
11 0.0	12 0.0	13 0.0	14 0.0	15 0.0
16 0.0	17 0.0	18 0.0	19 0.0	20 0.0
21 0.0	22 0.0	23 0.0	24 0.0	25 0.0
26 0.0	27 0.0	28 0.0	29 0.0	30 0.0
31 0.0	32 0.0	33 0.0	34 0.0	35 0.0
36 0.0	37 0.0	38 0.0	39 0.0	40 0.0
41 0.0	42 0.0	43 0.0	44 0.0	45 0.0
46 0.0	47 0.0	48 0.0	49 0.0	50 0.0
51 0.0	52 0.0	53 0.0	54 0.0	55 0.0
56 0.0	57 0.0	58 0.0	59 0.0	60 0.0
61 0.0	62 0.0	63 0.0	64 0.0	65 0.0
66 0.0	67 0.0	68 0.0	69 0.944444	

ROW NUMBER= 66

1 0.0	2 0.0	3 0.0	4 0.0	5 0.0
6 0.0	7 0.0	8 0.0	9 0.0	10 0.0
11 0.0	12 0.0	13 0.0	14 0.0	15 0.0
16 0.0	17 0.0	18 0.0	19 0.0	20 0.0
21 0.0	22 0.0	23 0.0	24 0.0	25 0.0
26 0.500000	27 0.0	28 0.0	29 0.0	30 0.0
31 0.0	32 0.0	33 0.0	34 0.0	35 0.0
36 0.0	37 0.0	38 0.0	39 0.0	40 0.0
41 0.0	42 0.0	43 0.0	44 0.0	45 0.0
46 0.0	47 0.0	48 0.0	49 0.0	50 0.0
51 0.500000	52 0.0	53 0.0	54 0.0	55 0.0
56 0.0	57 0.0	58 0.0	59 0.0	60 0.0
61 0.0	62 0.0	63 0.0	64 0.0	65 0.0
66 0.0	67 0.0	68 0.0	69 0.0	

ROW NUMBER= 67

1 0.0	2 0.0	3 0.0	4 0.0	5 0.0
6 0.0	7 0.0	8 0.0	9 0.0	10 0.0
11 0.0	12 0.0	13 0.0	14 0.0	15 0.0
16 0.0	17 0.0	18 0.0	19 0.0	20 0.0
21 0.0	22 0.0	23 0.0	24 0.0	25 0.0
26 0.0	27 0.500000	28 0.0	29 0.0	30 0.0
31 0.0	32 0.0	33 0.0	34 0.0	35 0.0
36 0.0	37 0.0	38 0.0	39 0.0	40 0.0
41 0.0	42 0.0	43 0.0	44 0.0	45 0.0
46 0.0	47 0.0	48 0.0	49 0.0	50 0.0
51 0.0	52 0.0	53 0.0	54 0.0	55 0.0
56 0.0	57 0.0	58 0.0	59 0.0	60 0.0
61 0.0	62 0.0	63 0.0	64 0.0	65 0.0
66 0.0	67 0.0	68 0.0	69 0.500000	

ROW NUMBER= 68

1 0.0	2 1.000000	3 0.0	4 0.0	5 0.0
6 0.0	7 0.0	8 0.0	9 0.0	10 0.0
11 0.0	12 0.0	13 0.0	14 0.0	15 0.0
16 0.0	17 0.0	18 0.0	19 0.0	20 0.0
21 0.0	22 0.0	23 0.0	24 0.0	25 0.0

26 0.0	27 0.0	28 0.0	29 0.0	30 0.0
31 0.0	32 0.0	33 0.0	34 0.0	35 0.0
36 0.0	37 0.0	38 0.0	39 0.0	40 0.0
41 0.0	42 0.0	43 0.0	44 0.0	45 0.0
46 0.0	47 0.0	48 0.0	49 0.0	50 0.0
51 0.0	52 0.0	53 0.0	54 0.0	55 0.0
56 0.0	57 0.0	58 0.0	59 0.0	60 0.0
61 0.0	62 0.0	63 0.0	64 0.0	65 0.0
66 0.0	67 0.0	68 0.0	69 0.0	

ROW NUMBER= 69

1 0.0	2 0.0	3 0.0	4 0.0	5 0.0
6 0.0	7 0.0	8 0.0	9 0.0	10 0.0
11 0.0	12 0.0	13 0.0	14 0.0	15 0.0
16 0.0	17 0.0	18 0.0	19 0.0	20 0.0
21 0.0	22 0.0	23 0.0	24 0.0	25 0.0
26 0.0	27 0.0	28 0.0	29 0.0	30 0.0
31 0.0	32 0.0	33 0.0	34 0.0	35 0.0
36 0.0	37 0.0	38 0.0	39 0.0	40 0.0
41 0.0	42 0.0	43 0.0	44 0.0	45 0.0
46 0.0	47 0.0	48 0.0	49 0.0	50 0.0
51 0.0	52 0.0	53 0.0	54 0.0	55 0.0
56 0.0	57 0.0	58 0.0	59 0.0	60 0.0
61 0.0	62 0.0	63 0.0	64 0.0	65 0.0
66 0.0	67 0.0	68 0.0	69 0.0	

APPENDIX D

Distances Between Sidewinder Locations

ROCKET LOCATIONS

1	H5A	YORKTOWN VA
2	P5H	SUBIC BAY PR
3	ISRAE	ISRAEL
4	H5D	FALLBROOK CA
5	KITT	KITTY HAWK CV-63
6	GUAM	AGANA GUAM
7	INDE	INDEPENDENCE CV-62
8	CORA	CORAL SEA CV-43
9	JFK	JFK CV-67
10	MIDW	MIDWAY CV-41
11	CONS	CONSTELLATION CV-64
12	SUBIC	SUBIC BAY NAVMAG
13	CONC	CONCORD CA
14	YUMA	YUMA AZ
15	SEAL	SEAL BEACH CA
16	MIRA	MIRAMAR CA
17	DALL	DALLAS TX
18	RANG	RANGER CV-61
19	KANE	KANEOHE HI
20	ENTE	ENTERPRISE CVN-65
21	ORIS	ORISKANY CV-34
22	SHAS	SHASTA AE-33
23	ATSUG	ATSUGI JA
24	SBAR	SANTA BARBARA AE-28
25	SURI	SURIBACHI AE-21
26	OCEA	OCEANA VA
27	ELT	EL TORO CA
28	SARA	SARATOGA CV-60
29	BUTT	BUTTE AE-27
30	NIMI	NIMITZ CVN-68
31	FORR	FORRESTAL CV-59
32	ROOS	ROOSEVELT ROADS PR
33	NAHA	NAHA OKINAWA
34	KADE	KADENA
35	FLIN	FLINT AE-32
36	HULL	USS HULL
37	NMC	PT. MUGU CA
38	NAFS	SIGONELLA ITALY
39	BAKE	MT. BAKER AE-34
40	BEAU	BEAUFORT SC
41	AOR5	WABASH AOR-5
42	NOR	NORFOLK VA
43	SINGP	SINGAPORE
44	AMER	AMERICA CV-66
45	FDR	F. D. ROOSEVELT CV-4
46	DANG	DA NANG VIETNAM
47	DET	DETROIT AOE-4
48	VF-17	VF-17
49	NITR	NITRO AE-23
50	NELL	NELLIS AFB NV
51	VF-43	VF-43
52	ROTA	ROTA SPAIN
53	VF-10	VF-10
54	VF-11	VF-11
55	CHER	CHERRY PT. NC
56	EISEN	EISENHOWER CVN-69
57	KEY	KEY WEST FL
58	CANI	CANISTEO AO-99
59	IWAK	IWAKUNI JA
60	CAMD	CAMDEN AOE-2
61	HALE	HALEKALA AE-25
62	GA1	H & M 15 MAG
63	KISK	KISKA AE-35
64	D59	HANCOCK CV-19
65	NPG	NAM PHONG
66	CVSG-	CVSG
67	GF1	HCM 11 MAG
68	FSRPA	FORCE SERVICE REG 3
69		CAPTIVE FLIGHT

ROCKET LOCATION= 1

		FROM	TO	MILES
1	2	YORKTOWN VA	SUBIC BAY PR	9086
1	3	YORKTOWN VA	ISRAEL	5595
1	4	YORKTOWN VA	FALLBROOK CA	2694
1	6	YORKTOWN VA	AGANA GUAM	8338
1	7	YORKTOWN VA	INDEPENDENCE CV-62	
1	9	YORKTOWN VA	JFK CV-67	
1	10	YORKTOWN VA	MIDWAY CV-41	
1	11	YORKTOWN VA	CONSTELLATION CV-64	
1	13	YORKTOWN VA	CONCORD CA	2903
1	14	YORKTOWN VA	YUMA AZ	2506
1	15	YORKTOWN VA	SEAL BEACH CA	2700
1	16	YORKTOWN VA	MIRAMAR CA	2734
1	18	YORKTOWN VA	RANGER CV-61	
1	19	YORKTOWN VA	KANEOHE HI	5016
1	20	YORKTOWN VA	ENTERPRISE CVN-65	
1	24	YORKTOWN VA	SANTA BARBARA AE-28	
1	25	YORKTOWN VA	SURIBACHI AE-21	
1	26	YORKTOWN VA	OCEANA VA	49
1	28	YORKTOWN VA	SARATOGA CV-60	
1	29	YORKTOWN VA	BUTTE AE-27	
1	30	YORKTOWN VA	NIMITZ CVN-68	
1	31	YORKTOWN VA	FORRESTAL CV-59	
1	32	YORKTOWN VA	ROOSEVELT ROADS PR	1409
1	33	YORKTOWN VA	NAHA OKINAWA	8222
1	35	YORKTOWN VA	SIGONELLA ITALY	4593
1	39	YORKTOWN VA	MT. BAKER AE-34	
1	40	YORKTOWN VA	BEAUFORT SC	497
1	41	YORKTOWN VA	WABASH AOR-5	
1	43	YORKTOWN VA	SINGAPORE	10,509
1	44	YORKTOWN VA	AMERICA CV-66	
1	47	YORKTOWN VA	DETROIT AOE-4	
1	49	YORKTOWN VA	NITRO AE-23	
1	55	YORKTOWN VA	CHERRY PT. NC	224
1	57	YORKTOWN VA	KEY WEST FL	
1	59	YORKTOWN VA	IWAKUNI JA	1065
1	69	YORKTOWN VA	CAPTIVE FLIGHT	7959

ROCKET LOCATION= 2

		FROM	TO	MILES
2	1	SUBIC BAY PR	YORKTOWN VA	9086
2	3	SUBIC BAY PR	ISRAEL	6461
2	4	SUBIC BAY PR	FALLBROOK CA	7089
2	5	SUBIC BAY PR	KITTY HAWK CV-63	
2	8	SUBIC BAY PR	CORAL SEA CV-43	
2	10	SUBIC BAY PR	MIDWAY CV-41	
2	12	SUBIC BAY PR	SUBIC BAY NAVMAG	
2	13	SUBIC BAY PR	CONCORD CA	6183
2	15	SUBIC BAY PR	SEAL BEACH CA	
2	16	SUBIC BAY PR	MIRAMAR CA	7068
2	18	SUBIC BAY PR	RANGER CV-61	
2	21	SUBIC BAY PR	ORISKANY CV-34	
2	22	SUBIC BAY PR	SHASTA AE-33	
2	26	SUBIC BAY PR	OCEANA VA	9135
2	27	SUBIC BAY PR	EL TORO CA	7199
2	30	SUBIC BAY PR	NIMITZ CVN-68	
2	32	SUBIC BAY PR	ROOSEVELT ROADS PR	10,746
2	33	SUBIC BAY PR	NAHA OKINAWA	915
2	34	SUBIC BAY PR	KADENA	928
2	35	SUBIC BAY PR	FLINT AE-32	
2	44	SUBIC BAY PR	AMERICA CV-66	
2	59	SUBIC BAY PR	IWAKUNI JA	1538
2	69	SUBIC BAY PR	CAPTIVE FLIGHT	

ROCKET LOCATION= 3

		FROM	TO	MILES
3	1	ISRAEL	YORKTOWN VA	5595
3	8	ISRAEL	CORAL SEA CV-43	
3	16	ISRAEL	MIRAMAR CA	8085
3	69	ISRAEL	CAPTIVE FLIGHT	

ROCKET LOCATION= 4

		FROM	TO	MILES
4	1	FALLBROOK CA	YORKTOWN VA	2694
4	2	FALLBROOK CA	SUBIC BAY PR	7249
4	3	FALLBROOK CA	ISRAEL	
4	5	FALLBROOK CA	KITTY HAWK CV-63	
4	7	FALLBROOK CA	INDEPENDENCE CV-62	
4	8	FALLBROOK CA	CORAL SEA CV-43	
4	9	FALLBROOK CA	JFK CV-67	
4	10	FALLBROOK CA	MIDWAY CV-41	
4	11	FALLBROOK CA	CONSTELLATION CV-64	
4	13	FALLBROOK CA	CONCORD CA	506
4	15	FALLBROOK CA	SEAL BEACH CA	77
4	16	FALLBROOK CA	MIRAMAR CA	44
4	19	FALLBROOK CA	KANEOHE HI	2300
4	20	FALLBROOK CA	ENTERPRISE CVN-65	
4	21	FALLBROOK CA	ORISKANY CV-34	
4	27	FALLBROOK CA	EL TORO CA	59
4	37	FALLBROOK CA	PT. MUGU CA	165
4	38	FALLBROOK CA	SIGONELLA ITALY	7287
4	44	FALLBROOK CA	AMERICA CV-66	
4	59	FALLBROOK CA	IWAKUNI JA	6199
4	65	FALLBROOK CA	NAM PHONG	8148
4	68	FALLBROOK CA	FORCE SERVICE REG 3	
4	69	FALLBROOK CA	CAPTIVE FLIGHT	

ROCKET LOCATION= 5

		FROM	TO
5	1	KITTY HAWK CV-63	YORKTOWN VA
5	2	KITTY HAWK CV-63	SUBIC BAY PR
5	10	KITTY HAWK CV-63	MIDWAY CV-41
5	13	KITTY HAWK CV-63	CONCORD CA
5	17	KITTY HAWK CV-63	DALLAS TX
5	18	KITTY HAWK CV-63	RANGER CV-61
5	20	KITTY HAWK CV-63	ENTERPRISE CVN-65
5	30	KITTY HAWK CV-63	NIMITZ CVN-68
5	31	KITTY HAWK CV-63	FORRESTAL CV-59
5	38	KITTY HAWK CV-63	SIGONELLA ITALY
5	69	KITTY HAWK CV-63	CAPTIVE FLIGHT

ROCKET LOCATION= 6

		FROM	TO
6	7	AGANA GUAM	INDEPENDENCE CV-62
6	69	AGANA GUAM	CAPTIVE FLIGHT

ROCKET LOCATION= 7

		FROM	TO
7	1	INDEPENDENCE CV-62	YORKTOWN VA
7	4	INDEPENDENCE CV-62	FALLBROOK CA
7	8	INDEPENDENCE CV-62	CORAL SEA CV-43
7	16	INDEPENDENCE CV-62	MIRAMAR CA
7	20	INDEPENDENCE CV-62	ENTERPRISE CVN-65
7	28	INDEPENDENCE CV-62	SARATOGA CV-60
7	69	INDEPENDENCE CV-62	CAPTIVE FLIGHT

ROCKET LOCATION= 8

		FROM	TO
8	1	CORAL SEA CV-43	YORKTOWN VA
8	3	CORAL SEA CV-43	ISRAEL
8	4	CORAL SEA CV-43	FALLBROOK CA
8	5	CORAL SEA CV-43	KITTY HAWK CV-63
8	9	CORAL SEA CV-43	JFK CV-67
8	10	CORAL SEA CV-43	MIDWAY CV-41
8	11	CORAL SEA CV-43	CONSTELLATION CV-64
8	13	CORAL SEA CV-43	CONCORD CA
8	14	CORAL SEA CV-43	YUMA AZ
8	16	CORAL SEA CV-43	MIRAMAR CA
8	17	CORAL SEA CV-43	DALLAS TX
8	35	CORAL SEA CV-43	FLINT AE-32
8	44	CORAL SEA CV-43	AMERICA CV-66
8	59	CORAL SEA CV-43	IWAKUNI JA
8	69	CORAL SEA CV-43	CAPTIVE FLIGHT

ROCKET LOCATION= 9

		FROM	TO
9	1	JFK CV-67	YORKTOWN VA
9	10	JFK CV-67	MIDWAY CV-41
9	11	JFK CV-67	CONSTELLATION CV-64
9	13	JFK CV-67	CONCORD CA
9	20	JFK CV-67	ENTERPRISE CVN-65
9	26	JFK CV-67	OCEANA VA
9	39	JFK CV-67	MT. BAKER AE-34
9	52	JFK CV-67	ROTA SPAIN
9	69	JFK CV-67	CAPTIVE FLIGHT

ROCKET LOCATION=10

		FROM	TO
10	1	MIDWAY CV-41	YORKTOWN VA
10	2	MIDWAY CV-41	SUBIC BAY PR
10	4	MIDWAY CV-41	FALLBROOK CA
10	8	MIDWAY CV-41	CORAL SEA CV-43
10	9	MIDWAY CV-41	JFK CV-67
10	12	MIDWAY CV-41	SUBIC BAY NAVMAG
10	13	MIDWAY CV-41	CONCORD CA
10	14	MIDWAY CV-41	YUMA AZ
10	18	MIDWAY CV-41	RANGER CV-61
10	20	MIDWAY CV-41	ENTERPRISE CVN-65
10	21	MIDWAY CV-41	ORISKANY CV-34
10	23	MIDWAY CV-41	ATSUGI JA
10	28	MIDWAY CV-41	SARATOGA CV-60
10	33	MIDWAY CV-41	NAHA OKINAWA
10	40	MIDWAY CV-41	BEAUFORT SC
10	42	MIDWAY CV-41	NORFOLK VA
10	44	MIDWAY CV-41	AMERICA CV-66
10	46	MIDWAY CV-41	DA NANG VIETNAM
10	49	MIDWAY CV-41	NITRO AE-23
10	61	MIDWAY CV-41	HALEKALA AE-25
10	63	MIDWAY CV-41	KISKA AE-35
10	65	MIDWAY CV-41	NAM PHONG
10	69	MIDWAY CV-41	CAPTIVE FLIGHT

ROCKET LOCATION=11

		FROM	TO
11	1	CONSTELLATION CV-64	YORKTOWN VA
11	2	CONSTELLATION CV-64	SUBIC BAY PR
11	7	CONSTELLATION CV-64	INDEPENDENCE CV-62
11	8	CONSTELLATION CV-64	CORAL SEA CV-43
11	9	CONSTELLATION CV-64	JFK CV-67
11	12	CONSTELLATION CV-64	SUBIC BAY NAVMAG
11	13	CONSTELLATION CV-64	CONCORD CA
11	16	CONSTELLATION CV-64	MIRAMAR CA
11	20	CONSTELLATION CV-64	ENTERPRISE CVN-65
11	37	CONSTELLATION CV-64	PT. MUGU CA
11	69	CONSTELLATION CV-64	CAPTIVE FLIGHT

ROCKET LOCATION=12

		FROM	TO
12	1	SUBIC BAY NAVMAG	YORKTOWN VA
12	2	SUBIC BAY NAVMAG	SUBIC BAY PR
12	10	SUBIC BAY NAVMAG	MIDWAY CV-41
12	13	SUBIC BAY NAVMAG	CONCORD CA
12	16	SUBIC BAY NAVMAG	MIRAMAR CA
12	28	SUBIC BAY NAVMAG	SARATOGA CV-60
12	30	SUBIC BAY NAVMAG	NIMITZ CVN-68
12	36	SUBIC BAY NAVMAG	USS HULL
12	69	SUBIC BAY NAVMAG	CAPTIVE FLIGHT

ROCKET LOCATION=13

		FROM	TO	MILES
13	1	CONCORD CA	YORKTOWN VA	2903
13	2	CONCORD CA	SUBIC BAY PR	6343
13	3	CONCORD CA	ISRAEL	8253
13	4	CONCORD CA	FALLBROOK CA	506
13	5	CONCORD CA	KITTY HAWK CV-63	
13	8	CONCORD CA	CORAL SEA CV-43	
13	9	CONCORD CA	JFK CV-67	
13	10	CONCORD CA	MIDWAY CV-41	
13	11	CONCORD CA	CONSTELLATION CV-64	
13	12	CONCORD CA	SUBIC BAY NAVMAG	6343
13	14	CONCORD CA	YUMA AZ	654
13	16	CONCORD CA	MIRAMAR CA	502
13	18	CONCORD CA	RANGER CV-61	
13	19	CONCORD CA	KANEOHE HI	2133
13	20	CONCORD CA	ENTERPRISE CVN-65	
13	21	CONCORD CA	ORISKANY CV-34	
13	22	CONCORD CA	SHASTA AE-33	
13	26	CONCORD CA	OCEANA VA	2952
13	27	CONCORD CA	EL TORO CA	436
13	30	CONCORD CA	NIMITZ CVN-68	
13	32	CONCORD CA	ROOSEVELT ROADS PR	4403
13	33	CONCORD CA	NAHA OKINAWA	5319
13	38	CONCORD CA	SIGONELLA ITALY	7246
13	51	CONCORD CA	VF-43	
13	59	CONCORD CA	IWAKUNI JA	5056
13	61	CONCORD CA	HALEKALA AE-25	
13	69	CONCORD CA	CAPTIVE FLIGHT	

ROCKET LOCATION=14

		FROM	TO	MILES
14	2	YUMA AZ	SUBIC BAY PR	7390
14	23	YUMA AZ	ATSUGI JA	9078
14	69	YUMA AZ	CAPTIVE FLIGHT	

ROCKET LOCATION=15

		FROM	TO	MILES
15	1	SEAL BEACH CA	YORKTOWN VA	2700
15	2	SEAL BEACH CA	SUBIC BAY PR	7017
15	4	SEAL BEACH CA	FALLBROOK CA	77
15	5	SEAL BEACH CA	KITTY HAWK CV-63	
15	9	SEAL BEACH CA	JFK CV-67	
15	10	SEAL BEACH CA	MIDWAY CV-41	
15	11	SEAL BEACH CA	CONSTELLATION CV-64	
15	13	SEAL BEACH CA	CONCORD CA	418
15	47	SEAL BEACH CA	DETROIT AOE-4	
15	69	SEAL BEACH CA	CAPTIVE FLIGHT	

ROCKET LOCATION=16

		FROM	TO	MILES
16	1	MIRAMAR CA	YORKTOWN VA	2734
16	2	MIRAMAR CA	SUBIC BAY PR	7228
16	8	MIRAMAR CA	CORAL SEA CV-43	
16	10	MIRAMAR CA	MIDWAY CV-41	
16	11	MIRAMAR CA	CONSTELLATION CV-64	
16	12	MIRAMAR CA	SUBIC BAY NAVMAG	
16	13	MIRAMAR CA	CONCORD CA	502
16	14	MIRAMAR CA	YUMA AZ	180
16	15	MIRAMAR CA	SEAL BEACH CA	84
16	26	MIRAMAR CA	OCEANA VA	2703
16	45	MIRAMAR CA	F. D. ROOSEVELT CV-4	
16	69	MIRAMAR CA	CAPTIVE FLIGHT	

ROCKET LOCATION=17

		FROM	TO	MILES
17	59	DALLAS TX	IWAKUNI JA	7500
17	69	DALLAS TX	CAPTIVE FLIGHT	

ROCKET LOCATION=18

		FROM	TO
18	1	RANGER CV-61	YORKTOWN VA
18	2	RANGER CV-61	SUBIC BAY PR
18	8	RANGER CV-61	CORAL SEA CV-43
18	28	RANGER CV-61	SARATOGA CV-60
18	59	RANGER CV-61	IWAKUNI JA
18	62	RANGER CV-61	H & M 15 MAG
18	69	RANGER CV-61	CAPTIVE FLIGHT

ROCKET LOCATION=19

		FROM	TO
19	69	KANEOHE HI	CAPTIVE FLIGHT

ROCKET LOCATION=20

		FROM	TO
20	1	ENTERPRISE CVN-65	YORKTOWN VA
20	2	ENTERPRISE CVN-65	SUBIC BAY PR
20	4	ENTERPRISE CVN-65	FALLBROOK CA
20	5	ENTERPRISE CVN-65	KITTY HAWK CV-63
20	7	ENTERPRISE CVN-65	INDEPENDENCE CV-62
20	8	ENTERPRISE CVN-65	CORAL SEA CV-43
20	9	ENTERPRISE CVN-65	JFK CV-67
20	10	ENTERPRISE CVN-65	MIDWAY CV-41
20	11	ENTERPRISE CVN-65	CONSTELLATION CV-64
20	12	ENTERPRISE CVN-65	SUBIC BAY NAVMAG
20	13	ENTERPRISE CVN-65	CONCORD CA
20	15	ENTERPRISE CVN-65	SEAL BEACH CA
20	44	ENTERPRISE CVN-65	AMERICA CV-66
20	60	ENTERPRISE CVN-65	CAMDEN AOE-2
20	69	ENTERPRISE CVN-65	CAPTIVE FLIGHT

ROCKET LOCATION=21

		FROM	TO
21	2	ORISKANY CV-34	SUBIC BAY PR
21	9	ORISKANY CV-34	JFK CV-67
21	13	ORISKANY CV-34	CONCORD CA
21	22	ORISKANY CV-34	SHASTA AE-33
21	69	ORISKANY CV-34	CAPTIVE FLIGHT

ROCKET LOCATION=22

		FROM	TO
22	2	SHASTA AE-33	SUBIC BAY PR
22	13	SHASTA AE-33	CONCORD CA

ROCKET LOCATION=23

		FROM	TO	MILES
23	2	ATSUGI JA	SUBIC BAY PR	
23	37	ATSUGI JA	PT. MUGU CA	5508

ROCKET LOCATION=24

		FROM	TO
24	1	SANTA BARBARA AE-28	YORKTOWN VA
24	16	SANTA BARBARA AE-28	MIRAMAR CA
24	40	SANTA BARBARA AE-28	BEAUFORT SC
24	69	SANTA BARBARA AE-28	CAPTIVE FLIGHT

ROCKET LOCATION=25

		FROM	TO
25	1	SURIBACHI AE-21	YORKTOWN VA
25	10	SURIBACHI AE-21	MIDWAY CV-41
25	18	SURIBACHI AE-21	RANGER CV-61
25	69	SURIBACHI AE-21	CAPTIVE FLIGHT

ROCKET LOCATION=26

		FROM	TO	MILES
26	1	OCEANA VA	YORKTOWN VA	49
26	9	OCEANA VA	JFK CV-67	
26	13	OCEANA VA	CONCORD CA	2952
26	23	OCEANA VA	ATSUGI JA	7498
26	27	OCEANA VA	EL TORO CA	2796
26	28	OCEANA VA	SARATOGA CV-60	
26	30	OCEANA VA	NIMITZ CVN-68	
26	32	OCEANA VA	ROOSEVELT ROADS PR	1354
26	37	OCEANA VA	PT. MUGU CA	2816
26	44	OCEANA VA	AMERICA CV-66	
26	48	OCEANA VA	VF-17	
26	51	OCEANA VA	VF-43	
26	53	OCEANA VA	VF-10	
26	54	OCEANA VA	VF-11	
26	56	OCEANA VA	EISENHOWER CVN-69	
26	58	OCEANA VA	CANISTEO AO-99	
26	66	OCEANA VA	CVSG	
26	69	OCEANA VA	CAPTIVE FLIGHT	

ROCKET LOCATION=27

		FROM	TO	MILES
27	4	EL TORO CA	FALLBROOK CA	59
27	15	EL TORO CA	SEAL BEACH CA	30
27	69	EL TORO CA	CAPTIVE FLIGHT	

ROCKET LOCATION=28

		FROM	TO
28	1	SARATOGA CV-60	YORKTOWN VA
28	4	SARATOGA CV-60	FALLBROOK CA
28	7	SARATOGA CV-60	INDEPENDENCE CV-62
28	9	SARATOGA CV-60	JFK CV-67
28	13	SARATOGA CV-60	CONCORD CA
28	16	SARATOGA CV-60	MIRAMAR CA
28	20	SARATOGA CV-60	ENTERPRISE CVN-65
28	29	SARATOGA CV-60	BUTTE AE-27
28	69	SARATOGA CV-60	CAPTIVE FLIGHT

ROCKET LOCATION=29

		FROM	TO
29	1	BUTTE AE-27	YORKTOWN VA
29	20	BUTTE AE-27	ENTERPRISE CVN-65
29	30	BUTTE AE-27	NIMITZ CVN-68
29	41	BUTTE AE-27	WABASH AOR-5
29	69	BUTTE AE-27	CAPTIVE FLIGHT

ROCKET LOCATION=30

		FROM	TO
30	1	NIMITZ CVN-68	YORKTOWN VA
30	26	NIMITZ CVN-68	OCEANA VA
30	44	NIMITZ CVN-68	AMERICA CV-66
30	69	NIMITZ CVN-68	CAPTIVE FLIGHT

ROCKET LOCATION=31

		FROM	TO
31	26	FORRESTAL CV-59	OCEANA VA
31	69	FORRESTAL CV-59	CAPTIVE FLIGHT

ROCKET LOCATION=32

		FROM	TO	MILES
32	1	ROOSEVELT ROADS PR	YORKTOWN VA	1409
32	22	ROOSEVELT ROADS PR	SHASTA AE-33	
32	26	ROOSEVELT ROADS PR	OCEANA VA	1354
32	69	ROOSEVELT ROADS PR	CAPTIVE FLIGHT	

ROCKET LOCATION=33

		FROM	TO	MILES
33	1	NAHA OKINAWA	YORKTOWN VA	8222
33	13	NAHA OKINAWA	CONCORD CA	5319
33	26	NAHA OKINAWA	OCEANA VA	8271
33	34	NAHA OKINAWA	KADENA	13
33	69	NAHA OKINAWA	CAPTIVE FLIGHT	

ROCKET LOCATION=34

		FROM	TO	MILES
34	13	KADENA	CONCORD CA	5306

ROCKET LOCATION=35

		FROM	TO
35	10	FLINT AE-32	MIDWAY CV-41
35	12	FLINT AE-32	SUBIC BAY NAVMAG
35	36	FLINT AE-32	USS HULL

ROCKET LOCATION=36

		FROM	TO
36	2	USS HULL	SUBIC BAY PR
36	13	USS HULL	CONCORD CA

ROCKET LOCATION=37

		FROM	TO	MILES
37	16	PT. MUGU CA	MIRAMAR CA	202
37	26	PT. MUGU CA	OCEANA VA	2816

ROCKET LOCATION=38

		FROM	TO	MILES
38	1	SIGONELLA ITALY	YORKTOWN VA	4593
38	4	SIGONELLA ITALY	FALLBROOK CA	7287
38	39	SIGONELLA ITALY	MT. BAKER AE-34	

ROCKET LOCATION=39

		FROM	TO
39	1	MT. BAKER AE-34	YORKTOWN VA
39	28	MT. BAKER AE-34	SARATOGA CV-60
39	69	MT. BAKER AE-34	CAPTIVE FLIGHT

ROCKET LOCATION=40

		FROM	TO	MILES
40	1	BEAUFORT SC	YORKTOWN VA	497
40	12	BEAUFORT SC	SUBIC BAY NAVMAG	8992
40	55	BEAUFORT SC	CHERRY PT. NC	326
40	69	BEAUFORT SC	CAPTIVE FLIGHT	

ROCKET LOCATION=41

		FROM	TO
41	1	WABASH AOR-5	YORKTOWN VA

ROCKET LOCATION=42

		FROM	TO	MILES
42	3	NORFOLK VA	ISRAEL	5329

ROCKET LOCATION=43

		FROM	TO

ROCKET LOCATION=44

		FROM	TO
44	1	AMERICA CV-66	YORKTOWN VA
44	2	AMERICA CV-66	SUBIC BAY PR
44	5	AMERICA CV-66	KITTY HAWK CV-63
44	8	AMERICA CV-66	CORAL SEA CV-43
44	10	AMERICA CV-66	MIDWAY CV-41
44	20	AMERICA CV-66	ENTERPRISE CVN-65
44	28	AMERICA CV-66	SARATOGA CV-60
44	39	AMERICA CV-66	MT. BAKER AE-34
44	40	AMERICA CV-66	BEAUFORT SC
44	45	AMERICA CV-66	F. D. ROOSEVELT CV-4
44	47	AMERICA CV-66	DETROIT AOE-4
44	69	AMERICA CV-66	CAPTIVE FLIGHT

ROCKET LOCATION=45

		FROM	TO
45	13	F. D. ROOSEVELT CV-4	CONCORD CA
45	44	F. D. ROOSEVELT CV-4	AMERICA CV-66
45	69	F. D. ROOSEVELT CV-4	CAPTIVE FLIGHT

ROCKET LOCATION=46

		FROM	TO
46	10	DA NANG VIETNAM	MIDWAY CV-41
46	69	DA NANG VIETNAM	CAPTIVE FLIGHT

ROCKET LOCATION=47

		FROM	TO	
47	1	DETROIT AOE-4	YORKTOWN VA	
47	7	DETROIT AOE-4	INDEPENDENCE CV-62	
47	28	DETROIT AOE-4	SARATOGA CV-60	
47	44	DETROIT AOE-4	AMERICA CV-66	
47	69	DETROIT AOE-4	CAPTIVE FLIGHT	

ROCKET LOCATION=48

FROM TO

ROCKET LOCATION=49

		FROM	TO
49	1	NITRO AE-23	YORKTOWN VA
49	28	NITRO AE-23	SARATOGA CV-60
49	30	NITRO AE-23	NIMITZ CVN-68
49	32	NITRO AE-23	ROOSEVELT ROADS PR
49	50	NITRO AE-23	NELLIS AFB NV
49	69	NITRO AE-23	CAPTIVE FLIGHT

ROCKET LOCATION=50

		FROM	TO	MILES
50	1	NELLIS AFB NV	YORKTOWN VA	2558
50	69	NELLIS AFB NV	CAPTIVE FLIGHT	

ROCKET LOCATION=51

FROM TO

ROCKET LOCATION=52

FROM TO

ROCKET LOCATION=53

		FROM	TO
53	26	VF-10	OCEANA VA
53	69	VF-10	CAPTIVE FLIGHT

ROCKET LOCATION=54

FROM TO

ROCKET LOCATION=55

		FROM	TO	MILES
55	1	CHERRY PT. NC	YORKTOWN VA	224
55	14	CHERRY PT. NC	YUMA AZ	2477

ROCKET LOCATION=56

		FROM	TO
56	1	EISENHOWER CVN-69	YORKTOWN VA
56	26	EISENHOWER CVN-69	OCEANA VA
56	69	EISENHOWER CVN-69	CAPTIVE FLIGHT

ROCKET LOCATION=57

		FROM	TO	MILES
57	2	KEY WEST FL	SUBIC BAY PR	9873
57	10	KEY WEST FL	MIDWAY CV-41	
57	69	KEY WEST FL	CAPTIVE FLIGHT	

ROCKET LOCATION=58

		FROM	TO
58	56	CANISTEO AO-99	EISENHOWER CVN-69

ROCKET LOCATION=59

		FROM	TO	MILES
59	10	IWAKUNI JA	MIDWAY CV-41	
59	12	IWAKUNI JA	SUBIC BAY NAVMAG	1538
59	13	IWAKUNI JA	CONCORD CA	5056
59	18	IWAKUNI JA	RANGER CV-61	
59	62	IWAKUNI JA	H & M 15 MAG	
59	69	IWAKUNI JA	CAPTIVE FLIGHT	

ROCKET LOCATION=60

		FROM	TO
60	4	CAMDEN AOE-2	FALLBROOK CA
60	15	CAMDEN AOE-2	SEAL BEACH CA

ROCKET LOCATION=61

		FROM	TO
61	10	HALEKALA AE-25	MIDWAY CV-41
61	13	HALEKALA AE-25	CONCORD CA

ROCKET LOCATION=62

		FROM	TO
62	12	H & M 15 MAG	SUBIC BAY NAVMAG
62	13	H & M 15 MAG	CONCORD CA
62	32	H & M 15 MAG	NAHA OKINAWA
62	59	H & M 15 MAG	IWAKUNI JA
62	69	H & M 15 MAG	CAPTIVE FLIGHT

ROCKET LOCATION=63

		FROM	TO
63	12	KISKA AE-35	SUBIC BAY NAVMAG
63	13	KISKA AE-35	CONCORD CA

ROCKET LOCATION=64

		FROM	TO
64	8	HANCOCK CV-19	CORAL SEA CV-43

ROCKET LOCATION=65

		FROM	TO
65	10	NAM PHONG	MIDWAY CV-41
65	69	NAM PHONG	CAPTIVE FLIGHT

ROCKET LOCATION=66

		FROM	TO
66	26	CVSG	OCEANA VA
66	51	CVSG	VF-43

ROCKET LOCATION=67

		FROM	TO
67	27	HCM 11 MAG	EL TORO CA
67	69	HCM 11 MAG	CAPTIVE FLIGHT

ROCKET LOCATION=68

		FROM	TO
68	2	FORCE SERVICE REG 3	SUBIC BAY PR

APPENDIX E

Computer Programs

COMPUTER PROGRAMS

STORAGE

CAPTIVE.FLIGHT

MARKOV

WEATHER

LOGISTIC

AIRCARRY

DAMAGE.SORT

```
// JOB (A95$X5,516,0.25,10),'GEORGE DERBALIAN'
/*JOBPARM FORMS=1481
// EXEC FORTCLG
//FORT.SYSIN DD *
C// EXEC WATFIV
C//GO.SYSIN DD *
C$WATFIV
C *******************************************
C
C          STORAGE
C  GEORGE DERBALIAN
C  THIS PROGRAM COMPUTES ROCKET DAMAGE IN STORAGE LOCATIONS
C  APRIL 1981
C
C ***************************************
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /GREGW/W,WA
      COMMON/FAIL/AT,TEMP,DD,SU,SCR,XIO
      DIMENSION ITP(25),SHIFT(25),TITL(20),CDFA(200),CDFD(200),DMG(100)
      READ(5,500) TITL
      WRITE(6,600) TITL
  500 FORMAT(20A4)
  600 FORMAT('1',/20X,'*** ',20A4,' ***')
      READ(5,551) ITIME,SCFD,SCFA,XKT
      IF (XKT.EQ.0.0) XKT=1.0
      READ(5,501) EC,EP,PR,PRC,DIFF,RI,B,ALP,ALPC,H
  501 FORMAT(8F10.3)
      WRITE(6,601) EC,EP,PR,PRC,DIFF,RI,B,ALP,ALPC,H
  601 FORMAT(/1X,'MODULUS OF CASING',F15.2,1X,'PSI',/1X,
     $ 'MODULUS OF PROPELLANT ',F15.2,1X,'PSI',/1X,
     1 'POISSON''S RATIO OF PROPELLANT',F10.4,/1X,
     2 'POISSON''S RATIO OF CASE',F10.4,/1X,
     3 'THERMAL DIFFUSIVITY OF PROPELLANT ',E12.4,1X,'IN*IN/HR',/1X,
     $ 'INNER RADIUS OF PROPELLANT',F10.4,1X,'IN',/1X,
     4 'RADIUS OF PROPELLANT ',F10.4,1X,'IN',/1X,
     5 'COEFFICIENT OF THERMAL EXPANSION OF PROPELLANT ',E12.4,1X,'1/F',
     6 /1X,'COEFFICIENT OF THERMAL EXPANSION OF CASE ',E12.4,1X,'1/F'
     $ /1X,'THICKNESS OF CASING',F10.5,1X,'IN')
      WRITE(6,651) ITIME,SCFD,SCFA,XKT
  651 FORMAT(/1X,'TOTAL TIME=',I5,
     1 ' HRS',5X,'DIURNAL AMPLITUDE SCALE=',F10.4,5X,'SEASONAL',
     2 ' AMPLITUDE SCALE=',F10.4,5X,'KT=',F10.2)
  551 FORMAT(I5,3F10.3)
      READ(5,501) SU,SCR
      WRITE(6,614) SU,SCR
  614 FORMAT(/1X,'STRESS REQUIRED TO CAUSE FAILURE IN 1 MIN',E12.4,1X,'P
     $SI',5X,'CRITICAL STRESS BELOW WHICH NO FAILURES OC   ',E12.4,1X,
     $ 'PSI')
      READ(5,551) NAT
      READ(5,552) (ITP(I),SHIFT(I),I=1,NAT)
  552 FORMAT(I5,F10.2)
      WRITE(6,610)
  610 FORMAT(/1X,'TEMPERATURE F',5X,'SHIFT FACTOR')
      WRITE(6,611) (ITP(I),SHIFT(I),I=1,NAT)
  611 FORMAT(3X,I5,8X,F10.3)
C     DEFINE GLOBAL CONSTANTS
      W=0.2617994D0
      WA=W/365.0
      CALL CONST(RI,B,H,EP,PR,ALP,DIFF,EC,PRC,ALPC)
C INITIALIZE RANDOM NUMBER GENERATING VARIABLE ISEED
```

```
      ISEED=983145267
C
      IR=0
   77 IR=IR+1
      WRITE(6,658)
  658 FORMAT('1')
      DO 1 I=1,8
      READ(9,500,END=99) TITL
    1 WRITE(6,602) TITL
  602 FORMAT(1X,20A4)
      READ(9,551) IAMT
C REFERENCE STRESS FREE TEMPERATURE 75F
      AMT=IAMT-75
      WRITE(6,652) IAMT,IR
  652 FORMAT(I5,1X,'AVERAGE ANNUAL TEMPERATURE',5X,'LOCATION=',I5)
      READ(9,500) TITL
      READ(9,553) (CDFA(I),I=1,200)
  553 FORMAT(10X,F10.6)
      WRITE(6,654)
      DO 2 I=1,200
      XI=I
      X=0.5*(XI-1.)
    2 IF (I.LT.15) WRITE(6,653) X,CDFA(I)
  653 FORMAT(F10.2,F10.6)
  654 FORMAT(' SEASONAL TEMPERATURE AMPLITUDE C.D.F.')
      READ(9,500) TITL
      READ(9,553) (CDFD(I),I=1,200)
      WRITE(6,655)
  655 FORMAT(' DIURNAL TEMPERATURE AMPLITUDE C.D.F.')
      DO 3 I=1,200
      XI=I
      X=0.5*XI-0.5
    3 IF (I.LT.15) WRITE(6,653) X,CDFD(I)
C
      DMG(IR)=0.
      ATO=1.0D0
      XIO=0.0
C
      DO 8 I=1,ITIME
      TIME=I
      K=(I-1)/24
      K=K*24
      K=I-1-K
      IF (K.EQ.0) CALL RANDAY(SCFD,CDFD,TDL,ISEED)
      K=(I-1)/720
      K=K*720
      K=I-1-K
      IF (K.EQ.0) CALL RANSEA(SCFA,CDFA,TAL,ISEED)
      CALL TEMPER (TIME,AMT,TAL,TDL,TEMP)
      IF(TEMP+75.0.GT.ITP(1)) GO TO 10
      AT=SHIFT(1)+(TEMP+75.0-ITP(1))*(SHIFT(2)-SHIFT(1))/
     $ (ITP(2)-ITP(1))
      GO TO 11
   10 DO 12 II=2,NAT
      IF (ITP(II).LT.TEMP+75.) GO TO 12
      AT=SHIFT(II-1)+(TEMP+75.0-ITP(II-1))*(SHIFT(II)-SHIFT(II-1))/
     $ (ITP(II)-ITP(II-1))
      GO TO 11
   12 CONTINUE
      AT=SHIFT(NAT)+(TEMP+75.0-ITP(NAT))*(SHIFT(NAT)-SHIFT(NAT-1))/
```

```
       $ (ITP(NAT)-ITP(NAT-1))
    11 AT=DEXP(2.302585D0*AT)
       CALL STRESS(TIME,AMT,TAL,TDL,SH)
       DD=0.
       SH=SH*XKT
       CALL DAMAGE(SH)
       DMG(IR)=DD+DMG(IR)
       ATO=AT
     8 CONTINUE
C
       GO TO 77
COULD USE A SORTING ROUTINE HERE
    99 IR=IR-1
       WRITE(6,657)
   657 FORMAT('1',1X,'RELATIVE DAMAGE')
       WRITE(6,656) (I,DMG(I),I=1,IR)
   656 FORMAT(1X,'LOCATION=',I3,5X,'DAMAGE=',E12.5)
       STOP
       END
C
C****************************************************
C
       SUBROUTINE RANSEA(SCFA,CDFA,TAL,ISEED)
       IMPLICIT REAL*8(A-H,O-Z)
       REAL*8 CDFA(200)
       REAL*4 X
       CALL RANDK(ISEED,X,0)
       XX=X
       DO 4 I=1,200
       XI=I
     4 IF (CDFA(I).GT.XX) GO TO 5
     5 TAL=(0.5*XI-0.5)*SCFA
       RETURN
       END
C
C****************************************************
C
       SUBROUTINE RANDAY(SCFD,CDFD,TDL,ISEED)
       IMPLICIT REAL*8(A-H,O-Z)
       REAL*8 CDFD(200)
       REAL*4 X
       CALL RANDK(ISEED,X,0)
       DO 6 I=1,200
       XX=X
       XI=I
     6 IF(CDFD(I).GT.XX) GO TO 7
     7 TDL=SCFD*(0.5*XI-0.5)
       RETURN
       END
C
C****************************************************
C
       SUBROUTINE RANDK (IY, YFL, INDEX)
C
C      THIS IS A UNIFORM RANDOM NUMBER GENERATOR WRITTEN BY G. E.
C  FORSYTHE IN SPRING 1969, FOLLOWING D. KNUTH, THE ART OF COMPUTER
C  PROGRAMMING, VOL. 2, PP. 155-156.  IT IS MUCH SUPERIOR TO RANDU,
C  THE RANDOM NUMBER GENERATOR FOUND IN IBM'S SCIENTIFIC SUBROUTINE
C  PACKAGE.
C
```

```
C     BEFORE THE FIRST CALL OF RANDK, IY SHOULD BE SET OUTSIDE RANDK
C TO AN ARBITRARY INTEGER VALUE.  (IN WATFOR THIS IS ESSENTIAL.)
C FOR PROGRAM CHECKOUT, THE INITIAL VALUE OF IY SHOULD BE A FIXED
C INTEGER.  FOR RANDOM NUMBERS DIFFERENT ON EVERY RUN (AND HENCE
C NOT REPRODUCIBLE), DECLARE INTEGER CLOCK1 AND THEN INITIALIZE
C IY TO CLOCK1(4).
C
C     IF RANDK IS CALLED WITH AN INTEGER INDEX = 1, THEN THE OUTPUT
C VALUE OF IY IS A PSEUDORANDOM INTEGER UNIFORMLY DISTRIBUTED IN THE
C RANGE 0 <= IY < 2**31.
C
C     IF RANDK IS CALLED WITH INDEX = 0, THEN NOT ONLY IS IY PRODUCED,
C BUT ALSO (AT SOME EXTRA COST IN TIME) A FLOATING NUMBER YFL, UNI-
C FORMLY DISTRIBUTED IN THE INTERVAL 0.0 <= YFL < 1.0.
C
        IY = IY*314159269 + 453806245
      4 IF (IY .GE. 0) GO TO 6
C
C CAUTION:  THE STATEMENT LABEL 4 IS ESSENTIAL IN ORDER TO PREVENT
C CERTAIN COMPILERS (E.G., FORTRAN H WITH OPT 0) FROM PERFORMING
C UNWANTED "OPTIMIZATIONS."  IT SHOULD NOT BE REMOVED.
C
      5    IY = IY + 2147483647 + 1
C          STATEMENT 5 ADDS 2**31 TO NEGATIVE VALUES OF IY
C
      6 IF (INDEX) 7, 7, 8
C
      7    YFL = IY
           YFL = YFL*.4656613E-9
C
      8 RETURN
        END
C
C ******************************************
C
        SUBROUTINE DAMAGE(S)
        IMPLICIT REAL*8(A-H,O-Z)
        COMMON/FAIL/AT,TEMP,DD,SU,SCR,XIO
        IF (S-SCR .GT. 0.0) GO TO 1
        XIO=0.0
        RETURN
      1 CN=DLOG10(S-SCR)
        BB=9.3D0
        XI=((S-SCR)**BB)/AT
        DD=30.0D0*(XI+XIO)/(SU-SCR)**BB
        XIO=XI
        RETURN
        END
C
C ************************************************
C
        SUBROUTINE TEMPER(T,AMT,TAL,TDL,TEMP)
        IMPLICIT REAL*8(A-H,O-Z)
        COMMON /GREGT/ TCOSD,TSIND,TCOSY,TSINY
        COMMON /GREGW/ WDAY,WYEAR
        WT = WYEAR * T
        TEMP = TAL * (TCOSY*DCOS(WT) + TSINY*DSIN(WT))
        WT = WDAY * T
        TEMP = TEMP + TDL*(TCOSD*DCOS(WT)+TSIND*DSIN(WT))
        TEMP = TEMP + AMT
```

```
      RETURN
      END
      SUBROUTINE STRESS(T,AMT,TAL,TDL,SH)
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /GREGS/ SHCD,SHSD,SHCY,SHSY,SHCON
      COMMON /GREGW/ WDAY,WYEAR
      WT = WYEAR * T
      SH = TAL * (SHCY*DCOS(WT)+SHSY*DSIN(WT))
      WT = WDAY*T
      SH = SH + TDL*(SHCD*DCOS(WT)+SHSD*DSIN(WT))
      SH = SH + SHCON*AMT
      RETURN
      END
C
C ***********************************************
C
      SUBROUTINE CONST(RI,RO,H,EP,VP,ALP,DIFF,EC,VC,ALC)
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 INTC,INTS
      COMPLEX*16 BOA
      COMMON /GREG/ PR,E,F,DE,DF,C,D
      COMMON /GREGT/ TCOSD,TSIND,TCOSY,TSINY
      COMMON /GREGS/ SHCD,SHSD,SHCY,SHSY,SHCON
      COMMON /GREGW/ WDAY,WYEAR
      I = 1
      PDAY = DSQRT(WDAY/DIFF)
      PYEAR = DSQRT(WYEAR/DIFF)
      P = PDAY
   10 PA = P * RI
      CALL MMKEL1(PA,BER,BEI,XKER,XKEI,IER)
      BOA = DCMPLX(BEI,-BER)/DCMPLX(-XKEI,XKER)
      C = DREAL(BOA)
      D = DIMAG(BOA)
      PR = P * RO
      CALL EANDF
      A = -F/(E*E+F*F)
      B =  E/(E*E+F*F)
      PR = PA
      CALL EANDF
      TSINY = B*E - A*F
      TCOSY = A*E + B*F
      IF(I.EQ.2) GO TO 20
      TSIND = TSINY
      TCOSD = TCOSY
      I = 2
      P = PYEAR
      GO TO 10
   20 I = 1
      P = PDAY
      DENOM=(1.D0+VP)*((1.D0-2.D0*VP)*RO*RO+RI*RI)/(RO*RO-RI*RI)
     C + (1.D0-VC*VC)*RO*EP/(H*EC)
   30 PR = P * RO
      CALL DEADF
      Z1 = 2.D0*EP*ALP*(1.D0+VP)*RO/(P*(RO*RO-RI*RI))
      ZS = A*DE + B*DF
      ZC = A*DF - B*DE
      PS = (Z1*ZS - ALC*(1.D0+VC)*EP)/DENOM
      PC = Z1*ZC/DENOM
      FCON = ((Z1*P*(RO-RI)/RO)-ALC*(1.D0+VC)*EP)/DENOM
      Z1 = RO*RO/(RO*RO-RI*RI)
```

```
      PC = PC * Z1
      PS = PS * Z1
      PCON = 2.D0*Z1*PCON
      Z1 = ALP*EP*RO/(P*(1.D0-VP)*(RO*RO-RI*RI))
      INTC = Z1*ZC
      INTS = Z1*ZS
      INTCON = 2.D0*ALP*EP/((1.D0-VP)*(RO+RI))
      PR = P * RI
      CALL EANDF
      CALL DEADF
      Z1 = ALP*EP/(P*(1.D0-VP)*RI)
      SHCY = Z1 * (A*DF-B*DE)
      SHSY = Z1 * (A*DE + B*DF)
      Z1 = ALP*EP/(1.D0-VP)
      SHCON = INTCON - PCON - Z1
      SHCY = SHCY - Z1*(A*E+B*F)
      SHCY = SHCY + (INTC-PC)*2.D0
      SHSY = SHSY - Z1*(-A*F+B*E)
      SHSY = SHSY + (INTS-PS)*2.D0
      IF(I.EQ.2) RETURN
      I = 2
      P = PYEAR
      SHCD = SHCY
      SHSD = SHSY
      GO TO 30
      END
C
C *******************************************************************
C
      SUBROUTINE EANDF
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON /GREG/ PR,E,F,DE,DF,C,D
      CALL MMKELO(PR,BER,BEI,XKER,XKEI,IER)
      E = BER + C*XKER - D*XKEI
      F = BEI + C*XKEI + D*XKER
      RETURN
      END
      SUBROUTINE DEADF
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON /GREG/ PR,E,F,DE,DF,C,D
      CALL MMKELD(PR,BER,BEI,XKER,XKEI,IER)
      DE = BER + C*XKER - D*XKEI
      DF = BEI + C*XKEI + D*XKER
      RETURN
      END
C     SUBROUTINE MMKELO (X,BER,BEI,XKER,XKEI,IER)                       MML00010
C                                                                       MML00020
C-MMKELO--------D*------LIBRARY 1---------------------------------------MML00030
C                                                                       MML00040
C   FUNCTION            - EVALUATE THE KELVIN FUNCTIONS BER,BEI,KER AND MML00050
C                           KEI OF ORDER ZERO                           MML00060
C   USAGE               - CALL MMKELO(X,BER,BEI,XKER,XKEI,IER)          MML00070
C   PARAMETERS   X      - INPUT ARGUMENT. IF X IS NEGATIVE, A WARNING   MML00080
C                           ERROR IS PRODUCED AND VALUES OF POSITIVE    MML00090
C                           MACHINE INFINITY WILL BE RETURNED FOR XKER  MML00100
C                           AND XKEI.                                   MML00110
C                BER    - OUTPUT ARGUMENT                               MML00120
C                BEI    - OUTPUT ARGUMENT                               MML00130
C                XKER   - OUTPUT ARGUMENT RETURNED ONLY WHEN X IS       MML00140
C                           POSITIVE.                                   MML00150
```

```
C                 XKEI   - OUTPUT ARGUMENT RETURNED ONLY WHEN X IS          MML00160
C                           POSITIVE.                                       MML00170
C                 IER    - ERROR PARAMETER                                  MML00180
C                          TERMINAL ERROR = 128+N.                          MML00190
C                            N = 1 INDICATES THAT THE ABSOLUTE VALUE OF     MML00200
C                              X WAS GREATER THAN 119. BER AND BEI ARE      MML00210
C                              SET TO ZERO. IF X IS NON-NEGATIVE, XKER      MML00220
C                              AND XKEI ARE ALSO SET TO ZERO. OTHERWISE,    MML00230
C                              XKER AND XKEI ARE SET TO POSITIVE MACHINE    MML00240
C                              INFINITY.                                    MML00250
C                          WARNING ERROR = 32 + N                           MML00260
C                            N = 2 INDICATES THAT X IS NEGATIVE.            MML00270
C                              XKER AND XKEI WILL BE RETURNED AS            MML00280
C                              POSITIVE MACHINE INFINITY.                   MML00290
C   PRECISION            - DOUBLE                                           MML00300
C   REQD. IMSL ROUTINES  - UERTST                                           MML00310
C   LANGUAGE             - FORTRAN                                          MML00320
C-----------------------------------------------------------------------MML00330
C   LATEST REVISION      - APRIL 30,1975                                    MML00340
C                                                                           MML00350
      SUBROUTINE MMKELO (X,BER,BEI,XKER,XKEI,IER)                           MML00360
C                                                                           MML00370
      DIMENSION          C1(9),C2(9),C3(9),C4(9),E1(9),E2(9)                MML00380
      DOUBLE PRECISION   C1,C2,C3,C4,E1,E2,PIO8,RT2,XINF,                   MML00390
     *                   PI,EUL,TEN,ZERO,HALF,ONE,ARG,BER,BEI,B1,B2,B3,     MML00400
     *                   B4,CON,DC,DCM,DE,DS,DSM,DSQ,PIO2,R1,R2,S,SM,T,     MML00410
     *                   TM,TWOPI,X,XKER,XKEI,Z,ZI,ZIM,ZSQ,Z4,ZMAX          MML00420
C                                                                           MML00430
C                                  COEFFICIENTS FOR EVALUATION OF           MML00440
C                                  BER-SUB-ZERO(X) FOR X GREATER THAN       MML00450
C                                  0. AND LESS THAN OR EQUAL TO 10.         MML00460
C                                                                           MML00470
      DATA               C1(1)/5.160704650-5/,C1(2)/-4.89871257270-3/       MML00480
      DATA               C1(3)/.25977730007D0/,C1(4)/-7.2422567278207D0/MML00490
      DATA               C1(5)/93.859669297172600/                          MML00500
      DATA               C1(6)/-470.950279588996800/                        MML00510
      DATA               C1(7)/678.168402766309100/                         MML00520
      DATA               C1(8)/-156.249999999570100/                        MML00530
      DATA               C1(9)/.99999999999974D0/                           MML00540
C                                                                           MML00550
C                                  COEFFICIENTS FOR EVALUATION OF           MML00560
C                                  BEI-SUB-ZERO(X) FOR X GREATER THAN 0.MML00570
C                                  AND X LESS THAN OR EQUAL TO 10.          MML00580
C                                                                           MML00590
      DATA               C2(1)/4.49130000-6/,C2(2)/-5.4442431750-4/         MML00600
      DATA               C2(3)/3.84288282734D-2/                            MML00610
      DATA               C2(4)/-1.496334274974200/                          MML00620
      DATA               C2(5)/28.969033878649900/                          MML00630
      DATA               C2(6)/-240.280754944257400/                        MML00640
      DATA               C2(7)/678.168402776980700/                         MML00650
      DATA               C2(8)/-434.027777777747900/                        MML00660
      DATA               C2(9)/24.999999999999800/                          MML00670
C                                                                           MML00680
C                                  COEFFICIENTS FOR EVALUATION OF           MML00690
C                                  KEI-SUB-ZERO(X) FOR X GREATER THAN       MML00700
C                                  OR EQUAL TO ZERO AND X LESS THAN OR      MML00710
C                                  EQUAL TO 10.                             MML00720
C                                                                           MML00730
      DATA               C3(1)/1.543630470-5/,C3(2)/-1.80647778600-3/       MML00740
      DATA               C3(3)/.12220873821920D0/                           MML00750
```

```
      DATA          C3(4)/-4.5187459132639D0/                    MML00760
      DATA          C3(5)/81.9524771606200D0/                    MML00770
      DATA          C3(6)/-623.0136717405201D0/                  MML00780
      DATA          C3(7)/1548.4845196730992D0/                  MML00790
      DATA          C3(8)/-795.7175925924866D0/                  MML00800
      DATA          C3(9)/24.9999999999993D0/                    MML00810
C                                                                MML00820
C                          COEFFICIENTS FOR EVALUATION OF        MML00830
C                          KER-SUB-ZERO(X) FOR X GREATER THAN ORMML00840
C                          EQUAL TO ZERO AND X LESS THAN OR      MML00850
C                          EQUAL TO TEN                          MML00860
C                                                                MML00870
      DATA          C4(1)/1.2161109D-6/,C4(2)/-1.797627986D-4/   MML00880
      DATA          C4(3)/1.593801497050-2/                      MML00890
      DATA          C4(4)/-.806152902787600/                     MML00900
      DATA          C4(5)/21.2123451660231D0/                    MML00910
      DATA          C4(6)/-255.0971742710479D0/                  MML00920
      DATA          C4(7)/1153.8281852814561D0/                  MML00930
      DATA          C4(8)/-1412.8508391203636D0/                 MML00940
      DATA          C4(9)/234.375D0/                             MML00950
C                                                                MML00960
C                          COEFFICIENTS FOR EVALUATION OF        MML00970
C                          AUXILIARY FUNCTIONS FOR X GREATER     MML00980
C                          THAN 10.                              MML00990
C                                                                MML01000
      DATA          E1(1)/4.92D-8/,E1(2)/1.452D-7/,E1(3)/1.35D-8/ MML01010
      DATA          E1(4)/-1.6192D-6/,E1(5)/-1.12207D-5/         MML01020
      DATA          E1(6)/-5.17869D-5/,E1(7)/7.0D-10/            MML01030
      DATA          E1(8)/8.8388346D-3/,E1(9)/1.0D0/             MML01040
      DATA          E2(1)/-2.43D-8/,E2(2)/7.5D-8/,E2(3)/5.929D-7/ MML01050
      DATA          E2(4)/1.6431D-6/,E2(5)/-7.2D-9/              MML01060
      DATA          E2(6)/-5.18006D-5/,E2(7)/-7.031241D-4/       MML01070
      DATA          E2(8)/-8.8388340D-3/,E2(9)/0.0D0/            MML01080
C                                                                MML01090
C                          MISCELLANEOUS CONSTANTS               MML01100
C                                                                MML01110
      DATA          PIO2/1.5707963267948966D0/                   MML01120
      DATA          TWOPI/6.2831853071795860D0/                  MML01130
      DATA          PIO8/.39269908169872415D0/                   MML01140
      DATA          RT2/.70710678118654752D0/                    MML01150
      DATA          XINF/Z7FFFFFFFFFFFFFFF/                      MML01160
      DATA          PI/3.14159265358979320D0/                    MML01170
      DATA          EUL/.57721566490153286D0/                    MML01180
      DATA          TEN/10.D0/,ZERO/0.D0/,HALF/.5D0/,ONE/1.D0/   MML01190
      DATA          ZMAX/119.D0/                                 MML01200
      IER = 0                                                    MML01210
      Z = DABS(X)                                                MML01220
      IF (Z .GT. TEN) GO TO 15                                   MML01230
      IF (Z .EQ. ZERO) GO TO 10                                  MML01240
C                          CALCULATION OF FUNCTIONS FOR ABS(X)   MML01250
C                          LESS THAN 10.                         MML01260
      Z = Z/TEN                                                  MML01270
      ZSQ = Z*Z                                                  MML01280
      Z4 = ZSQ*ZSQ                                               MML01290
      B1 = C1(1)                                                 MML01300
      B2 = C2(1)                                                 MML01310
      B3 = C3(1)                                                 MML01320
      B4 = C4(1)                                                 MML01330
      DO 5 I = 2,9                                               MML01340
         B1 = B1*Z4+C1(I)                                        MML01350
```

```
            B2 = B2*Z4+C2(I)                                      MML01360
            B3 = B3*Z4+C3(I)                                      MML01370
            B4 = B4*Z4+C4(I)                                      MML01380
    5 CONTINUE                                                    MML01390
      BER = B1                                                    MML01400
      BEI = ZSQ*B2                                                MML01410
      IF (X .LT. ZERO) GO TO 30                                   MML01420
      R1 = ZSQ*B3                                                 MML01430
      R2 = Z4*B4                                                  MML01440
      CON = (DLOG(X*HALF)+EUL)                                    MML01450
      XKEI = -PIO2*HALF*BER+(R1-BEI*CON)                          MML01460
      XKER = PIO2*HALF*BEI-(R2+BER*CON)                           MML01470
      GO TO 9005                                                  MML01480
C                             X EQUAL 0. DEFAULT TO PROPER VALUES MML01490
   10 BER = ONE                                                   MML01500
      BEI = ZERO                                                  MML01510
      XKEI = -HALF*PIO2                                           MML01520
      XKER = XINF                                                 MML01530
      GO TO 9005                                                  MML01540
C                             X GREATER THAN 10. CALCULATE        MML01550
C                             AUXILIARY FUNCTIONS                 MML01560
   15 IF (Z .GT. ZMAX) GO TO 25                                   MML01570
      ZI = TEN/Z                                                  MML01580
      ZIM = -ZI                                                   MML01590
      S = E1(1)                                                   MML01600
      SM = S                                                      MML01610
      T = E2(1)                                                   MML01620
      TM = T                                                      MML01630
      DO 20 I = 2,9                                               MML01640
         S = S*ZI+E1(I)                                           MML01650
         T = T*ZI+E2(I)                                           MML01660
         SM = SM*ZIM+E1(I)                                        MML01670
         TM = TM*ZIM+E2(I)                                        MML01680
   20 CONTINUE                                                    MML01690
      ARG = Z*RT2                                                 MML01700
      DS = DSIN(ARG-PIO8)                                         MML01710
      DC = DCOS(ARG-PIO8)                                         MML01720
      DSM = DSIN(ARG+PIO8)                                        MML01730
      DCM = DCOS(ARG+PIO8)                                        MML01740
      DE = DEXP(ARG)                                              MML01750
      DSQ = DSQRT(TWOPI*Z)                                        MML01760
C                             CALCULATE THE DESIRED FUNCTIONS     MML01770
      BER = DE*(S*DC-T*DS)/DSQ                                    MML01780
      BEI = DE*(T*DC+S*DS)/DSQ                                    MML01790
      IF (X .LT. ZERO) GO TO 30                                   MML01800
      XKEI = PI*(TM*DCM-SM*DSM)/(DE*DSQ)                          MML01810
      XKER = PI*(SM*DCM+TM*DSM)/(DE*DSQ)                          MML01820
      GO TO 9005                                                  MML01830
C                             Z TOO LARGE.                        MML01840
   25 BER = ZERO                                                  MML01850
      BEI = ZERO                                                  MML01860
      IER = 129                                                   MML01870
      IF (X .LT. ZERO) GO TO 35                                   MML01880
      XKEI = ZERO                                                 MML01890
      XKER = ZERO                                                 MML01900
      GO TO 9000                                                  MML01910
C                             X LESS THAN 0. DEFAULT TO PROPER    MML01920
C                             VALUES                              MML01930
   30 IER = 34                                                    MML01940
   35 XKEI = XINF                                                 MML01950
```

```
      XKER = XINF                                                       MMLO1960
      IF (IER .EQ. 0) GO TO 9005                                        MMLO1970
 9000 CONTINUE                                                          MMLO1980
      CALL UERTST (IER,6HMMKELO)                                        MMLO1990
 9005 RETURN                                                            MMLO2000
      END                                                               MMLO2010
C     SUBROUTINE MMKELD (X,BERP,BEIP,XKERP,XKEIP,IER)                   MMLL0010
C                                                                       MMLL0020
C-MMKELD--------D-------LIBRARY 1---------------------------------------MMLL0030
C                                                                       MMLL0040
C   FUNCTION            - EVALUATE THE DERIVATIVES OF THE KELVIN        MMLL0050
C                           FUNCTIONS (BER,BEI,KER AND KEI) OF ORDER    MMLL0060
C                           ZERO.                                       MMLL0070
C   USAGE               - CALL MMKELD(X,BERP,BEIP,XKERP,XKEIP,IER)      MMLL0080
C   PARAMETERS   X      - INPUT ARGUMENT. IF X IS NEGATIVE, A WARNING   MMLL0090
C                           ERROR IS PRODUCED AND VALUES OF POSITIVE    MMLL0100
C                           MACHINE INFINITY WILL BE RETURNED FOR XKERP MMLL0110
C                           AND KEIP.                                   MMLL0120
C                BERP   - OUTPUT ARGUMENT                               MMLL0130
C                BEIP   - OUTPUT ARGUMENT                               MMLL0140
C                XKERP  - OUTPUT ARGUMENT RETURNED ONLY WHEN X IS       MMLL0150
C                           POSITIVE.                                   MMLL0160
C                XKEIP  - OUTPUT ARGUMENT RETURNED ONLY WHEN X IS       MMLL0170
C                           POSITIVE.                                   MMLL0180
C                IER    - ERROR PARAMETER.                              MMLL0190
C                         TERMINAL ERROR = 128+N.                       MMLL0200
C                           N = 1 INDICATES THAT THE ABSOLUTE VALUE OF  MMLL0210
C                             X WAS GREATER THAN 119. BERP AND BEIP ARE MMLL0220
C                             SET TO ZERO. IF X IS NON-NEGATIVE, XKERP  MMLL0230
C                             AND XKEIP ARE ALSO SET TO ZERO. OTHERWISE,MMLL0240
C                             XKERP AND XKEIP ARE SET TO POSITIVE       MMLL0250
C                             MACHINE INFINITY.                         MMLL0260
C                         WARNING ERROR = 32+N.                         MMLL0270
C                           N = 2 INDICATES THAT X IS NEGATIVE.         MMLL0280
C                             XKERP AND XKEIP WILL BE RETURNED AS       MMLL0290
C                             POSITIVE MACHINE INFINITY.                MMLL0300
C   PRECISION           - DOUBLE                                        MMLL0310
C   REQD. IMSL ROUTINES - MMKELO,UERTST                                 MMLL0320
C   LANGUAGE            - FORTRAN                                       MMLL0330
C-----------------------------------------------------------------------MMLL0340
C   LATEST REVISION     - SEPTEMBER 22,1976                             MMLL0350
C                                                                       MMLL0360
      SUBROUTINE MMKELD(X,BERP,BEIP,XKERP,XKEIP,IER)                    MMLL0370
C                                                                       MMLL0380
      DIMENSION          D1(9),D2(9),D3(9),D4(9),E3(9),E4(9)            MMLL0390
      DOUBLE PRECISION   ARG,BEI,BEIP,BER,BERP,B1,B2,B3,B4,CON,DC,DCM,  MMLL0400
     *                   DE,DS,DSM,DSQ,D1,D2,D3,D4,EUL,E3,E4,PI,PIO2,   MMLL0410
     *                   PIO8,RT2,R1P,R2P,TWOPI,U,UM,V,VM,X,XINF,XKEI,  MMLL0420
     *                   XKEIP,XKER,XKERP,Z,ZI,ZIM,ZSQ,Z3,Z4,ZMAX       MMLL0430
      DOUBLE PRECISION   TEN,ZERO,HALF                                  MMLL0440
      DATA               TEN/10.D0/,ZERO/0.D0/,HALF/.5D0/               MMLL0450
C                                                                       MMLL0460
C                               COEFFICIENTS FOR EVALUATION OF BERP-    MMLL0470
C                               SUB-ZERO(X) FOR X GREATER THAN 0. ANDMMLL0480
C                               LESS THAN OR EQUAL TO 10.               MMLL0490
C                                                                       MMLL0500
      DATA               D1(1)/-1.250604 6D-6/,D1(2)/1.701453451D-4/    MMLL0510
      DATA               D1(3)/-1.372460361 90D-2/                      MMLL0520
      DATA               D1(4)/.623472634824 3D0/                       MMLL0530
      DATA               D1(5)/-14.484516949840 3D0/                    MMLL0540
```

```
      DATA     D1(6)/150.1754718432278D0/                                MMLL0550
      DATA     D1(7)/-565.1403356479486D0/                               MMLL0560
      DATA     D1(8)/542.5347222222147D0/                                MMLL0570
      DATA     D1(9)/-62.4999999999999D0/                                MMLL0580
C                                                                        MMLL0590
C                            COEFFICIENTS FOR EVALUATION OF BEIP-        MMLL0600
C                            SUB-ZERO(X) FOR X GREATER THAN 0.           MMLL0610
C                            AND LESS THAN OR EQUAL TO 10.               MMLL0620
C                                                                        MMLL0630
      DATA     D2(1)/1.52269884D-5/,D2(2)/-1.6331100837D-3/              MMLL0640
      DATA     D2(3)/9.991470649320-2/                                   MMLL0650
      DATA     D2(4)/-3.2919352108579D0/                                 MMLL0660
      DATA     D2(5)/52.1442608975905D0/                                 MMLL0670
      DATA     D2(6)/-336.3930569023651D0/                               MMLL0680
      DATA     D2(7)/678.1684027747539D0/                                MMLL0690
      DATA     D2(8)/-260.4166666665533D0/                               MMLL0700
      DATA     D2(9)/4.9999999999993D0/                                  MMLL0710
C                                                                        MMLL0720
C                            COEFFICIENTS FOR EVALUTION OF KEIP-         MMLL0730
C                            SUB-ZERO(X) FOR X GREATER THAN 0.           MMLL0740
C                            AND LESS THAN OR EQUAL TO 10.               MMLL0750
C                                                                        MMLL0760
      DATA     D3(1)/5.23294314D-5/                                      MMLL0770
      DATA     D3(2)/-5.4188558408D-3/                                   MMLL0780
      DATA     D3(3)/.3177418434686D0/                                   MMLL0790
      DATA     D3(4)/-9.9412403209725D0/                                 MMLL0800
      DATA     D3(5)/147.5144585913337D0/                                MMLL0810
      DATA     D3(6)/-872.2191403672455D0/                               MMLL0820
      DATA     D3(7)/1548.4845196652035D0/                               MMLL0830
      DATA     D3(8)/-477.4305555551536D0/                               MMLL0840
      DATA     D3(9)/4.9999999999975D0/                                  MMLL0850
C                                                                        MMLL0860
C                            COEFFICIENTS FOR EVALUATION OF KERP-        MMLL0870
C                            SUB-ZERO(X) FOR X GREATER THAN OR           MMLL0880
C                            EQUAL TO 0. AND LESS THAN OR EQUAL          MMLL0890
C                            TO 10.                                      MMLL0900
C                                                                        MMLL0910
      DATA     D4(1)/4.3682053D-6/,D4(2)/-5.7520422830-4/                MMLL0920
      DATA     D4(3)/4.462638621450-2/                                   MMLL0930
      DATA     D4(4)/-1.9347669229237D0/                                 MMLL0940
      DATA     D4(5)/42.4246903131088D0/                                 MMLL0950
      DATA     D4(6)/-408.1554788292578D0/                               MMLL0960
      DATA     D4(7)/1384.5938223372452D0/                               MMLL0970
      DATA     D4(8)/-1130.2806712962694D0/                              MMLL0980
      DATA     D4(9)/93.7499999999998D0/                                 MMLL0990
C                                                                        MMLL1000
C                                                                        MMLL1010
C                            COEFFICIENTS FOR EVALUTION OF               MMLL1020
C                            AUXILIARY FUNCTIONS FOR X GREATER           MMLL1030
C                            THAN 10.                                    MMLL1040
C                                                                        MMLL1050
      DATA     E3(1)/-5.63D-8/,E3(2)/-1.671D-7/                          MMLL1060
      DATA     E3(3)/-1.47D-8/,E3(4)/1.9780D-6/                          MMLL1070
      DATA     E3(5)/1.44255D-5/,E3(6)/7.25024D-5/                       MMLL1080
      DATA     E3(7)/-8.0D-10/,E3(8)/-2.65165040D-2/                     MMLL1090
      DATA     E3(9)/1.0D0/                                              MMLL1100
      DATA     E4(1)/-2.69D-8/,E4(2)/-8.83D-8/                           MMLL1110
      DATA     E4(3)/-6.992D-7/,E4(4)/-2.0042D-6/                        MMLL1120
      DATA     E4(5)/7.9D-9/,E4(6)/7.25179D-5/                           MMLL1130
      DATA     E4(7)/1.1718740D-3/,E4(8)/2.65165034D-2/                  MMLL1140
```

```
      DATA             E4(9)/0.0D0/                                     MMLL1150
C                                                                       MMLL1160
C                                MISCELLANEOUS CONSTANTS                MMLL1170
C                                                                       MMLL1180
      DATA             PIO2/1.5707963267948966D0/                       MMLL1190
      DATA             TWOPI/6.2831853071795860D0/                      MMLL1200
      DATA             PIO8/0.39269908169872415D0/                      MMLL1210
      DATA             RT2/0.70710678118654752D0/                       MMLL1220
      DATA             XINF/Z7FFFFFFFFFFFFFFF/                          MMLL1230
      DATA             PI/3.141592653589793D0/                          MMLL1240
      DATA             EUL/0.5772156649015328 6D0/                      MMLL1250
      DATA             ZMAX/119.D0/                                     MMLL1260
      IER = 0                                                           MMLL1270
      CALL MMKELO(X,BER,BEI,XKER,XKEI,IER)                              MMLL1280
      Z = DABS(X)                                                       MMLL1290
      IF (Z .GT. TEN) GO TO 15                                          MMLL1300
      IF (Z .EQ. ZERO) GO TO 10                                         MMLL1310
C                                CALCULATION OF FUNCTIONS FOR ABS(X)    MMLL1320
C                                LESS THAN 10.                          MMLL1330
      Z = Z/TEN                                                         MMLL1340
      ZSQ = Z*Z                                                         MMLL1350
      Z3 = ZSQ*Z                                                        MMLL1360
      Z4 = ZSQ*ZSQ                                                      MMLL1370
      B1 = D1(1)                                                        MMLL1380
      B2 = D2(1)                                                        MMLL1390
      B3 = D3(1)                                                        MMLL1400
      B4 = D4(1)                                                        MMLL1410
      DO 5 I = 2,9                                                      MMLL1420
         B1 = B1*Z4+D1(I)                                               MMLL1430
         B2 = B2*Z4+D2(I)                                               MMLL1440
         B3 = B3*Z4+D3(I)                                               MMLL1450
         B4 = B4*Z4+D4(I)                                               MMLL1460
    5 CONTINUE                                                          MMLL1470
      BERP = B1*Z3                                                      MMLL1480
      BEIP = Z*B2                                                       MMLL1490
      IF ( X .LT. ZERO) GO TO 30                                        MMLL1500
      R1P = Z*B3                                                        MMLL1510
      R2P = Z3*B4                                                       MMLL1520
      CON = (DLOG(X*HALF) + EUL)                                        MMLL1530
      V = DABS(X)                                                       MMLL1540
      XKEIP = -PIO2*HALF*BERP+(R1P-BEIP*CON-BEI/V)                      MMLL1550
      XKERP = PIO2*HALF*BEIP-(R2P+BERP*CON+BER/V)                       MMLL1560
      GO TO 9005                                                        MMLL1570
C                                X EQUAL TO 0. DEFAULT TO PROPER        MMLL1580
C                                VALUES                                 MMLL1590
   10 BERP = ZERO                                                       MMLL1600
      BEIP = ZERO                                                       MMLL1610
      XKEIP = ZERO                                                      MMLL1620
      XKERP = -XINF                                                     MMLL1630
      GO TO 9005                                                        MMLL1640
C                                X GREATER THAN 10. CALCULATE           MMLL1650
C                                AUXILIARY FUNCTIONS                    MMLL1660
   15 IF (Z .GT. ZMAX) GO TO 25                                         MMLL1670
      ZI = TEN/Z                                                        MMLL1680
      ZIM = -ZI                                                         MMLL1690
      U = E3(1)                                                         MMLL1700
      UM = U                                                            MMLL1710
      V = E4(1)                                                         MMLL1720
      VM = V                                                            MMLL1730
      DO 20 I = 2,9                                                     MMLL1740
```

```
         U = U*ZI+E3(I)                                          MMLL1750
         V = V*ZI+E4(I)                                          MMLL1760
         UM = UM*ZIM+E3(I)                                       MMLL1770
         VM = VM*ZIM+E4(I)                                       MMLL1780
   20 CONTINUE                                                   MMLL1790
      ARG = Z*RT2                                                MMLL1800
      DS = DSIN(ARG-PIO8)                                        MMLL1810
      DC = DCOS(ARG-PIO8)                                        MMLL1820
      DSM = DSIN(ARG+PIO8)                                       MMLL1830
      DCM = DCOS(ARG+PIO8)                                       MMLL1840
      DE = DEXP(ARG)                                             MMLL1850
      DSQ = DSQRT(TWOPI*Z)                                       MMLL1860
C                                  CALCULATE THE DESIRED FUNCTIONS MMLL1870
      BERP = DE*(U*DCM-V*DSM)/DSQ                                MMLL1880
      BEIP = DE*(V*DCM+U*DSM)/DSQ                                MMLL1890
      IF (X .LT. ZERO) GO TO 30                                  MMLL1900
      XKEIP = -PI*(VM*DC-UM*DS)/(DE*DSQ)                         MMLL1910
      XKERP = -PI*(UM*DC+VM*DS)/(DE*DSQ)                         MMLL1920
      GO TO 9005                                                 MMLL1930
C                                  Z TOO LARGE.                  MMLL1940
   25 BERP = ZERO                                                MMLL1950
      BEIP = ZERO                                                MMLL1960
      IER = 129                                                  MMLL1970
      IF (X .LT. ZERO) GO TO 35                                  MMLL1980
      XKEIP = ZERO                                               MMLL1990
      XKERP = ZERO                                               MMLL2000
      GO TO 9000                                                 MMLL2010
C                                  X LESS THAN 0. DEFAULT TO PROPER MMLL2020
C                                  VALUES                        MMLL2030
   30 IER = 34                                                   MMLL2040
      BERP = -BERP                                               MMLL2050
      BEIP = -BEIP                                               MMLL2060
   35 XKERP = XINF                                               MMLL2070
      XKEIP = XINF                                               MMLL2080
      IF (IER .EQ. 0) GO TO 9005                                 MMLL2090
 9000 CONTINUE                                                   MMLL2100
      CALL UERTST(IER,6HMMKELD)                                  MMLL2110
 9005 RETURN                                                     MMLL2120
      END                                                        MMLL2130
C     SUBROUTINE UERTST (IER,NAME)                               UERT0010
C                                                                UERT0020
C-UERTST----------------LIBRARY 1--------------------------------UERT0030
C                                                                UERT0040
C   FUNCTION            - ERROR MESSAGE GENERATION               UERT0050
C   USAGE               - CALL UERTST(IER,NAME)                  UERT0060
C   PARAMETERS   IER    - ERROR PARAMETER. TYPE + N  WHERE       UERT0070
C                           TYPE= 128 IMPLIES TERMINAL ERROR     UERT0080
C                                  64 IMPLIES WARNING WITH FIX   UERT0090
C                                  32 IMPLIES WARNING            UERT0100
C                           N   = ERROR CODE RELEVANT TO CALLING ROUTINEUERT0110
C                NAME   - INPUT VECTOR CONTAINING THE NAME OF THE  UERT0120
C                           CALLING ROUTINE AS A SIX CHARACTER LITERAL UERT0130
C                           STRING.                              UERT0140
C   LANGUAGE            - FORTRAN                                UERT0150
C----------------------------------------------------------------UERT0160
C   LATEST REVISION     - JANUARY 18, 1974                       UERT0170
C                                                                UERT0180
      SUBROUTINE UERTST(IER,NAME)                                UERT0190
C                                                                UERT0200
      DIMENSION        ITYP(5,4),IBIT(4)                         UERT0210
```

```
      INTEGER*2          NAME(3)                                        UERT0220
      INTEGER            WARN,WARF,TERM,PRINTR                          UERT0230
      EQUIVALENCE        (IBIT(1),WARN),(IBIT(2),WARF),(IBIT(3),TERM)   UERT0240
      DATA     ITYP      /'WARN','ING ',' ',' ',' ',                   UERT0250
     *                    'WARN','ING(','WITH',' FIX',')   ',           UERT0260
     *                    'TERM','INAL',' ',' ',' ',                    UERT0270
     *                    'NON-','DEFI','NED ',' ',' '/,                UERT0280
     *         IBIT      / 32,64,128,0/                                 UERT0290
      DATA     PRINTR    / 6/                                           UERT0300
      IER2=IER                                                          UERT0310
      IF (IER2 .GE. WARN) GO TO 5                                       UERT0320
C                                  NON-DEFINED                          UERT0330
      IER1=4                                                            UERT0340
      GO TO 20                                                          UERT0350
    5 IF (IER2 .LT. TERM) GO TO 10                                      UERT0360
C                                  TERMINAL                             UERT0370
      IER1=3                                                            UERT0380
      GO TO 20                                                          UERT0390
   10 IF (IER2 .LT. WARF) GO TO 15                                      UERT0400
C                                  WARNING(WITH FIX)                    UERT0410
      IER1=2                                                            UERT0420
      GO TO 20                                                          UERT0430
C                                  WARNING                              UERT0440
   15 IER1=1                                                            UERT0450
C                                  EXTRACT 'N'                          UERT0460
   20 IER2=IER2-IBIT(IER1)                                              UERT0470
C                                  PRINT ERROR MESSAGE                  UERT0480
      WRITE (PRINTR,25) (ITYP(I,IER1),I=1,5),NAME,IER2,IER              UERT0490
   25 FORMAT(' *** I M S L(UERTST) ***  ',5A4,4X,3A2,4X,I2,             UERT0500
     *   ' (IER = ',I3,')')                                             UERT0510
      RETURN                                                            UERT0520
      END                                                               UERT0530
      SUBROUTINE MMKEL1 (X,BER1,BEI1,XKER1,XKEI1,IER)                   MML10360
C                                                                       MML10370
      DOUBLE PRECISION   BER1,BEI1,BERP,BEIP,RT2,X,XINF,XKEIP,XKEI1,    MML10380
     *                   XKERP,XKER1,ZERO,ZMAX                          MML10390
      DATA               XINF/Z7FFFFFFFFFFFFFFF/,ZERO/0.D0/             MML10400
      DATA               RT2/0.70710678118654752D0/                     MML10410
      DATA               ZMAX/119.D0/                                   MML10420
      IER = 0                                                           MML10430
      IF (X .EQ. ZERO) GO TO 15                                         MML10440
      IF (DABS(X) .GT. ZMAX) GO TO 10                                   MML10450
      CALL MMKELD(X,BERP,BEIP,XKERP,XKEIP,IER)                          MML10460
      BEI1 = (BERP+BEIP) *RT2                                           MML10470
      BER1 = (BERP - BEIP) * RT2                                        MML10480
      IF (X .LT. ZERO) GO TO 5                                          MML10490
      XKEI1 = (XKERP + XKEIP) * RT2                                     MML10500
      XKER1 = (XKERP - XKEIP) * RT2                                     MML10510
      GO TO 9005                                                        MML10520
C                                  ARGUMENT IS NEGATIVE                 MML10530
    5 XKER1 = XINF                                                      MML10540
      XKEI1 = XINF                                                      MML10550
      IER = 34                                                          MML10560
      GO TO 9000                                                        MML10570
   10 BEI1 = ZERO                                                       MML10580
      BER1 = ZERO                                                       MML10590
      XKER1 = ZERO                                                      MML10600
      XKEI1 = ZERO                                                      MML10610
      IER = 129                                                         MML10620
      IF (X .GT. ZERO) GO TO 9000                                       MML10630
```

```
         XKER1 = XINF                                                   MML10640
         XKEI1 = XINF                                                   MML10650
         GO TO 9000                                                     MML10660
C                                        ARGUMENT IS 0.0                MML10670
      15 BEI1 = ZERO                                                    MML10680
         BER1 = ZERO                                                    MML10690
         XKER1 = -XINF                                                  MML10700
         XKEI1 = -XINF                                                  MML10710
         GO TO 9005                                                     MML10720
    9000 CONTINUE                                                       MML10730
         CALL UERTST(IER,6HMMKEL1)                                      MML10740
    9005 RETURN                                                         MML10750
         END                                                            MML10760
//GO.FT09F001 DD DSN=WYL.X5.A95.DATA,DISP=SHR
//GO.SYSIN DD *
RELATIVE DAMAGE IN STORAGE LOCATIONS
   10 0.4      0.667       2.0
30000000. 450.       0.499     0.30     1.1       0.9       2.44          5.4E-5
    6.0E-6  0.06
160.0       8.
   11
  -60 5.59
  -40 4.46
  -20 3.47
    0 2.59
   20 1.81
   40 1.18
   60 0.48
   80 -0.08
  100 -0.59
  120 -1.16
  140 -1.48
```

```
// JOB (A95$X5,516,0.25,10),'GEORGE DERBALIAN'
/*JOBPARM FORMS=1481
// EXEC FORTCLG
//FORT.SYSIN DD *
C// EXEC WATFIV
C//GO.SYSIN DD *
C$WATFIV
C
C      CAPTIVE.FLIGHT
C     GEORGE DERBALIAN
C         APRIL 1981
C THIS PROGRAM COMPUTES ROCKET DAMAGE DURING CAPTIVE FLIGHT
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /GREGW/W,WA
      COMMON/FAIL/AT,TEMP,DD,SU,SCR,XIO
      DIMENSION ITP(25),SHIFT(25),TITL(20),CDFA(200),CDFD(200),DMG(100)
      READ(5,500) TITL
      WRITE(6,600) TITL
  500 FORMAT(20A4)
  600 FORMAT('1',/20X,'*** ',20A4,' ***')
      READ(5,551) ITIME,SCFD,SCFA,XKT
      IF (XKT.EQ.0.0) XKT=1.0
      READ(5,501) EC,EP,PR,PRC,DIFF,RI,B,ALP,ALPC,H
  501 FORMAT(8F10.3)
      WRITE(6,601) EC,EP,PR,PRC,DIFF,RI,B,ALP,ALPC,H
  601 FORMAT(/1X,'MODULUS OF CASING',F15.2,1X,'PSI',/1X,
     $ 'MODULUS OF PROPELLANT ',F15.2,1X,'PSI',/1X,
     1 'POISSON''S RATIO OF PROPELLANT',F10.4,/1X,
     2 'POISSON''S RATIO OF CASE',F10.4,/1X,
     3 'THERMAL DIFFUSIVITY OF PROPELLANT ',E12.4,1X,'IN*IN/HR',/1X,
     $ 'INNER RADIUS OF PROPELLANT',F10.4,1X,'IN',/1X,
     4 'RADIUS OF PROPELLANT ',F10.4,1X,'IN',/1X,
     5 'COEFFICIENT OF THERMAL EXPANSION OF PROPELLANT ',E12.4,1X,'1/F',
     6 /1X,'COEFFICIENT OF THERMAL EXPANSION OF CASE ',E12.4,1X,'1/F'
     $ /1X,'THICKNESS OF CASING',F10.5,1X,'IN')
      WRITE(6,651) ITIME,SCFD,SCFA,XKT
  651 FORMAT(/1X,'TOTAL TIME=',I5,
     1 ' HRS',5X,'DIURNAL AMPLITUDE SCALE=',F10.4,5X,'SEASONAL',
     2 ' AMPLITUDE SCALE=',F10.4,5X,'XKT=',F10.2)
  551 FORMAT(I5,3F10.3)
      READ(5,501) SU,SCR
      WRITE(6,614) SU,SCR
  614 FORMAT(/1X,'STRESS REQUIRED TO CAUSE FAILURE IN 1 MIN',E12.4,1X,'P
     $SI',5X,'CRITICAL STRESS BELOW WHICH NO FAILURES OCCUR',E12.4,1X,
     $ 'PSI')
      READ(5,551) NAT
      READ(5,552) (ITP(I),SHIFT(I),I=1,NAT)
  552 FORMAT(I5,F10.2)
      WRITE(6,610)
  610 FORMAT(/1X,'TEMPERATURE F',5X,'SHIFT FACTOR')
      WRITE(6,611) (ITP(I),SHIFT(I),I=1,NAT)
  611 FORMAT(3X,I5,8X,F10.3)
C     DEFINE GLOBAL CONSTANTS
      W=0.261799400
      WA=W/365.0
      CALL CONST(RI,B,H,EP,PR,ALP,DIFF,EC,PRC,ALPC)
C INITIALIZE RANDOM NUMBER GENERATING VARIABLE ISEED
      ISEED=983145267
C
```

```
      IR=0
   77 IR=IR+1
      WRITE(6,658)
  658 FORMAT('1')
      READ(5,551,END=99) IAMT,SD
C REFERENCE STRESS FREE TEMPERATURE 75F
      IAMT=IAMT-75
      WRITE(6,652) IAMT,SD,IR
  652 FORMAT(I5,1X,'AVERAGE ANNUAL TEMPERATURE',5X,'SD=',F10.2,5X,
     $ 'LOCATION=',I5)
C
      DMG(IR)=0.
      ATO=1.0D0
      XIO=0.0
      TAL=0.0
      TDL=0.0
C
      DO 8 I=1,ITIME
      TIME=I
      CALL AMPL(SD,AMP,ISEED)
      AMT=IAMT+AMP
      CALL TEMPER (TIME,AMT,TAL,TDL,TEMP)
      IF(TEMP+75.0.GT.ITP(1)) GO TO 10
      AT=SHIFT(1)+(TEMP+75.0-ITP(1))*(SHIFT(2)-SHIFT(1))/
     $ (ITP(2)-ITP(1))
      GO TO 11
   10 DO 12 II=2,NAT
      IF (ITP(II).LT.TEMP+75.) GO TO 12
      AT=SHIFT(II-1)+(TEMP+75.0-ITP(II-1))*(SHIFT(II)-SHIFT(II-1))/
     $ (ITP(II)-ITP(II-1))
      GO TO 11
   12 CONTINUE
      AT=SHIFT(NAT)+(TEMP+75.0-ITP(NAT))*(SHIFT(NAT)-SHIFT(NAT-1))/
     $ (ITP(NAT)-ITP(NAT-1))
   11 AT=DEXP(2.302585D0*AT)
      CALL STRESS(TIME,AMT,TAL,TDL,SH)
      DD=0.
CC    WRITE(6,640) I,SH
CC640 FORMAT(1X,'TIME=',I5,5X,'STRESS=',E12.5)
      SH=XKT*SH
      CALL DAMAGE(SH)
      DMG(IR)=DD+DMG(IR)
      ATO=AT
      K=I/100
      K=K*100
      K=I-K
      IF (K.EQ.0) WRITE(6,641) I,DMG(IR),TEMP
  641 FORMAT(1X,'TIME=',I5,5X,'DAMAGE=',E12.5,5X,'TEMP=',F10.2)
    8 CONTINUE
C
      GO TO 77
COULD USE A SORTING ROUTINE HERE
   99 IR=IR-1
      WRITE(6,657)
  657 FORMAT('1',1X,'RELATIVE DAMAGE')
      WRITE(6,656) (I,DMG(I),I=1,IR)
  656 FORMAT(1X,'LOCATION=',I3,5X,'DAMAGE=',E12.5)
      STOP
      END
C
```

```
C***************************************************************
C
      SUBROUTINE RANSEA(SCFA,CDFA,TAL,ISEED)
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*8 CDFA(200)
      REAL*4 X
      CALL RANDK(ISEED,X,0)
      XX=X
      DO 4 I=1,200
      XI=I
    4 IF (CDFA(I).GT.XX) GO TO 5
    5 TAL=(0.5*XI-0.5)*SCFA
      RETURN
      END
C
C***************************************************************
C
      SUBROUTINE RANDAY(SCFD,CDFD,TDL,ISEED)
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*8 CDFD(200)
      REAL*4 X
      CALL RANDK(ISEED,X,0)
      DO 6 I=1,200
      XX=X
      XI=I
    6 IF(CDFD(I).GT.XX) GO TO 7
    7 TDL=SCFD*(0.5*XI-0.5)
      RETURN
      END
C
C ***********************************************************
C
      SUBROUTINE AMPL(SD,AMP,ISEED)
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*4 P
      A0=2.30753
      A1=0.27061
      B1=0.99229
      B2=0.04481
      CALL RANDK(ISEED,P,0)
      PP=1.0-P
      IF (PP.GT.0.5) GO TO  1
      V=DSQRT(DLOG(1./PP/PP))
      AMP=(V-(A0+A1*V)/(1.0+B1*V+B2*V*V))*SD
      RETURN
    1 PP=1.0-PP
      V=DSQRT(DLOG(1.0/PP/PP))
      AMP=(-V+(A0+A1*V)/(1.0+B1*V+B2*V*V))*SD
      RETURN
      END
C
C***************************************************************
C
      SUBROUTINE RANDK (IY, YFL, INDEX)
C
C     THIS IS A UNIFORM RANDOM NUMBER GENERATOR WRITTEN BY G. E.
C  FORSYTHE IN SPRING 1969, FOLLOWING D. KNUTH, THE ART OF COMPUTER
C  PROGRAMMING, VOL. 2, PP. 155-156.  IT IS MUCH SUPERIOR TO RANDU,
C  THE RANDOM NUMBER GENERATOR FOUND IN IBM'S SCIENTIFIC SUBROUTINE
C  PACKAGE.
```

```
C
C     BEFORE THE FIRST CALL OF RANDK, IY SHOULD BE SET OUTSIDE RANDK
C  TO AN ARBITRARY INTEGER VALUE.  (IN WATFOR THIS IS ESSENTIAL.)
C  FOR PROGRAM CHECKOUT, THE INITIAL VALUE OF IY SHOULD BE A FIXED
C  INTEGER.  FOR RANDOM NUMBERS DIFFERENT ON EVERY RUN (AND HENCE
C  NOT REPRODUCIBLE), DECLARE INTEGER CLOCK1 AND THEN INITIALIZE
C  IY TO CLOCK1(4).
C
C     IF RANDK IS CALLED WITH AN INTEGER INDEX = 1, THEN THE OUTPUT
C  VALUE OF IY IS A PSEUDORANDOM INTEGER UNIFORMLY DISTRIBUTED IN THE
C  RANGE 0 <= IY < 2**31.
C
C     IF RANDK IS CALLED WITH INDEX = 0, THEN NOT ONLY IS IY PRODUCED,
C  BUT ALSO (AT SOME EXTRA COST IN TIME) A FLOATING NUMBER YFL, UNI-
C  FORMLY DISTRIBUTED IN THE INTERVAL 0.0 <= YFL < 1.0.
C
      IY = IY*314159269 + 453806245
    4 IF (IY .GE. 0) GO TO 6
C
C  CAUTION:  THE STATEMENT LABEL 4 IS ESSENTIAL IN ORDER TO PREVENT
C  CERTAIN COMPILERS (E.G., FORTRAN H WITH OPT 0) FROM PERFORMING
C  UNWANTED "OPTIMIZATIONS."  IT SHOULD NOT BE REMOVED.
C
    5    IY = IY + 2147483647 + 1
C        STATEMENT 5 ADDS 2**31 TO NEGATIVE VALUES OF IY
C
    6 IF (INDEX) 7, 7, 8
C
    7    YFL = IY
         YFL = YFL*.4656613E-9
C
    8 RETURN
      END
C
C ********************************************
C
      SUBROUTINE DAMAGE(S)
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON/FAIL/AT,TEMP,DD,SU,SCR,XIO
      IF (S-SCR .GT. 0.0) GO TO 1
      XIO=0.0
      RETURN
    1 CN=DLOG10(S-SCR)
      BB=9.3D0
      XI=((S-SCR)**BB)/AT
      DD=30.0D0*(XI+XIO)/(SU-SCR)**BB
      XIO=XI
      RETURN
      END
C
C **************************************************
C
      SUBROUTINE TEMPER(T,AMT,TAL,TDL,TEMP)
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /GREGT/ TCOSD,TSIND,TCOSY,TSINY
      COMMON /GREGW/ WDAY,WYEAR
      WT = WYEAR * T
      TEMP = TAL * (TCOSY*DCOS(WT) + TSINY*DSIN(WT))
      WT = WDAY * T
      TEMP = TEMP + TDL*(TCOSD*DCOS(WT)+TSIND*DSIN(WT))
```

```
      TEMP = TEMP + AMT
      RETURN
      END
      SUBROUTINE STRESS(T,AMT,TAL,TDL,SH)
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /GREGS/ SHCD,SHSD,SHCY,SHSY,SHCON
      COMMON /GREGW/ WDAY,WYEAR
      WT = WYEAR * T
      SH = TAL * (SHCY*DCOS(WT)+SHSY*DSIN(WT))
      WT = WDAY*T
      SH = SH + TDL*(SHCD*DCOS(WT)+SHSD*DSIN(WT))
      SH = SH + SHCON*AMT
      RETURN
      END
C
C ***********************************************
C
      SUBROUTINE CONST(RI,RO,H,EP,VP,ALP,DIFF,EC,VC,ALC)
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 INTC,INTS
      COMPLEX*16 BOA
      COMMON /GREG/ PR,E,F,DE,DF,C,D
      COMMON /GREGT/ TCOSD,TSIND,TCOSY,TSINY
      COMMON /GREGS/ SHCD,SHSD,SHCY,SHSY,SHCON
      COMMON /GREGW/ WDAY,WYEAR
      I = 1
      PDAY = DSQRT(WDAY/DIFF)
      PYEAR = DSQRT(WYEAR/DIFF)
      P = PDAY
   10 PA = P * RI
      CALL MMKEL1(PA,BER,BEI,XKER,XKEI,IER)
      BOA = DCMPLX(BEI,-BER)/DCMPLX(-XKEI,XKER)
      C = DREAL(BOA)
      D = DIMAG(BOA)
      PR = P * RO
      CALL EANDF
      A = -F/(E*E+F*F)
      B =  E/(E*E+F*F)
      PR = PA
      CALL EANDF
      TSINY = B*E - A*F
      TCOSY = A*E + B*F
      IF(I.EQ.2) GO TO 20
      TSIND = TSINY
      TCOSD = TCOSY
      I = 2
      P = PYEAR
      GO TO 10
   20 I = 1
      P = PDAY
      DENOM=(1.D0+VP)*((1.D0-2.D0*VP)*RO*RO+RI*RI)/(RO*RO-RI*RI)
     C + (1.D0-VC*VC)*RO*EP/(H*EC)
   30 PR = P * RO
      CALL DEADF
      Z1 = 2.D0*EP*ALP*(1.D0+VP)*RO/(P*(RO*RO-RI*RI))
      ZS = A*DE + B*DF
      ZC = A*DF - B*DE
      PS = (Z1*ZS - ALC*(1.D0+VC)*EP)/DENOM
      PC = Z1*ZC/DENOM
      PCON = ((Z1*P*(RO-RI)/RO)-ALC*(1.D0+VC)*EP)/DENOM
```

```
      Z1 = RO*RO/(RO*RO-RI*RI)
      PC = PC * Z1
      PS = PS * Z1
      PCON = 2.D0*Z1*PCON
      Z1 = ALP*EP*RO/(P*(1.D0-VP)*(RO*RO-RI*RI))
      INTC = Z1*ZC
      INTS = Z1*ZS
      INTCON = 2.D0*ALP*EP/((1.D0-VP)*(RO+RI))
      PR = P * RI
      CALL EANDF
      CALL DEADF
      Z1 = ALP*EP/(P*(1.D0-VP)*RI)
      SHCY = Z1 * (A*DF-B*DE)
      SHSY = Z1 * (A*DE + B*DF)
      Z1 = ALP*EP/(1.D0-VP)
      SHCON = INTCON - PCON - Z1
      SHCY = SHCY - Z1*(A*E+B*F)
      SHCY = SHCY + (INTC-PC)*2.D0
      SHSY = SHSY - Z1*(-A*F+B*E)
      SHSY = SHSY + (INTS-PS)*2.D0
      IF(I.EQ.2) RETURN
      I = 2
      P = PYEAR
      SHCD = SHCY
      SHSD = SHSY
      GO TO 30
      END
C
C ******************************************************************
C
      SUBROUTINE EANDF
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON /GREG/ PR,E,F,DE,DF,C,D
      CALL MMKELO(PR,BER,BEI,XKER,XKEI,IER)
      E = BER + C*XKER - D*XKEI
      F = BEI + C*XKEI + D*XKER
      RETURN
      END
      SUBROUTINE DEADF
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON /GREG/ PR,E,F,DE,DF,C,D
      CALL MMKELD(PR,BER,BEI,XKER,XKEI,IER)
      DE = BER + C*XKER - D*XKEI
      DF = BEI + C*XKEI + D*XKER
      RETURN
      END
C     SUBROUTINE MMKELO (X,BER,BEI,XKER,XKEI,IER)                       MML00010
C                                                                       MML00020
C-MMKELO--------D-------LIBRARY 1---------------------------------------MML00030
C                                                                       MML00040
C   FUNCTION            - EVALUATE THE KELVIN FUNCTIONS BER,BEI,KER AND MML00050
C                           KEI OF ORDER ZERO                           MML00060
C   USAGE               - CALL MMKELO(X,BER,BEI,XKER,XKEI,IER)          MML00070
C   PARAMETERS   X      - INPUT ARGUMENT. IF X IS NEGATIVE, A WARNING   MML00080
C                           ERROR IS PRODUCED AND VALUES OF POSITIVE    MML00090
C                           MACHINE INFINITY WILL BE RETURNED FOR XKER  MML00100
C                           AND XKEI.                                   MML00110
C                BER    - OUTPUT ARGUMENT                               MML00120
C                BEI    - OUTPUT ARGUMENT                               MML00130
C                XKER   - OUTPUT ARGUMENT RETURNED ONLY WHEN X IS       MML00140
```

```
C                              POSITIVE.                                     MML00150
C                 XKEI   - OUTPUT ARGUMENT RETURNED ONLY WHEN X IS           MML00160
C                            POSITIVE.                                       MML00170
C                 IER    - ERROR PARAMETER                                   MML00180
C                          TERMINAL ERROR = 128+N.                           MML00190
C                            N = 1 INDICATES THAT THE ABSOLUTE VALUE OF      MML00200
C                              X WAS GREATER THAN 119. BER AND BEI ARE       MML00210
C                              SET TO ZERO. IF X IS NON-NEGATIVE, XKER       MML00220
C                              AND XKEI ARE ALSO SET TO ZERO. OTHERWISE,     MML00230
C                              XKER AND XKEI ARE SET TO POSITIVE MACHINE     MML00240
C                              INFINITY.                                     MML00250
C                          WARNING ERROR = 32 + N                            MML00260
C                            N = 2 INDICATES THAT X IS NEGATIVE.             MML00270
C                              XKER AND XKEI WILL BE RETURNED AS             MML00280
C                              POSITIVE MACHINE INFINITY.                    MML00290
C   PRECISION            - DOUBLE                                            MML00300
C   REQD. IMSL ROUTINES - UERTST                                             MML00310
C   LANGUAGE             - FORTRAN                                           MML00320
C-----------------------------------------------------------------------MML00330
C   LATEST REVISION      - APRIL 30,1975                                     MML00340
C                                                                            MML00350
      SUBROUTINE MMKELO (X,BER,BEI,XKER,XKEI,IER)                            MML00360
C                                                                            MML00370
      DIMENSION          C1(9),C2(9),C3(9),C4(9),E1(9),E2(9)                 MML00380
      DOUBLE PRECISION   C1,C2,C3,C4,E1,E2,PIO8,RT2,XINF,                    MML00390
     *                   PI,EUL,TEN,ZERO,HALF,ONE,ARG,BER,BEI,B1,B2,B3,      MML00400
     *                   B4,CON,DC,DCM,DE,DS,DSM,DSQ,PIO2,R1,R2,S,SM,T,      MML00410
     *                   TM,TWOPI,X,XKER,XKEI,Z,ZI,ZIM,ZSQ,Z4,ZMAX           MML00420
C                                                                            MML00430
C                                  COEFFICIENTS FOR EVALUATION OF            MML00440
C                                  BER-SUB-ZERO(X) FOR X GREATER THAN        MML00450
C                                  0. AND LESS THAN OR EQUAL TO 10.          MML00460
C                                                                            MML00470
      DATA               C1(1)/5.16070465D-5/,C1.. '-4.8987125727D-3/        MML00480
      DATA               C1(3)/.25977730007D0/,C1(4)/-7.2422567278207D0/MML00490
      DATA               C1(5)/93.8596692971726D0/                           MML00500
      DATA               C1(6)/-470.95027958899680D0/                        MML00510
      DATA               C1(7)/678.16840276630910D0/                         MML00520
      DATA               C1(8)/-156.24999999957010D0/                        MML00530
      DATA               C1(9)/.99999999999740D0/                            MML00540
C                                                                            MML00550
C                                  COEFFICIENTS FOR EVALUATION OF            MML00560
C                                  BEI-SUB-ZERO(X) FOR X GREATER THAN 0.MML00570
C                                  AND X LESS THAN OR EQUAL TO 10.           MML00580
C                                                                            MML00590
      DATA               C2(1)/4.9130000D-6/,C2(2)/-5.444243175D-4/          MML00600
      DATA               C2(3)/3.842882827340D-2/                            MML00610
      DATA               C2(4)/-1.4963342749742D0/                           MML00620
      DATA               C2(5)/28.969033878649900/                           MML00630
      DATA               C2(6)/-240.28075494425740D0/                        MML00640
      DATA               C2(7)/678.16840277698070D0/                         MML00650
      DATA               C2(8)/-434.02777777774790D0/                        MML00660
      DATA               C2(9)/24.9999999999998D0/                           MML00670
C                                                                            MML00680
C                                  COEFFICIENTS FOR EVALUATION OF            MML00690
C                                  KEI-SUB-ZERO(X) FOR X GREATER THAN        MML00700
C                                  OR EQUAL TO ZERO AND X LESS THAN OR       MML00710
C                                  EQUAL TO 10.                              MML00720
C                                                                            MML00730
      DATA               C3(1)/1.54363047D-5/,C3(2)/-1.8064777860D-3/        MML00740
```

```
          DATA                C3(3)/.122208738219200/                   MML00750
          DATA                C3(4)/-4.518745913263900/                 MML00760
          DATA                C3(5)/81.952477160620000/                 MML00770
          DATA                C3(6)/-623.01367174052010D0/              MML00780
          DATA                C3(7)/1548.484519673099200/               MML00790
          DATA                C3(8)/-795.717592592486600/               MML00800
          DATA                C3(9)/24.999999999999300/                 MML00810
      C                                                                 MML00820
      C                                 COEFFICIENTS FOR EVALUATION OF  MML00830
      C                                 KER-SUB-ZERO(X) FOR X GREATER THAN ORMML00840
      C                                 EQUAL TO ZERO AND X LESS THAN OR   MML00850
      C                                 EQUAL TO TEN                    MML00860
      C                                                                 MML00870
          DATA                C4(1)/1.2161109D-6/,C4(2)/-1.797627986D-4/   MML00880
          DATA                C4(3)/1.59380149705D-2/                   MML00890
          DATA                C4(4)/-.806152902787600/                  MML00900
          DATA                C4(5)/21.212345166023100/                 MML00910
          DATA                C4(6)/-255.097174271047900/               MML00920
          DATA                C4(7)/1153.828185281456100/               MML00930
          DATA                C4(8)/-1412.850839120363600/              MML00940
          DATA                C4(9)/234.37500/                          MML00950
      C                                                                 MML00960
      C                                 COEFFICIENTS FOR EVALUATION OF  MML00970
      C                                 AUXILIARY FUNCTIONS FOR X GREATER  MML00980
      C                                 THAN 10.                        MML00990
      C                                                                 MML01000
          DATA                E1(1)/4.92D-8/,E1(2)/1.452D-7/,E1(3)/1.35D-8/  MML01010
          DATA                E1(4)/-1.6192D-6/,E1(5)/-1.12207D-5/      MML01020
          DATA                E1(6)/-5.17869D-5/,E1(7)/7.0D-10/         MML01030
          DATA                E1(8)/8.8388346D-3/,E1(9)/1.0D0/          MML01040
          DATA                E2(1)/-2.43D-8/,E2(2)/7.5D-8/,E2(3)/5.929D-7/  MML01050
          DATA                E2(4)/1.6431D-6/,E2(5)/-7.2D-9/           MML01060
          DATA                E2(6)/-5.18006D-5/,E2(7)/-7.031241D-4/    MML01070
          DATA                E2(8)/-8.8388340D-3/,E2(9)/0.0D0/         MML01080
      C                                                                 MML01090
      C                                 MISCELLANEOUS CONSTANTS         MML01100
      C                                                                 MML01110
          DATA                PIO2/1.570796326794896600/                MML01120
          DATA                TWOPI/6.283185307179586D0/                MML01130
          DATA                PIO8/.392699081698724150D0/               MML01140
          DATA                RT2/.7071067811865475200/                 MML01150
          DATA                XINF/Z7FFFFFFFFFFFFFFF/                   MML01160
          DATA                PI/3.141592653589793200/                  MML01170
          DATA                EUL/.577215664901532860D0/                MML01180
          DATA                TEN/10.D0/,ZERO/0.D0/,HALF/.5D0/,ONE/1.D0/   MML01190
          DATA                ZMAX/119.D0/                              MML01200
          IER = 0                                                       MML01210
          Z = DABS(X)                                                   MML01220
          IF (Z .GT. TEN) GO TO 15                                      MML01230
          IF (Z .EQ. ZERO) GO TO 10                                     MML01240
      C                                 CALCULATION OF FUNCTIONS FOR ABS(X)  MML01250
      C                                 LESS THAN 10.                   MML01260
          Z = Z/TEN                                                     MML01270
          ZSQ = Z*Z                                                     MML01280
          Z4 = ZSQ*ZSQ                                                  MML01290
          B1 = C1(1)                                                    MML01300
          B2 = C2(1)                                                    MML01310
          B3 = C3(1)                                                    MML01320
          B4 = C4(1)                                                    MML01330
          DO 5 I = 2,9                                                  MML01340
```

```
          B1 = B1*Z4+C1(I)                                          MML01350
          B2 = B2*Z4+C2(I)                                          MML01360
          B3 = B3*Z4+C3(I)                                          MML01370
          B4 = B4*Z4+C4(I)                                          MML01380
    5 CONTINUE                                                      MML01390
      BER = B1                                                      MML01400
      BEI = ZSQ*B2                                                  MML01410
      IF (X .LT. ZERO) GO TO 30                                     MML01420
      R1 = ZSQ*B3                                                   MML01430
      R2 = Z4*B4                                                    MML01440
      CON = (DLOG(X*HALF)+EUL)                                      MML01450
      XKEI = -PIO2*HALF*BER+(R1-BEI*CON)                            MML01460
      XKER = PIO2*HALF*BEI-(R2+BER*CON)                             MML01470
      GO TO 9005                                                    MML01480
C                             X EQUAL 0. DEFAULT TO PROPER VALUES   MML01490
   10 BER = ONE                                                     MML01500
      BEI = ZERO                                                    MML01510
      XKEI = -HALF*PIO2                                             MML01520
      XKER = XINF                                                   MML01530
      GO TO 9005                                                    MML01540
C                             X GREATER THAN 10. CALCULATE          MML01550
C                             AUXILIARY FUNCTIONS                   MML01560
   15 IF (Z .GT. ZMAX) GO TO 25                                     MML01570
      ZI = TEN/Z                                                    MML01580
      ZIM = -ZI                                                     MML01590
      S = E1(1)                                                     MML01600
      SM = S                                                        MML01610
      T = E2(1)                                                     MML01620
      TM = T                                                        MML01630
      DO 20 I = 2,9                                                 MML01640
         S = S*ZI+E1(I)                                             MML01650
         T = T*ZI+E2(I)                                             MML01660
         SM = SM*ZIM+E1(I)                                          MML01670
         TM = TM*ZIM+E2(I)                                          MML01680
   20 CONTINUE                                                      MML01690
      ARG = Z*RT2                                                   MML01700
      DS = DSIN(ARG-PIO8)                                           MML01710
      DC = DCOS(ARG-PIO8)                                           MML01720
      DSM = DSIN(ARG+PIO8)                                          MML01730
      DCM = DCOS(ARG+PIO8)                                          MML01740
      DE = DEXP(ARG)                                                MML01750
      DSQ = DSQRT(TWOPI*Z)                                          MML01760
C                             CALCULATE THE DESIRED FUNCTIONS       MML01770
      BER = DE*(S*DC-T*DS)/DSQ                                      MML01780
      BEI = DE*(T*DC+S*DS)/DSQ                                      MML01790
      IF (X .LT. ZERO) GO TO 30                                     MML01800
      XKEI = PI*(TM*DCM-SM*DSM)/(DE*DSQ)                            MML01810
      XKER = PI*(SM*DCM+TM*DSM)/(DE*DSQ)                            MML01820
      GO TO 9005                                                    MML01830
C                             Z TOO LARGE.                          MML01840
   25 BER = ZERO                                                    MML01850
      BEI = ZERO                                                    MML01860
      IER = 129                                                     MML01870
      IF (X .LT. ZERO) GO TO 35                                     MML01880
      XKEI = ZERO                                                   MML01890
      XKER = ZERO                                                   MML01900
      GO TO 9000                                                    MML01910
C                             X LESS THAN 0. DEFAULT TO PROPER      MML01920
C                             VALUES                                MML01930
   30 IER = 34                                                      MML01940
```

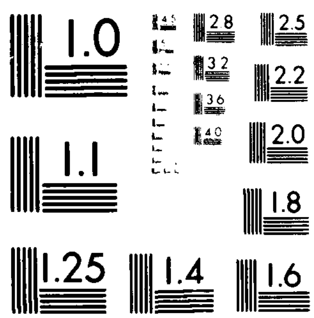
AD-A117 651   FAILURE ANALYSIS ASSOCIATES PALO ALTO CA   F/G 21/8.2
PROBABILISTIC ENVIRONMENTAL MODEL FOR SOLID ROCKET MOTOR LIFE P--ETC(U)
MAR 82   G DERBALIAN, J M THOMAS, P JOHNSTON   N60530-78-C-0127

UNCLASSIFIED   NWC-TP-6305   NL

3 OF 3

AD
A117651

END
DATE
FILMED
8 82
DTIC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

```
   35 XKEI = XINF                                                      MMLO1950
      XKER = XINF                                                      MMLO1960
      IF (IER .EQ. 0) GO TO 9005                                       MMLO1970
 9000 CONTINUE                                                         MMLO1980
      CALL UERTST (IER,6HMMKELO)                                       MMLO1990
 9005 RETURN                                                           MMLO2000
      END                                                              MMLO2010
C     SUBROUTINE MMKELD (X,BERP,BEIP,XKERP,XKEIP,IER)                  MMLL0010
C                                                                      MMLL0020
C-MMKELD--------D-------LIBRARY 1--------------------------------------MMLL0030
C                                                                      MMLL0040
C   FUNCTION            - EVALUATE THE DERIVATIVES OF THE KELVIN       MMLL0050
C                           FUNCTIONS (BER,BEI,KER AND KEI) OF ORDER   MMLL0060
C                           ZERO.                                      MMLL0070
C   USAGE               - CALL MMKELD(X,BERP,BEIP,XKERP,XKEIP,IER)     MMLL0080
C   PARAMETERS   X      - INPUT ARGUMENT. IF X IS NEGATIVE, A WARNING  MMLL0090
C                           ERROR IS PRODUCED AND VALUES OF POSITIVE   MMLL0100
C                           MACHINE INFINITY WILL BE RETURNED FOR XKERP MMLL0110
C                           AND KEIP.                                  MMLL0120
C                BERP   - OUTPUT ARGUMENT                              MMLL0130
C                BEIP   - OUTPUT ARGUMENT                              MMLL0140
C                XKERP  - OUTPUT ARGUMENT RETURNED ONLY WHEN X IS      MMLL0150
C                           POSITIVE.                                  MMLL0160
C                XKEIP  - OUTPUT ARGUMENT RETURNED ONLY WHEN X IS      MMLL0170
C                           POSITIVE.                                  MMLL0180
C                IER    - ERROR PARAMETER.                             MMLL0190
C                         TERMINAL ERROR = 128+N.                      MMLL0200
C                           N = 1 INDICATES THAT THE ABSOLUTE VALUE OF MMLL0210
C                             X WAS GREATER THAN 119. BERP AND BEIP ARE MMLL0220
C                             SET TO ZERO. IF X IS NON-NEGATIVE, XKERP MMLL0230
C                             AND XKEIP ARE ALSO SET TO ZERO. OTHERWISE,MMLL0240
C                             XKERP AND XKEIP ARE SET TO POSITIVE      MMLL0250
C                             MACHINE INFINITY.                        MMLL0260
C                         WARNING ERROR = 32+N.                        MMLL0270
C                           N = 2 INDICATES THAT X IS NEGATIVE.        MMLL0280
C                             XKERP AND XKEIP WILL BE RETURNED AS      MMLL0290
C                             POSITIVE MACHINE INFINITY.               MMLL0300
C   PRECISION           - DOUBLE                                       MMLL0310
C   REQD. IMSL ROUTINES - MMKELO,UERTST                                MMLL0320
C   LANGUAGE            - FORTRAN                                      MMLL0330
C----------------------------------------------------------------------MMLL0340
C   LATEST REVISION     - SEPTEMBER 22,1976                            MMLL0350
C                                                                      MMLL0360
      SUBROUTINE MMKELD(X,BERP,BEIP,XKERP,XKEIP,IER)                   MMLL0370
C                                                                      MMLL0380
      DIMENSION          D1(9),D2(9),D3(9),D4(9),E3(9),E4(9)           MMLL0390
      DOUBLE PRECISION   ARG,BEI,BEIP,BER,BERP,B1,B2,B3,B4,CON,DC,DCM, MMLL0400
     *                   DE,DS,DSM,DSQ,D1,D2,D3,D4,EUL,E3,E4,PI,PID2,  MMLL0410
     *                   PIO8,RT2,R1P,R2P,TWOPI,U,UM,V,VM,X,XINF,XKEI, MMLL0420
     *                   XKEIP,XKER,XKERP,Z,ZI,ZIM,ZSQ,Z3,Z4,ZMAX      MMLL0430
      DOUBLE PRECISION   TEN,ZERO,HALF                                 MMLL0440
      DATA               TEN/10.D0/,ZERO/0.D0/,HALF/.5D0/              MMLL0450
C                                                                      MMLL0460
C                                  COEFFICIENTS FOR EVALUATION OF BERP- MMLL0470
C                                  SUB-ZERO(X) FOR X GREATER THAN 0. ANDMMLL0480
C                                  LESS THAN OR EQUAL TO 10.           MMLL0490
C                                                                      MMLL0500
      DATA               D1(1)/-1.2506046D-6/,D1(2)/1.701453451D-4/    MMLL0510
      DATA               D1(3)/-1.37246036190D-2/                      MMLL0520
      DATA               D1(4)/.62347263482430D0/                      MMLL0530
```

```
      DATA          D1(5)/-14.4845169498403D0/                     MMLL0540
      DATA          D1(6)/150.175471843227 8D0/                    MMLL0550
      DATA          D1(7)/-565.140335647948 6D0/                   MMLL0560
      DATA          D1(8)/542.534722222214 7D0/                    MMLL0570
      DATA          D1(9)/-62.4999999999999D0/                     MMLL0580
C                                                                  MMLL0590
C                             COEFFICIENTS FOR EVALUATION OF BEIP- MMLL0600
C                             SUB-ZERO(X) FOR X GREATER THAN 0.    MMLL0610
C                             AND LESS THAN OR EQUAL TO 10.        MMLL0620
C                                                                  MMLL0630
      DATA          D2(1)/1.52269884D-5/,D2(2)/-1.63311008370-3/   MMLL0640
      DATA          D2(3)/9.99147064932D-2/                        MMLL0650
      DATA          D2(4)/-3.29193521085790D0/                     MMLL0660
      DATA          D2(5)/52.1442608975905D0/                      MMLL0670
      DATA          D2(6)/-336.393056902365 1D0/                   MMLL0680
      DATA          D2(7)/678.168402774753 9D0/                    MMLL0690
      DATA          D2(8)/-260.416666666553 3D0/                   MMLL0700
      DATA          D2(9)/4.9999999999993D0/                       MMLL0710
C                                                                  MMLL0720
C                             COEFFICIENTS FOR EVALUTION OF KEIP-  MMLL0730
C                             SUB-ZERO(X) FOR X GREATER THAN 0.    MMLL0740
C                             AND LESS THAN OR EQUAL TO 10.        MMLL0750
C                                                                  MMLL0760
      DATA          D3(1)/5.232943140-5/                           MMLL0770
      DATA          D3(2)/-5.41885584080-3/                        MMLL0780
      DATA          D3(3)/.317741843468 6D0/                       MMLL0790
      DATA          D3(4)/-9.94124032097 25D0/                     MMLL0800
      DATA          D3(5)/147.514458591333 7D0/                    MMLL0810
      DATA          D3(6)/-872.219140367245 5D0/                   MMLL0820
      DATA          D3(7)/1548.48451966520 35D0/                   MMLL0830
      DATA          D3(8)/-477.430555555153 6D0/                   MMLL0840
      DATA          D3(9)/4.9999999999975D0/                       MMLL0850
C                                                                  MMLL0860
C                             COEFFICIENTS FOR EVALUATION OF KERP- MMLL0870
C                             SUB-ZERO(X) FOR X GREATER THAN OR    MMLL0880
C                             EQUAL TO 0. AND LESS THAN OR EQUAL   MMLL0890
C                             TO 10.                               MMLL0900
C                                                                  MMLL0910
      DATA          D4(1)/4.36820530-6/,D4(2)/-5.75204228 3D-4/    MMLL0920
      DATA          D4(3)/4.46263862145D-2/                        MMLL0930
      DATA          D4(4)/-1.93476692292 37D0/                     MMLL0940
      DATA          D4(5)/42.4246903131088D0/                      MMLL0950
      DATA          D4(6)/-408.155478829257 8D0/                   MMLL0960
      DATA          D4(7)/1384.59382233724 52D0/                   MMLL0970
      DATA          D4(8)/-1130.28067129626 94D0/                  MMLL0980
      DATA          D4(9)/93.7499999999998D0/                      MMLL0990
C                                                                  MMLL1000
C                                                                  MMLL1010
C                             COEFFICIENTS FOR EVALUTION OF        MMLL1020
C                             AUXILIARY FUNCTIONS FOR X GREATER    MMLL1030
C                             THAN 10.                             MMLL1040
C                                                                  MMLL1050
      DATA          E3(1)/-5.63D-8/,E3(2)/-1.671D-7/               MMLL1060
      DATA          E3(3)/-1.47D-8/,E3(4)/1.9780D-6/               MMLL1070
      DATA          E3(5)/1.44255D-5/,E3(6)/7.25024D-5/            MMLL1080
      DATA          E3(7)/-8.0D-10/,E3(8)/-2.65165040D-2/          MMLL1090
      DATA          E3(9)/1.0D0/                                   MMLL1100
      DATA          E4(1)/-2.69D-8/,E4(2)/-8.83D-8/                MMLL1110
      DATA          E4(3)/-6.992D-7/,E4(4)/-2.0042D-6/             MMLL1120
      DATA          E4(5)/7.9D-9/,E4(6)/7.25179D-5/                MMLL1130
```

```
      DATA          E4(7)/1.1718740D-3/,E4(8)/2.65165034D-2/          MMLL1140
      DATA          E4(9)/0.0D0/                                      MMLL1150
C                                                                     MMLL1160
C                               MISCELLANEOUS CONSTANTS               MMLL1170
C                                                                     MMLL1180
      DATA          PIO2/1.5707963267948966D0/                        MMLL1190
      DATA          TWOPI/6.2831853071795860D0/                       MMLL1200
      DATA          PIO8/0.39269908169872415D0/                       MMLL1210
      DATA          RT2/0.70710678118654752D0/                        MMLL1220
      DATA          XINF/Z7FFFFFFFFFFFFFFF/                           MMLL1230
      DATA          PI/3.1415926535897932D0/                          MMLL1240
      DATA          EUL/0.57721566490153286D0/                        MMLL1250
      DATA          ZMAX/119.D0/                                      MMLL1260
      IER = 0                                                         MMLL1270
      CALL MMKEL0(X,BER,BEI,XKER,XKEI,IER)                            MMLL1280
      Z = DABS(X)                                                     MMLL1290
      IF (Z .GT. TEN) GO TO 15                                        MMLL1300
      IF (Z .EQ. ZERO) GO TO 10                                       MMLL1310
C                               CALCULATION OF FUNCTIONS FOR ABS(X)   MMLL1320
C                               LESS THAN 10.                         MMLL1330
      Z = Z/TEN                                                       MMLL1340
      ZSQ = Z*Z                                                       MMLL1350
      Z3 = ZSQ*Z                                                      MMLL1360
      Z4 = ZSQ*ZSQ                                                    MMLL1370
      B1 = D1(1)                                                      MMLL1380
      B2 = D2(1)                                                      MMLL1390
      B3 = D3(1)                                                      MMLL1400
      B4 = D4(1)                                                      MMLL1410
      DO 5 I = 2,9                                                    MMLL1420
         B1 = B1*Z4+D1(I)                                             MMLL1430
         B2 = B2*Z4+D2(I)                                             MMLL1440
         B3 = B3*Z4+D3(I)                                             MMLL1450
         B4 = B4*Z4+D4(I)                                             MMLL1460
    5 CONTINUE                                                        MMLL1470
      BERP = B1*Z3                                                    MMLL1480
      BEIP = Z*B2                                                     MMLL1490
      IF ( X .LT. ZERO) GO TO 30                                      MMLL1500
      R1P = Z*B3                                                      MMLL1510
      R2P = Z3*B4                                                     MMLL1520
      CON = (DLOG(X*HALF) + EUL)                                      MMLL1530
      V = DABS(X)                                                     MMLL1540
      XKEIP = -PIO2*HALF*BERP+(R1P-BEIP*CON-BEI/V)                    MMLL1550
      XKERP = PIO2*HALF*BEIP-(R2P+BERP*CON+BER/V)                     MMLL1560
      GO TO 9005                                                      MMLL1570
C                               X EQUAL TO 0. DEFAULT TO PROPER       MMLL1580
C                               VALUES                                MMLL1590
   10 BERP = ZERO                                                     MMLL1600
      BEIP = ZERO                                                     MMLL1610
      XKEIP = ZERO                                                    MMLL1620
      XKERP = -XINF                                                   MMLL1630
      GO TO 9005                                                      MMLL1640
C                               X GREATER [illegible]). CALCULATE     MMLL1650
C                               AUXILIARY FUNC[illegible]S            MMLL1660
   15 IF (Z .GT. ZMAX) GO TO 25                                       MMLL1670
      ZI = TEN/Z                                                      MMLL1680
      ZIM = -ZI                                                       MMLL1690
      U = E3(1)                                                       MMLL1700
      UM = U                                                          MMLL1710
      V = E4(1)                                                       MMLL1720
      VM = V                                                          MMLL1730
```

```
      DO 20 I = 2,9                                                   MMLL1740
         U = U*ZI+E3(I)                                               MMLL1750
         V = V*ZI+E4(I)                                               MMLL1760
         UM = UM*ZIM+E3(I)                                            MMLL1770
         VM = VM*ZIM+E4(I)                                            MMLL1780
   20 CONTINUE                                                        MMLL1790
      ARG = Z*RT2                                                     MMLL1800
      DS = DSIN(ARG-PIO8)                                             MMLL1810
      DC = DCOS(ARG-PIO8)                                             MMLL1820
      DSM = DSIN(ARG+PIO8)                                            MMLL1830
      DCM = DCOS(ARG+PIO8)                                            MMLL1840
      DE = DEXP(ARG)                                                  MMLL1850
      DSQ = DSQRT(TWOPI*Z)                                            MMLL1860
C                                 CALCULATE THE DESIRED FUNCTIONS     MMLL1870
      BERP = DE*(U*DCM-V*DSM)/DSQ                                     MMLL1880
      BEIP = DE*(V*DCM+U*DSM)/DSQ                                     MMLL1890
      IF (X .LT. ZERO) GO TO 30                                       MMLL1900
      XKEIP = -PI*(VM*DC-UM*DS)/(DE*DSQ)                              MMLL1910
      XKERP = -PI*(UM*DC+VM*DS)/(DE*DSQ)                              MMLL1920
      GO TO 9005                                                      MMLL1930
C                                 Z TOO LARGE.                        MMLL1940
   25 BERP = ZERO                                                     MMLL1950
      BEIP = ZERO                                                     MMLL1960
      IER = 129                                                       MMLL1970
      IF (X .LT. ZERO) GO TO 35                                       MMLL1980
      XKEIP = ZERO                                                    MMLL1990
      XKERP = ZERO                                                    MMLL2000
      GO TO 9000                                                      MMLL2010
C                                 X LESS THAN 0. DEFAULT TO PROPER    MMLL2020
C                                 VALUES                              MMLL2030
   30 IER = 34                                                        MMLL2040
      BERP = -BERP                                                    MMLL2050
      BEIP = -BEIP                                                    MMLL2060
   35 XKERP = XINF                                                    MMLL2070
      XKEIP = XINF                                                    MMLL2080
      IF (IER .EQ. 0) GO TO 9005                                      MMLL2090
 9000 CONTINUE                                                        MMLL2100
      CALL UERTST(IER,6HMMKELD)                                       MMLL2110
 9005 RETURN                                                          MMLL2120
      END                                                             MMLL2130
C     SUBROUTINE UERTST (IER,NAME)                                    UERT0010
C                                                                     UERT0020
C-UERTST----------------LIBRARY 1-------------------------------------UERT0030
C                                                                     UERT0040
C   FUNCTION            - ERROR MESSAGE GENERATION                    UERT0050
C   USAGE               - CALL UERTST(IER,NAME)                       UERT0060
C   PARAMETERS   IER    - ERROR PARAMETER. TYPE + N  WHERE            UERT0070
C                           TYPE= 128 IMPLIES TERMINAL ERROR          UERT0080
C                                  64 IMPLIES WARNING WITH FIX        UERT0090
C                                  32 IMPLIES WARNING                 UERT0100
C                           N   = ERROR CODE RELEVANT TO CALLING ROUTINEUERT0110
C                NAME   - INPUT VECTOR CONTAINING THE NAME OF THE     UERT0120
C                           CALLING ROUTINE AS A SIX CHARACTER LITERAL UERT0130
C                           STRING.                                   UERT0140
C   LANGUAGE            - FORTRAN                                     UERT0150
C---------------------------------------------------------------------UERT0160
C   LATEST REVISION     - JANUARY 18, 1974                            UERT0170
C                                                                     UERT0180
      SUBROUTINE UERTST(IER,NAME)                                     UERT0190
C                                                                     UERT0200
```

```
          DIMENSION        ITYP(5,4),IBIT(4)                           UERT0210
          INTEGER*2        NAME(3)                                     UERT0220
          INTEGER          WARN,WARF,TERM,PRINTR                       UERT0230
          EQUIVALENCE      (IBIT(1),WARN),(IBIT(2),WARF),(IBIT(3),TERM) UERT0240
          DATA     ITYP    /'WARN','ING ','    ','    ','    ',        UERT0250
         *                  'WARN','ING(','WITH',' FIX',')   ',        UERT0260
         *                  'TERM','INAL','    ','    ','    ',        UERT0270
         *                  'NON-','DEFI','NED ','    ','    '/,       UERT0280
         *         IBIT    / 32,64,128,0/                              UERT0290
          DATA     PRINTR  / 6/                                        UERT0300
          IER2=IER                                                     UERT0310
          IF (IER2 .GE. WARN) GO TO 5                                  UERT0320
C                                     NON-DEFINED                      UERT0330
          IER1=4                                                       UERT0340
          GO TO 20                                                     UERT0350
    5     IF (IER2 .LT. TERM) GO TO 10                                 UERT0360
C                                     TERMINAL                         UERT0370
          IER1=3                                                       UERT0380
          GO TO 20                                                     UERT0390
   10     IF (IER2 .LT. WARF) GO TO 15                                 UERT0400
C                                     WARNING(WITH FIX)                UERT0410
          IER1=2                                                       UERT0420
          GO TO 20                                                     UERT0430
C                                     WARNING                          UERT0440
   15     IER1=1                                                       UERT0450
C                                     EXTRACT 'N'                      UERT0460
   20     IER2=IER2-IBIT(IER1)                                         UERT0470
C                                     PRINT ERROR MESSAGE              UERT0480
          WRITE (PRINTR,25) (ITYP(I,IER1),I=1,5),NAME,IER2,IER         UERT0490
   25     FORMAT(' *** I M S L(UERTST) ***  ',5A4,4X,3A2,4X,I2,        UERT0500
         *   ' (IER = ',I3,')')                                        UERT0510
          RETURN                                                       UERT0520
          END                                                          UERT0530
          SUBROUTINE MMKEL1 (X,BER1,BEI1,XKER1,XKEI1,IER)              MML10360
C                                                                      MML10370
          DOUBLE PRECISION   BER1,BEI1,BERP,BEIP,RT2,X,XINF,XKEIP,XKEI1,  MML10380
         *                   XKERP,XKER1,ZERO,ZMAX                     MML10390
          DATA               XINF/Z7FFFFFFFFFFFFFFF/,ZERO/0.D0/        MML10400
          DATA               RT2/0.70710678118654752D0/                MML10410
          DATA               ZMAX/119.D0/                              MML10420
          IER = 0                                                      MML10430
          IF (X .EQ. ZERO) GO TO 15                                    MML10440
          IF (DABS(X) .GT. ZMAX) GO TO 10                              MML10450
          CALL MMKELD(X,BERP,BEIP,XKERP,XKEIP,IER)                     MML10460
          BEI1 = (BERP+BEIP) *RT2                                      MML10470
          BER1 = (BERP - BEIP) * RT2                                   MML10480
          IF (X .LT. ZERO) GO TO 5                                     MML10490
          XKEI1 = (XKERP + XKEIP) * RT2                                MML10500
          XKER1 = (XKERP - XKEIP) * RT2                                MML10510
          GO TO 9005                                                   MML10520
C                                     ARGUMENT IS NEGATIVE             MML10530
    5     XKER1 = XINF                                                 MML10540
          XKEI1 = XINF                                                 MML10550
          IER = 34                                                     MML10560
          GO TO 9000                                                   MML10570
   10     BEI1 = ZERO                                                  MML10580
          BER1 = ZERO                                                  MML10590
          XKER1 = ZERO                                                 MML10600
          XKEI1 = ZERO                                                 MML10610
          IER = 129                                                    MML10620
```

```
      IF (X .GT. ZERO) GO TO 9000                                       MML10630
      XKER1 = XINF                                                      MML10640
      XKEI1 = XINF                                                      MML10650
      GO TO 9000                                                        MML10660
C                                 ARGUMENT IS 0.0                       MML10670
   15 BEI1 = ZERO                                                       MML10680
      BER1 = ZERO                                                       MML10690
      XKER1 = -XINF                                                     MML10700
      XKEI1 = -XINF                                                     MML10710
      GO TO 9005                                                        MML10720
 9000 CONTINUE                                                          MML10730
      CALL UERTST(IER,6HMMKEL1)                                         MML10740
 9005 RETURN                                                            MML10750
      END                                                               MML10760
//GO.SYSIN DD *
 Relative Damage During Captive Flight
 1000 1.0      1.0       2.0
30000000. 800.      0.499     0.30     1.1      0.9      2.44        5.4E-5
    6.0E-6  0.06
160.0       8.0
   11
  -60 5.59
  -40 4.46
  -20 3.47
    0 2.59
   20 1.81
   40 1.18
   60 0.48
   80 -0.08
  100 -0.59
  120 -1.16
  140 -1.48
   30 20.0
```

```
// JOB (A95$X5,516,0.25,10),'GEORGE DERBALIAN'
// EXEC FORTCLG
//FORT.SYSIN DD *
C$WATFIV
C
C               MARKOV
C          GEORGE DERBALIAN
C             APRIL 1981
C THIS PROGRAM DETERMINES THE CUMULATIVE DAMAGE IN ROCKET MOTORS
C BY RANDOMLY ALLOWING THE ROCKETS TO MOVE FROM ONE LOCATION TO ANOTHER
C
      DOUBLE PRECISION RLOC(100)
      REAL*4 D(99),T(100),P(100,100),PERIOD(100),AP(100,100),ROCDAY(100)
      READ(10) ROCDAY
      READ(10) RLOC,P,PERIOD,NIRL
      DO 33 I=1,NIRL
      WRITE(6,616) I
   33 WRITE(6,615) (J,P(I,J),J=1,NIRL)
  615 FORMAT(10(I3,F9.6,1X))
  616 FORMAT(/1X,'ROW NUMBER=',I3)
      WRITE(6,622)
  622 FORMAT(1H1,'RELATIVE PERIOD IN EACH LOCATION')
      WRITE(6,623) (I,PERIOD(I),I=1,NIRL)
  623 FORMAT(1X,I5,3X,E12.5)
      DO 18 I=1,NIRL
   18 PERIOD(I)=PERIOD(I)*24.0
C RLOC ROCKET LOCATION CODE
C P PROBABILITY MATRIX
C PERIOD TIME SPENT IN EACH LOCATION
C NIRL NUMBER OF ROCKET LOCATIONS INCLUDING CAPTIVE FLIGHT
      READ(5,502) NR,MAXTIM
  502 FORMAT(2I10)
      READ(5,501) (D(I),T(I),I=1,NIRL)
  501 FORMAT(2E10.4)
      WRITE(6,621) NR,MAXTIM
  621 FORMAT(1H1,'LOCATION',5X,'DAMAGE',4X,'PERIOD',20X,
     1 'NO ROCKETS=',I5,5X,'MAXTIME=',I5)
      WRITE(6,620) (I,D(I),T(I),I=1,NIRL)
  620 FORMAT(1X,I5,5X,E10.4,2X,E10.4)
C
      DO 11 I=1,NIRL
      AP(I,1)=P(I,1)
      DO 10 J=2,NIRL
   10 AP(I,J)=AP(I,J-1)+P(I,J)
   11 CONTINUE
CC    DO 20 I=1,NIRL
CC 20 WRITE(6,606) (AP(I,J),J=1,NIRL)
CC606 FORMAT(1X,10F10.6)
      NRL=NIRL-1
      DO 12 I=1,NRL
      IF (P(I,NIRL).EQ.1.0) GO TO 12
      DO 13 J=1,NRL
   13 P(I,J)=P(I,J)/(1.0-P(I,NIRL))
   12 CONTINUE
      DO 14 I=1,NRL
      DO 14 J=2,NRL
   14 P(I,J)=P(I,J-1)+P(I,J)
C
      C=0.
      DO 15 I=1,NIRL
```

```
   15 C=ROCDAY(I)+C
      DO 16 I=1,NIRL
   16 ROCDAY(I)=ROCDAY(I)/C
      WRITE(6,602) C
  602 FORMAT(1H1,'TOTAL ROCKET TIME ',E12.5,/1H1,1X,'CDF OF INITIAL '
     $'LOCATION')
      DO 19 I=2,NIRL
   19 ROCDAY(I)=ROCDAY(I-1)+ROCDAY(I)
      ISEED=135792867
      WRITE(6,624) (I,ROCDAY(I),I=1,NIRL)
  624 FORMAT(1X,I5,1X,F10.6)
      DO 1 IR=1,NR
COMPUTE RANDOMLY INITIAL LOCATION
    5 I=0
      CALL RANDK(ISEED,X,0)
   17 I=I+1
      IF (I.GT.NIRL) STOP
      IF (X.GT.ROCDAY(I)) GO TO 17
      IF (I.EQ.NIRL) GO TO 5
      LO=I
C .................................
      NLOC=0
      NCAP=0
      TIME=0.0
      DMG=0.0
      L=LO
    2 CALL RANDOM(AP,NIRL,L,ISEED)
      IF (L.EQ.0) GO TO 6
      IF (L.NE.NIRL) GO TO 3
      TIME=TIME+PERIOD(L)
      DMG=DMG+D(L)*PERIOD(L)/T(L)
      WRITE(8) DMG,TIME,L,IR
      L=LO
      NCAP=NCAP+1
      IF (TIME.GT.MAXTIM) GO TO 4
      GO TO 2
    6 DMG=DMG+D(LO)/T(LO)*(MAXTIM-TIME)
      GO TO 4
    3 TIME=TIME+PERIOD(LO)
      NLOC=NLOC+1
      DMG=DMG+D(LO)*PERIOD(LO)/T(LO)
      WRITE(8) DMG,TIME,LO,IR
      LO=L
      IF (TIME.LT.MAXTIM) GO TO 2
    4 WRITE(6,601) IR,DMG,TIME,NLOC,NCAP
  601 FORMAT(1X,'IR=',I5,5X,'DMG=',E12.5,5X,'TIME=',F10.0,2X,'NLOC=',I6,
     $ 2X,'NCAP=',I6)
    1 CONTINUE
      STOP
      END
C
C *****************************************************
C
      SUBROUTINE RANDOM(AP,N,L,ISEED)
      REAL*4 AP(100,100),X,XP
      I=0
      CALL RANDK(ISEED,X,0)
    3 I=I+1
CC    WRITE(6,601) I,X
CC601 FORMAT(1X,'I=',I5,1X,'X=',F10.6)
```

```
      IF(I.GT.N) GO TO 4
      XP=AP(L,I)
      IF(X.GT.XP) GO TO 3
      L=I
      RETURN
    4 WRITE(6,601) L
  601 FORMAT(/1X,I5,2X,'*ILL DEFINED CDF*')
      L=0
      RETURN
      END
C
C****************************************************
C
      SUBROUTINE RANDK (IY, YFL, INDEX)
C
C     THIS IS A UNIFORM RANDOM NUMBER GENERATOR WRITTEN BY G. E.
C  FORSYTHE IN SPRING 1969, FOLLOWING D. KNUTH, THE ART OF COMPUTER
C  PROGRAMMING, VOL. 2, PP. 155-156.  IT IS MUCH SUPERIOR TO RANDU,
C  THE RANDOM NUMBER GENERATOR FOUND IN IBM'S SCIENTIFIC SUBROUTINE
C  PACKAGE.
C
C     BEFORE THE FIRST CALL OF RANDK, IY SHOULD BE SET OUTSIDE RANDK
C  TO AN ARBITRARY INTEGER VALUE.  (IN WATFOR THIS IS ESSENTIAL.)
C  FOR PROGRAM CHECKOUT, THE INITIAL VALUE OF IY SHOULD BE A FIXED
C  INTEGER.  FOR RANDOM NUMBERS DIFFERENT ON EVERY RUN (AND HENCE
C  NOT REPRODUCIBLE), DECLARE INTEGER CLOCK1 AND THEN INITIALIZE
C  IY TO CLOCK1(4).
C
C     IF RANDK IS CALLED WITH AN INTEGER INDEX = 1, THEN THE OUTPUT
C  VALUE OF IY IS A PSEUDORANDOM INTEGER UNIFORMLY DISTRIBUTED IN THE
C  RANGE 0 <= IY < 2**31.
C
C     IF RANDK IS CALLED WITH INDEX = 0, THEN NOT ONLY IS IY PRODUCED,
C  BUT ALSO (AT SOME EXTRA COST IN TIME) A FLOATING NUMBER YFL, UNI-
C  FORMLY DISTRIBUTED IN THE INTERVAL 0.0 <= YFL < 1.0.
C
      IY = IY*314159269 + 453806245
    4 IF (IY .GE. 0) GO TO 6
C
C  CAUTION:  THE STATEMENT LABEL 4 IS ESSENTIAL IN ORDER TO PREVENT
C  CERTAIN COMPILERS (E.G., FORTRAN H WITH OPT 0) FROM PERFORMING
C  UNWANTED "OPTIMIZATIONS."  IT SHOULD NOT BE REMOVED.
C
    5    IY = IY + 2147483647 + 1
C        STATEMENT 5 ADDS 2**31 TO NEGATIVE VALUES OF IY
C
    6 IF (INDEX) 7, 7, 8
C
    7    YFL = IY
         YFL = YFL*.4656613E-9
C
    8 RETURN
      END
//GO.FT08F001 DD DSN=WYL.X5.A95.MONTE.CARLO.DAMAGE,UNIT=DISK,
// DISP=(,CATLG),DCB=(RECFM=VBS,BLKSIZE=3200),SPACE=(TRK,(10,5),RLSE)
//GO.FT10F001 DD DSN=WYL.X5.A95.PROB,DISP=SHR
//GO.SYSIN DD *
      1000     87600
  0.210E-06 87600.
 0.         87600.
```

203.	0.	87600.
204.	0.407E- 9	87600.
205.	0.573E-16	87600.
206.	0.0	87600.
207.	0.573E-16	87600.
208.	0.573E-16	87600.
209.	0.573E-16	87600.
210.	0.573E-16	87600.
211.	0.573E-16	87600.
212.	0.	87600.
213.	0.298E- 7	87600.
214.	0.440E- 9	87600.
215.	0.459E- 9	87600.
216.	0.479E- 9	87600.
217.	0.305E- 7	87600.
218.	0.573E-16	87600.
219.	0.0	87600.
220.	0.573E-16	87600.
221.	0.573E-16	87600.
222.	0.573E-16	87600.
223.	0.337E- 6	87600.
224.	0.573E-16	87600.
225.	0.573E-16	87600.
226.	0.185E- 4	87600.
227.	0.662E- 9	87600.
228.	0.573E-16	87600.
229.	0.573E-16	87600.
230.	0.573E-16	87600.
231.	0.573E-16	87600.
232.	0.0	87600.
233.	0.161E-16	87600.
234.	0.547E-16	87600.
235.	0.573E-16	87600.
236.	0.573E-16	87600.
237.	0.268E- 9	87600.
238.	0.180E- 7	87600.
239.	0.573E-16	87600.
240.	0.383E- 8	87600.
241.	0.573E-16	87600.
242.	0.290E- 6	87600.
243.	0.	87600.
244.	0.573E-16	87600.
245.	0.573E-16	87600.
246.	0.0	87600.
247.	0.573E-16	87600.
248.	0.	87600.
249.	0.573E-16	87600.
250.	0.176E- 6	87600.
251.	0.	87600.
252.	0.534E- 9	87600.
253.	0.	87600.
254.	0.	87600.
255.	0.217E- 7	87600.
256.	0.573E-16	87600.
257.	0.0	87600.
258.	0.573E-16	87600.
259.	0.111E- 6	87600.
260.	0.573E-16	87600.
261.	0.573E-16	87600.
262.	0.	87600.
263.	0.573E-16	87600.
264.	0.573E-16	87600.
265.	0.0	87600.
266.	0.	87600.
267.	0.	87600.
268.	0.	87600.
269.	0.1314	87600.

```
//CDFLMOD JOB
// EXEC FORTCL
//FORT.SYSIN DD *
C
C  WEATHER
C PROGRAM TO READ TEMPERATURE DATA FROM A TDF-14 SURFACE OBSERVATIONS
C TAPE, AND SAVE A C.D.F. OF DAILY AND ANNUAL TEMPERATURE AMPLITUDES
C
C PAUL R. JOHNSTON AND GEORGE DERBALIAN
C 09-20-1980
C
      IMPLICIT INTEGER*4 (A-Z)
      REAL*4 RAMP,RCDF
      COMMON ITEMP(24),CDFDAY(200),CDFYR(200),DAY(31),CTEMP(24),
     2MONTH(12),YEAR,LDAY,LMONTH,NDAY,NMONTH,NYEAR,NSN,NTDN
C
C INITIALIZE VARIABLES
C
      LDAY=0
      DO 1 I=1,200
      CDFDAY(I)=0
    1 CDFYR(I)=0
      DO 2 I=1,12
    2 MONTH(I)=999
      YEAR=0
C
C READ AND PRINT LOCATION AND START DATE
C
      READ(11,1101)NTDN,NSN,NYEAR,NMONTH,NDAY
      REWIND 11
      WRITE(8,801)NTDN,NSN,NYEAR,NMONTH,NDAY
C
C READ TEMPERATURES FROM TAPE, ONE DAY AT A TIME
C
  100 READ(11,1102,END=99)NTDN,NSN,NYEAR,NMONTH,NDAY,(ITEMP(I),
     2CTEMP(I),I=1,6)
      READ(11,1103,END=99)(ITEMP(I),CTEMP(I),I=7,12)
      READ(11,1103,END=99)(ITEMP(I),CTEMP(I),I=13,18)
      READ(11,1103,END=99)(ITEMP(I),CTEMP(I),I=19,24)
C
C COMPUTE MONTHLY AVARAGE IF IT IS THE END OF A MONTH
C
      IF (NDAY.EQ.1) CALL EMONTH
C
C DECODE TEMPERATURES
C
      DO 3 I=1,24
    3 CALL SIGNCK(ITEMP(I),CTEMP(I))
C
C CALCULATE THE DAILY AMPLITUDE AND AVERAGE
C
      TSUM=0
      TMIN=1000
      TMAX=-1000
      N=0
      DO 4 I=1,24
      T=ITEMP(I)
      IF (T.EQ.999) GO TO 4
      TSUM=TSUM+T
      IF (T.GT.TMAX) TMAX=T
```

```
      IF (T.LT.TMIN) TMIN=T
      N=N+1
    4 CONTINUE
      IF (N.EQ.0) GO TO 5
      DAY(NDAY)=1.*TSUM/N+0.5
      AMP=TMAX-TMIN+1
      IF (AMP.GT.200) GO TO 6
      CDFDAY(AMP)=CDFDAY(AMP)+1
      GO TO 6
    5 DAY(NDAY)=999
    6 CONTINUE
      LDAY=NDAY
      LMONTH=NMONTH
      GO TO 100
   99 CALL EMONTH
C
C PRINT FINAL DATE
C
      WRITE(8,802)NYEAR,NMONTH,NDAY
C
C CALCULATE NORMALIZATION CONSTANTS
C
      NA=0
      ND=0
      DO 7 I=1,200
      NA=NA+CDFYR(I)
      ND=ND+CDFDAY(I)
    7 CONTINUE
      IF ((ND.EQ.0).OR.(NA.EQ.0)) GO TO 98
      YEAR=1.*YEAR/NA+0.5
      WRITE(8,803)YEAR
C
C PRINT ANNUAL AMPLITUDE C.D.F.
C
      WRITE(8,804)
      RAMP=-0.5
      RCDF=0.0
      DO 8 I=1,200
      RAMP=RAMP+0.5
      RCDF=RCDF+1.*CDFYR(I)/NA
      WRITE(8,805)RAMP,RCDF
    8 CONTINUE
C
C PRINT DAILY AMPLITUDE C.D.F.
C
      WRITE(8,806)
      RAMP=-0.5
      RCDF=0.0
      DO 9 I=1,200
      RAMP=RAMP+0.5
      RCDF=RCDF+1.*CDFDAY(I)/ND
      WRITE(8,807)RAMP,RCDF
    9 CONTINUE
      STOP
C
C INSUFFICIENT DATA ON THE TAPE
C
   98 WRITE(8,808)
      STOP
  801 FORMAT(I5,' TAPE DECK NUMBER',/,I5,' STATION NUMBER',/,I5,
```

```
     2' FIRST YEAR',/,I5,' FIRST MONTH',/,I5,' FIRST DAY')
  802 FORMAT(I5,' LAST YEAR',/,I5,' LAST MONTH',/,I5,' LAST DAY')
  803 FORMAT(I5,' AVERAGE TEMPERATURE')
  804 FORMAT('ANNUAL TEMPERATURE AMPLITUDE C.D.F.')
  805 FORMAT(F10.2,F10.6)
  806 FORMAT('DAILY TEMPERATURE AMPLITUDE C.D.F.')
  807 FORMAT(F10.2,F10.6)
  808 FORMAT('INSUFFICIENT DATA IN ANY YEAR TO COMPUTE AVERAGE')
 1101 FORMAT(I4,I5,3I2)
 1102 FORMAT(I4,I5,3I2,6(15X,I2,A1,62X))
 1103 FORMAT(15X,6(15X,I2,A1,62X))
      END
      SUBROUTINE SIGNCK(IFLD,ISGN)
C
C SUBROUTINE TO DECODE TEMPERATURES
C
      IMPLICIT INTEGER*4 (A-Z)
      DIMENSION NUM(10),IP(10),MIN(10)
      DATA IP/'A','B','C','D','E','F','G','H','I',ZC0/
      DATA MIN/'J','K','L','M','N','O','P','Q','R',ZD0/
      DATA NUM/1,2,3,4,5,6,7,8,9,0/,IAST/'*'/
      IF (ISGN.EQ.IAST) GO TO 16
      DO 14 K=1,10
      IF (ISGN.EQ.MIN(K)) GO TO 22
      IF (ISGN.EQ.IP(K)) GO TO 20
   14 CONTINUE
   16 IFLD=999
      RETURN
   20 IFLD=IFLD*10+NUM(K)
      RETURN
   22 IFLD=-(IFLD*10+NUM(K))
      RETURN
      END
      SUBROUTINE EMONTH
C
C SUBROUTINE TO PROCESS TEMPERATURE DATA AT THE END OF A MONTH
C
      IMPLICIT INTEGER*4 (A-Z)
      COMMON ITEMP(24),CDFDAY(200),CDFYR(200),DAY(31),CTEMP(24),
     2MONTH(12),YEAR,LDAY,LMONTH,NDAY,NMONTH,NYEAR,NSN,NTDN
      IF (LDAY.EQ.0) RETURN
C
C CALCULATE THE AVERAGE MONTHLY TEMPERATURE
C
      TSUM=0
      N=0
      DO 1 I=1,LDAY
      T=DAY(I)
      IF (T.EQ.999) GO TO 1
      TSUM=TSUM+T
      N=N+1
    1 CONTINUE
      IF (N.GT.0) MONTH(LMONTH)=1.*TSUM/N+0.5
      IF (N.EQ.0) MONTH(LMONTH)=999
      IF (LMONTH.LT.12) RETURN
C
C CALCULATE THE ANNUAL AMPLITUDE AND AVERAGE
C
      TSUM=0
      TMIN=1000
```

```
          TMAX=-1000
          N=0
          DO 2 I=1,12
          T=MONTH(I)
          IF (T.EQ.999) GO TO 2
          TSUM=TSUM+T
          IF (T.GT.TMAX) TMAX=T
          IF (T.LT.TMIN) TMIN=T
          N=N+1
        2 CONTINUE
          IF (N.EQ.0) GO TO 3
          YEAR=YEAR+1.*TSUM/N+0.5
          AMP=TMAX-TMIN+1
          IF (AMP.GT.200) GO TO 3
          CDFYR(AMP)=CDFYR(AMP)+1
        3 DO 4 I=1,12
        4 MONTH(I)=999
          RETURN
          END
//LKED.SYSLMOD DD UNIT=DISK,VOL=SER=PUB003,DISP=(NEW,KEEP),
// DSN=WYL.X5.A95.CHINA(CDFGEN),SPACE=(TRK,(3,1,1),RLSE)
```

```
// JOB (A95$X5,516,0.25,10),'GEORGE DERBALIAN',REGION=512K
/*SETUP T=1 INPUT=AU0523
// EXEC FORTCLG
//FORT.SYSIN DD *
C
C        LOGISTIC
C    GEORGE DERBALIAN
C        APRIL 1981
C THIS PROGRAM DETERMINES THE PROBABILITY TRANSITION (MARKOV) MATRIX
C FOR ANY ROCKET SYSTEM USING THE FLTAC ROCKET MOTOR HISORY TAPE AS INPUT
C
      INTEGER YEAR,MONTH,DAY,Y,M,D
      DOUBLE PRECISION MRL(100),LOC,RLOC(100),RCT,RCTO,LOCO,CF,SR,SRO
      DIMENSION INDEX(100),P(100,100),TIME(100)
      DATA CF/'CF      '/,EXP/'EXP'/
      NRL=0
   10 READ(15,501,END=99) LOC
      READ(15,502,END=99)
      READ(15,502,END=99)
      READ(15,502,END=99)
  501 FORMAT(23X,A5)
  502 FORMAT(1X)
      IF (NRL.EQ.0) GO TO 1
      DO 2 J=1,NRL
      IF (LOC.EQ.RLOC(J)) GO TO 10
    2 CONTINUE
    1 NRL=NRL+1
      IF (NRL.GE.100) WRITE(6,610) NRL
  610 FORMAT(/1X,'NUMBER OF ROCKET LOCATIONS EXCEEDING ARRAY SIZE',I4)
      RLOC(NRL)=LOC
      WRITE(6,601) NRL,RLOC(NRL)
      GO TO 10
  601 FORMAT(I5,1X,A5)
   99 DO 3 I=1,100
      TIME(I)=0.0
      DO 4 J=1,100
    4 P(J,I)=0.0
    3 CONTINUE
C
      WRITE(6,607)
  607 FORMAT('1ROCKET LOCATION CODES')
      DO 8 I=1,NRL
    8 INDEX(I)=I
      K=0
    9 READ(5,500,END=88) I,INDEX(I)
      K=K+1
      GO TO 9
  500 FORMAT(2I3)
   88 NIRL=NRL-K
C
      K=0
      DO 11 I=1,NRL
      J=INDEX(I)
      IF (J.EQ.I) GO TO 12
      INDEX(I)=INDEX(J)
      GO TO 11
   12 K=K+1
      INDEX(I)=K
      WRITE(6,601) K,RLOC(I)
      MRL(K)=RLOC(I)
```

```
   11 CONTINUE
C
      XMISS=0.
      REWIND 15
      READ(15,503,END=98) SR,YEAR,MONTH,DAY,LOC,RCT,OPC
  503 FORMAT(1X,A8,7X,3I2,1X,A5,A8,1X,A3)
      READ(15,504,END=98) CFH
  504 FORMAT(35X,F10.0)
      READ(15,502,END=98)
      READ(15,502,END=98)
   21 SRO=SR
      Y=YEAR
      M=MONTH
      D=DAY
      LOCO=LOC
      RCTO=RCT
      CFHO=CFH
      T=(79-Y)*365+(8-M)*30
      READ(15,503,END=98) SR,YEAR,MONTH,DAY,LOC,RCT,OPC
      READ(15,504,END=98) CFH
      READ(15,502,END=98)
      READ(15,502,END=98)
      IF (SR.NE.SRO) GO TO 24
      T=(YEAR-Y)*365+(MONTH-M)*30+(DAY-D)
   24 DO 22 I=1,NRL
      CALL MAP(RLOC,RLOC(I),INDEX,J,NRL)
      IF (RLOC(I).EQ.LOCO) GO TO 23
   22 CONTINUE
   23 IF (OPC.NE.EXP) TIME(J)=TIME(J)+T
      IF (RCT.NE.CF) GO TO 27
      P(J,NIRL+1)=P(J,NIRL+1)+1.
      IF (CFH.LE.0.0) XMISS=XMISS+1.
      TIME(NIRL+1)=TIME(NIRL+1)+CFH/24.0
      TIME(J)=TIME(J)-CFH/24.
   27 IF (SR.NE.SRO) GO TO 21
      DO 25 I=1,NRL
      CALL MAP(RLOC,RLOC(I),INDEX,K,NRL)
      IF (RLOC(I).EQ.LOC) GO TO 26
   25 CONTINUE
   26 IF (J.NE.K) P(J,K)=P(J,K)+1.
      GO TO 21
   98 IF (OPC.NE.EXP) TIME(K)=TIME(K)+T
      XNT=0
      DO 34 I=1,NIRL
   34 XNT=XNT+P(I,NIRL+1)
      TIME(NIRL+1)=TIME(NIRL+1)*XNT/(XNT-XMISS)
      NIRL=NIRL+1
      WRITE(10) TIME
      WRITE(6,602)
  602 FORMAT('1NUMBER OF DAYS SPENT IN EACH LOCATION')
      WRITE(6,603) (I,TIME(I),I=1,NIRL)
  603 FORMAT(I5,1X,E12.4)
      TIME(NIRL)=TIME(NIRL)/XNT
      NM=NIRL-1
      DO 30 I=1,NM
      XK=0.
      DO 31 J=1,NM
   31 XK=XK+P(I,J)
      XKP=XK+P(I,NIRL)
      IF (XK.NE.0.) GO TO 36
```

```
      WRITE(6,611) I
  611 FORMAT(1X,'*ABSORBING STATE*',I5)
      P(I,I)=1.0
      GO TO 30
   36 TIME(I)=TIME(I)/XK
      DO 32 L=1,NIRL
   32 P(I,L)=P(I,L)/XKP
   30 CONTINUE
      WRITE(6,608)
  608 FORMAT('1RELATIVE TIME SPENT IN EACH LOCATION')
      WRITE(6,603) (I,TIME(I),I=1,NIRL)
      WRITE(6,604)
  604 FORMAT('1THE PROBABILITY MATRIX')
      DO 33 I=1,NIRL
      WRITE(6,606) I
   33 WRITE(6,605) (J,P(I,J),J=1,NIRL)
  605 FORMAT(10(I3,F9.6,1X))
  606 FORMAT(/1X,'ROW NUMBER=',I3)
      WRITE(10) MRL,P,TIME,NIRL
      STOP
      END
C
C ************************************
C
      SUBROUTINE MAP(RLOC,LOC,INDEX,N,NRL)
      DOUBLE PRECISION RLOC(100),LOC
      INTEGER INDEX(100)
      DO 1 I=1,NRL
      J=I
    1 IF (RLOC(I).EQ.LOC) GO TO 2
    2 N=INDEX(J)
      RETURN
      END
//GO.FT15F001 DD VOL=SER=AU0523,LABEL=(1,,,IN),UNIT=T6250,DISP=SHR,
// DCB=(RECFM=FB,BLKSIZE=15600,LRECL=78),DSN=WYL.X5.A95.SIDE.WINDER
//GO.FT10F001 DD DSN=WYL.X5.A95.PROB,DISP=(NEW,CATLG),UNIT=DISK,
// DCB=(RECFM=VBS,BLKSIZE=1000),SPACE=(TRK,(3,1),RLSE)
//GO.SYSIN DD *
  6  1
 27  2
 56  8
 86  9
 85 11
 13 11
 33 15
 16 15
 77 19
 60 26
 81 26
 31 30
 55 30
 64 30
 70 30
 67 36
 80 68
 76 74
/*
```

```
$WATFIV
C
C     AIRCARRY
C
      LOGICAL*1 A(80)
      REAL*8 CDFD(50),SUM
      INTEGER NDM(12,25),MAVG(12,25),D,Y,C(6)
      I=1
      READ(5,500) IRSTRT
  500 FORMAT(I1)
C
      DO 10 J=1,50
   10 CDFD(J)=0.0D0
      DO 11 J=1,25
      DO 11 K=1,12
      NDM(K,J)=0
   11 MAVG(K,J)=0
    1 READ(1,501,END=99) M,D,Y,MIN,MAX,C
  501 FORMAT(2X,3I2,27X,I3,3X,I3,6X,6A3,12X)
      WRITE(6,601) I,M,D,Y,MIN,MAX,C
  601 FORMAT(1X,I5,2X,3I3,5X,I3,3X,I3,5X,6A3)
C 1960 IS THE FIRST YEAR CONSIDERED
      Y=Y-59
      IF (Y.LE.0 .OR. Y.GT.25) GO TO 1
      IF (MAX.LT.MIN) GO TO 1
      ID=(MAX-MIN)/2
      CDFD(ID+1)=CDFD(ID+1)+1.0D0
      IAVG=(MAX+MIN)/2
      NDM(M,Y)=NDM(M,Y)+1
      MAVG(M,Y)=MAVG(M,Y)+IAVG
      I=I+1
      GO TO 1
   99 SUM=0.0D0
      DO 2 I=1,50
    2 SUM=SUM+CDFD(I)
      DO 3 I=1,50
    3 CDFD(I)=CDFD(I)/SUM
      DO 4 I=2,50
    4 CDFD(I)=CDFD(I-1)+CDFD(I)
      DO 7 I=1,50
      J=I-1
    7 WRITE(6,602) J,CDFD(I)
  602 FORMAT(I5,E15.7)
      DO 6 J=1,25
      DO 5 I=1,12
      IF (NDM(I,J).EQ.0) GO TO 5
      MAVG(I,J)=MAVG(I,J)/NDM(I,J)
    5 CONTINUE
    6 CONTINUE
      WRITE(6,603) ((MAVG(I,J),I=1,12),J=1,25)
  603 FORMAT(1X,12I10)
      STOP
      END
```

```
+VERSION 1.3.0 (01 MAY 80)            SYSTEM/370 FORTRAN H EXTENDED (ENHANCED)    DATE 81.155/17.19.59      PAGE   1
REQUESTED OPTIONS:
OPTIONS IN EFFECT: NAME(MAIN) OPTIMIZE(3) LINECOUNT(60) SIZE(MAX) AUTODBL(NONE)
                   SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT GOSTMT NOXREF ALC NOANSF TERM IBM FLAG(I)

  ISN 0002          REAL*8 DAM(1000)
  ISN 0003          IRO=1
  ISN 0004          PERIOD=2190.
  ISN 0005          TIME=PERIOD
  ISN 0006        6 DO 3 I=1,1000
  ISN 0007        3 DAM(I)=0.
  ISN 0008          DO=0.
  ISN 0009          TO=0
  ISN 0010        1 READ(12,END=88) D,T,LO,IR
  ISN 0011          IF (IR.NE.IRO) GO TO 22
  ISN 0013          IF (DAM(IR).NE.0.) GO TO 1
  ISN 0015        4 IF (T.GE.TIME) GO TO 2
  ISN 0017          DO=D
  ISN 0018          TO=T
  ISN 0019          GO TO 1
  ISN 0020        2 DAM(IR)=DO+(D-DO)*(TIME-TO)/(T-TO)
  ISN 0021          GO TO 1
  ISN 0022       22 DO=0.
  ISN 0023          TO=0.
  ISN 0024          IRO=IR
  ISN 0025          GO TO 4
  ISN 0026       88 CALL VSORTM(DAM,IR)
  ISN 0027          WRITE(6,601) TIME
  ISN 0028      601 FORMAT(1H1,'TIME=',F8.0)
  ISN 0029          WRITE(6,602) (I,DAM(I),I=1,IR)
  ISN 0030      602 FORMAT(5(1X,I5,2X,E11.4))
  ISN 0031          TIME=TIME+PERIOD
  ISN 0032          REWIND 12
  ISN 0033          IF (TIME.LE.87600.) GO TO 6
  ISN 0035          STOP
  ISN 0036          END
```

DAMAGE·SORT

The program reads the damage vs. time results from Markov and sorts the damages in an ascending order at every three-month interval.

```
*VERSION 1.3.0 (01 MAY 80)
REQUESTED OPTIONS:
OPTIONS IN EFFECT:    SYSTEM/370 FORTRAN H EXTENDED (ENHANCED)    DATE 81.155/17.20.02

NAME(MAIN) OPTIMIZE(3) LINECOUNT(60) SIZE(MAX) AUTODBL(NONE)
SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT GOSTMT NOXREF ALC NOANSF TERM IBM FLAG(I)


               C     SUBROUTINE VSORTM (A,LA)                                          VSOM0010
               C                                                                       VSOM0020
               C-VSORTM--------D-------LIBRARY 1---------------------------------------VSOM0030
               C-VSORTA                                                                VSOM0040
               C                                                                       VSOM0050
               C   FUNCTION     VSORTM - SORT ARRAYS BY ABSOLUTE VALUE                 VSOM0060
               C                VSORTA - SORT ARRAYS BY ALGEBRAIC VALUE                VSOM0070
               C   USAGE               - CALL VSORTM (A,LA)                            VSOM0080
               C                       - CALL VSORTA (A,LA)                            VSOM0090
               C   PARAMETERS   A      - ON INPUT, CONTAINS THE ARRAY TO BE SORTED     VSOM0100
               C                         ON OUTPUT, A CONTAINS THE SORTED ARRAY        VSOM0110
               C                LA     - INPUT VARIABLE CONTAINING THE NUMBER OF       VSOM0120
               C                           ELEMENTS IN THE ARRAY TO BE SORTED          VSOM0130
               C   PRECISION           - SINGLE/DOUBLE                                 VSOM0140
               C   LANGUAGE            - FORTRAN                                       VSOM0150
               C-----------------------------------------------------------------------VSOM0160
               C   LATEST REVISION     - DECEMBER 9, 1975                              VSOM0170
               C                                                                       VSOM0180
ISN 0002             SUBROUTINE VSORTM (A,LA)                                          VSOM0190
               C                                                                       VSOM0200
ISN 0003             DIMENSION          A(1),IU(21),IL(21)                             VSOM0210
ISN 0004             DOUBLE PRECISION   A,T,TT                                         VSOM0220
               C                                  FIND ABSOLUTE VALUES OF ARRAY A      VSOM0230
ISN 0005                DO 5 I=1,LA                                                    VSOM0240
ISN 0006                IF (A(I) .LT. 0.0) A(I)=-A(I)                                  VSOM0250
ISN 0008           5    CONTINUE                                                       VSOM0260
               C                                                                       VSOM0270
ISN 0009             ENTRY VSORTA (A,LA)                                               VSOM0280
               C                                                                       VSOM0290
ISN 0010             M=1                                                               VSOM0300
ISN 0011             I=1                                                               VSOM0310
ISN 0012             J=LA                                                              VSOM0320
ISN 0013             R=.375                                                            VSOM0330
ISN 0014          10 IF (I .EQ. J) GO TO 55                                            VSOM0340
ISN 0016          15 IF (R .GT. .5898437) GO TO 20                                     VSOM0350
ISN 0018             R=R+3.90625E-2                                                    VSOM0360
ISN 0019             GO TO 25                                                          VSOM0370
ISN 0020          20 R=R-.21875                                                        VSOM0380
ISN 0021          25 K=I                                                               VSOM0390
               C                                  SELECT A CENTRAL ELEMENT OF THE      VSOM0400
               C                                  ARRAY AND SAVE IT IN LOCATION T      VSOM0410
ISN 0022             IJ=I+(J-I)*R                                                      VSOM0420
ISN 0023             T=A(IJ)                                                           VSOM0430
               C                                  IF FIRST ELEMENT OF ARRAY IS GREATER VSOM0440
               C                                  THAN T, INTERCHANGE WITH T           VSOM0450
ISN 0024             IF (A(I) .LE. T) GO TO 30                                         VSOM0460
ISN 0026             A(IJ)=A(I)                                                        VSOM0470
ISN 0027             A(I)=T                                                            VSOM0480
ISN 0028             T=A(IJ)                                                           VSOM0490
ISN 0029          30 L=J                                                               VSOM0500
               C                                  IF LAST ELEMENT OF ARRAY IS LESS THANVSOM0510
               C                                  T, INTERCHANGE WITH T                VSOM0520
ISN 0030             IF (A(J) .GE. T) GO TO 40                                         VSOM0530
ISN 0032             A(IJ)=A(J)                                                        VSOM0540
ISN 003[illegible]             A(J)=T                                                            VSOM0550
```

```
                                                                         DATE 81.155/17.20.02
     *VERSION 1.3.0 (01 MAY 80)     VSORTM     SYSTEM/370 FORTRAN H EXTENDED (ENHANCED)

ISN 0034              T=A(IJ)                                                      VSOM0560
            C                                  IF FIRST ELEMENT OF ARRAY IS GREATER VSOM0570
            C                                  THAN T, INTERCHANGE WITH T           VSOM0580
ISN 0035              IF (A(I) .LE. T) GO TO 40                                    VSOM0590
ISN 0037              A(IJ)=A(I)                                                   VSOM0600
ISN 0038              A(I)=T                                                       VSOM0610
ISN 0039              T=A(IJ)                                                      VSOM0620
ISN 0040              GO TO 40                                                     VSOM0630
ISN 0041           35 TT=A(L)                                                      VSOM0640
ISN 0042              A(L)=A(K)                                                    VSOM0650
ISN 0043              A(K)=TT                                                      VSOM0660
            C                                  FIND AN ELEMENT IN THE SECOND HALF OFVSOM0670
            C                                  THE ARRAY WHICH IS SMALLER THAN T    VSOM0680
ISN 0044           40 L=L-1                                                        VSOM0690
ISN 0045              IF (A(L) .GT. T) GO TO 40                                    VSOM0700
            C                                  FIND AN ELEMENT IN THE FIRST HALF OF VSOM0710
            C                                  THE ARRAY WHICH IS GREATER THAN T    VSOM0720
ISN 0047           45 K=K+1                                                        VSOM0730
ISN 0048              IF (A(K) .LT. T) GO TO 45                                    VSOM0740
            C                                  INTERCHANGE THESE ELEMENTS           VSOM0750
ISN 0050              IF (K .LE. L) GO TO 35                                       VSOM0760
            C                                  SAVE UPPER AND LOWER SUBSCRIPTS OF   VSOM0770
            C                                  THE ARRAY YET TO BE SORTED           VSOM0780
ISN 0052              IF (L-I .LE. J-K) GO TO 50                                   VSOM0790
ISN 0054              IL(M)=I                                                      VSOM0800
ISN 0055              IU(M)=L                                                      VSOM0810
ISN 0056              I=K                                                          VSOM0820
ISN 0057              M=M+1                                                        VSOM0830
ISN 0058              GO TO 60                                                     VSOM0840
ISN 0059           50 IL(M)=K                                                      VSOM0850
ISN 0060              IU(M)=J                                                      VSOM0860
ISN 0061              J=L                                                          VSOM0870
ISN 0062              M=M+1                                                        VSOM0880
ISN 0063              GO TO 60                                                     VSOM0890
            C                                  BEGIN AGAIN ON ANOTHER PORTION OF    VSOM0900
            C                                  THE UNSORTED ARRAY                   VSOM0910
ISN 0064           55 M=M-1                                                        VSOM0920
ISN 0065              IF (M .EQ. 0) RETURN                                         VSOM0930
ISN 0067              I=IL(M)                                                      VSOM0940
ISN 0068              J=IU(M)                                                      VSOM0950
ISN 0069           60 IF (J-I .GE. 11) GO TO 25                                    VSOM0960
ISN 0071              IF (I .EQ. 1) GO TO 10                                       VSOM0970
ISN 0073              I=I-1                                                        VSOM0980
ISN 0074           65 I=I+1                                                        VSOM0990
ISN 0075              IF (I .EQ. J) GO TO 55                                       VSOM1000
ISN 0077              T=A(I+1)                                                     VSOM1010
ISN 0078              IF (A(I) .LE. T) GO TO 65                                    VSOM1020
ISN 0080              K=I                                                          VSOM1030
ISN 0081           70 A(K+1)=A(K)                                                  VSOM1040
ISN 0082              K=K-1                                                        VSOM1050
ISN 0083              IF (T .LT. A(K)) GO TO 70                                    VSOM1060
ISN 0085              A(K+1)=T                                                     VSOM1070
ISN 0086              GO TO 65                                                     VSOM1080
ISN 0087              END                                                          VSOM1090
```

NOMENCLATURE

a	Radius of cavity inside rocket
a_T	Time temperature shift factor of propellant material
b	Outer radius of propellant
B	is a constant for a given propellant material
c	Outer radius of rocket
C_1	Thermal conductivity of air
C_2	Thermal conductivity of propellant
D	Damage
E_c	Elastic modulus of casing
E_p	Elastic modulus of propellant
$\vec{E}_c$	Case properties
$\vec{E}_p$	Propellant properties
$E_r(\xi)$	Relaxation modulus of propellant
f_a	Scale factor, when multiplied by the ambient seasonal temperature amplitude T_a, gives the skin seasonal temperature amplitude
f_d	Scale factor when multiplied by the ambient diurnal temperature amplitude T_d gives the diurnal skin temperature amplitude
F_a	Seasonal frequency
F_d	Diurnal frequency
k_1	Thermal diffusivity of air

k_2	Thermal diffusivity of propellant
K_t	Stress concentration factor
p_{ij}	Probability Markov matrix
P	is the normalizing term used to define the probability distribution of failures
r	Radial distance
$\vec{r}$	Location
t	Time
t_{fi}	is the time to failure of the specimen is exposed to only the i^{th} stress level
t_o	is the unit value of the time for whatever units are used in measuring t_f
T_a	Seasonal temperature amplitude
T_d	Diurnal temperature am litude
T_m	Mean temperature
$T(\vec{r},t)$	Temperature distribution inside rocket
T_s	Rocket motor skin temperature
T_1	Temperature inside rocket cavity
T_2	Temperature in rocket propellant
α	Thermal expansion coefficient of propellant
α_c	Thermal expansion coefficient of casing
Δti	is the time the specimen is exposed to the i^{th} stress level
ν	Poisson's ratio of propellant

ν_c Poisson's ratio of case

ξ Reduced time

σ_{cr} is the critical true stress, below which no failures are observed

σ_{ij} Stress component

σ_r Radial stress

σ_t is the "true" stress applied to the specimen

σ_{to} is the true stress required to fail the specimen in the time t_o

σ_z Axial stress

σ_θ Hoop stress

ω Frequency

ω_a Annual angular frequency

ω_d Diurnal angular frequency

∇^2 Laplacian operator

INITIAL DISTRIBUTION

13 Naval Air Systems Command
    AIR-00D4 (2)
    AIR-03P2 (1)
    AIR-3021 (1)
    AIR-320 (1)
    AIR-330 (1)
    AIR-5163 (1)
    AIR-536 (1)
    AIR-541 (1)
    AIR-542 (1)
    AIR-5422 (3)
5 Naval Sea Systems Command
    SEA-003 (1)
    SEA-62R2, J. W. Murrin (1)
    SEA-62Z31B (1)
    SEA-99612 (2)
1 Chief of Naval Research, Arlington (ONR-412, R. Junker)
1 Marine Corps Development and Education Command, Quantico (Marine Corps Landing Force Development Center)
1 Air Test and Evaluation Squadron 5
1 Fleet Analysis Center, Naval Weapons Station, Seal Beach, Corona (Library)
2 Naval Air Development Center, Warminster
    Code 8131 (1)
    Code 3014 (1)
1 Naval Air Engineering Center, Lakehurst (1115, Technical Library)
2 Naval Intelligence Support Center
    J. B. Chalk (1)
    H. Ruskie (1)
1 Naval Ocean Systems Center, San Diego (Code 133)
7 Naval Ordnance Station, Indian Head
    J. P. Consaga (1)
    W. Haggerty (1)
    M. N. Jacobs (1)
    J. McDevitt (1)
    J. E. Thornton (1)
    W. H. Vreatt (1)
    Technical Library (1)
3 Naval Postgraduate School, Monterey
    Code 1424, Library Director (1)
    G. Cantin (1)
    G. H. Lindsey (1)
2 Naval Research Laboratory
    Code 6100 (1)
    W. D. Bascom (1)
4 Naval Surface Weapons Center, Dahlgren
    Code CG-33 (1)
    Code DG (1)
    Code DG-50 (1)
    Head, Technical Information Division, Technical Library (1)

2 Naval Surface Weapons Center, White Oak Laboratory, Silver Spring
  J. M. Augl (1)
  Technical Library (1)
1 Naval Underwater Systems Center, Newport (Library Division)
1 Naval Weapons Support Center, Crane (Code 5042)
2 Navy Strategic Systems Project Office
  M. Baron (1)
  Head, Technical Library (1)
1 Office of Naval Research, Pasadena Branch Office (Deputy Director, Chief Scientist)
1 Army Armament Materiel Readiness Command, Rock Island (DRSAR-LEM)
8 Army Armament Research and Development Command, Dover
  DRDAR-TSS
    Scientific & Technical Division, Bldg. 59 (1)
    J. Picard (1)
  B. D. Lehman (1)
  S. Nicolaides (1)
  L. Rosendorf (1)
  G. Silvestro (1)
  A. Taschler (1)
  D. Wiegand (1)
1 Army Development & Readiness Command, Alexandria (DRCDE-DW, S. R. Matos)
5 Army Missile Command, Redstone Arsenal
  DRSMI-R, Dr. R. G. Rhoades (1)
  C. W. Austin, Jr. (1)
  T. H. Duerr (1)
  D. L. Martin, Jr. (1)
  R. E. Siron (1)
1 Army Missile Command, Redstone Scientific Information Center, Redstone Arsenal (DRSMI-RPRD)
5 Army Ballistic Research Laboratories, Aberdeen Proving Ground
  DRDAR-BL I, J. M. Hurban (1)
  DRDAR-BL V, R. Vitali (1)
  A. S. Elder (1)
  J. J. Rocchio (1)
  R. Wires (1)
1 Army Material Systems Analysis Agency, Aberdeen Proving Ground (DRXSY-DE, Security Office)
1 Army Research Office, Research Triangle Park (DRXPO-IP-L, Information Processing Office)
1 Radford Army Ammunition Plant (DARRA-QA)
1 White Sands Missile Range (Technical Library)
2 Air Force Systems Command, Andrews Air Force Base
  AFSC-DLFP, R. Smith (1)
  AFSC-SDZ (1)
1 Air Force Academy, Colorado Springs (W. Siuru)
1 Air Force Armament Division, Eglin Air Force Base (XRC, T. O'Grady)
5 Air Force Armament Laboratory, Eglin Air Force Base
  AFATL/DLD (1)
  AFATL/DLJW (1)
  AFATL/DLMI, Aden (1)
  AFATL/DLO (1)
  AFATL/DLOD, STINFO (1)
1 Air Force Rocket Propulsion Laboratory, Edwards Air Force Base (AFRPL/LK)

1 Air Force Rocket Propulsion Laboratory, Edwards Air Force Base (AFRPL/MK)
1 Air Force Rocket Propulsion Laboratory, Edwards Air Force Base (AFRPL/MKCC)
1 Air Force Rocket Propulsion Laboratory, Edwards Air Force Base (AFRPL/MKP)
1 Air Force Rocket Propulsion Laboratory, Edwards Air Force Base (AFRPL/MKPA, J. L. Trout)
4 Air Force Rocket Propulsion Laboratory, Edwards Air Force Base
  AFRPL/MKPB
    T. J. C. Chew (1)
  C. T. Liu (1)
  R. G. Stacer (1)
  D. I. Thrasher (1)
1 Air Force Rocket Propulsion Laboratory, Edwards Air Force Base (AFRPL/PA)
1 Air Force Rocket Propulsion Laboratory, Edwards Air Force Base (AFRPL/TS, R. A. Biggers)
1 Air Force Rocket Propulsion Laboratory, Edwards Air Force Base (AFRPL/TSPR)
1 Air Force Rocket Propulsion Laboratory, Edwards Air Force Base (AFRPL/XRX, L. Quinn)
1 Air Force Rocket Propulsion Laboratory, Edwards Air Force Base (Technical Library)
4 Air Force Wright Aeronautical Laboratories, Wright-Patterson Air Force Base
  AFWAL/MLTN, C. S. Anderson (1)
  AFWAL/PORA (1)
  AFWAL/PORT (1)
  AFWAL/TST (1)
1 Air University Library, Maxwell Air Force Base (AUL-SE)
2 Foreign Technology Division, Wright-Patterson Air Force Base
  FTD/SDBP (1)
  FTD/TQTA, S. J. Radloff (1)
3 Ogden Air Logistics Center, Hill Air Force Base
  MANPA, A. Inverso (1)
  MMGM, E. F. Lund (1)
  MMWRV, J. K. Scambia (1)
1 San Antonio Air Logistics Center, Kelly Air Force Base (SFQT, W. E. Vande[illegible])
1 Eastern Space and Missile Center, Patrick Air Force Base (PM/Linda M. Adams)
1 Deputy Under Secretary of Defense, Research and Advanced Technology (Director, Engineering Technology)
1 Defense Advanced Research Projects Agency, Arlington, VA
12 Defense Technical Information Center
9 George C. Marshall Space Flight Center, Huntsville
  AS61L (1)
  EP-25, John Miller (1)
  C. J. Bianca (1)
  E. P. Jacobs (1)
  K. W. Jones (1)
  J. Q. Miller (1)
  L. B. Powers (1)
  W. L. Ray (1)
  J. H. Wise (1)
1 Langley Research Center (NASA), Hampton (MS-185, Technical Library)
2 Lewis Research Center (NASA), Cleveland
  D. A. Petrash (1)
  Technical Library (1)
1 Lyndon B. Johnson Space Center (NASA), Houston (Code JM2, Technical Library)

1 National Aeronautics & Space Administration, Scientific and Technical Information Facility, College Park, MD (Technical Library)
1 Aerojet Liquid Rocket Company, Sacramento, CA, Via AFPRO (Head, Technical Information Division)
2 Aerojet Strategic Propulsion Company, Sacramento, CA, Via AFPRO
  K. Bills, Jr. (1)
  S. J. Svob (1)
1 Aerojet Tactical Systems, Sacramento, CA, Via AFPRO (R. Mironenko)
1 Ashland Chemical Company, Dublin, OH (Specialty Fuels Department)
2 Atlantic Research Corporation, Alexandria, VA (Technical Library)
2 California Institute of Technology, Jet Propulsion Laboratory, Pasadena, CA
  R. Bailey, MS/125-224 (1)
  Lib. Op. CP, Lib. Acqs.-Standing Orders (1)
1 Failure Analysis Associates, Palo Alto, CA
1 Ford Aerospace and Communications Corporation, Newport Beach, CA (Aeronutronic Division, Technical Information Services, DDC Acqs.)
1 Hercules Incorporated, Allegany Ballistics Laboratory, Cumberland, MD (Technical Library)
1 Hercules Incorporated, Magna, UT (Library, 100-H-2)
1 Hercules Incorporated, McGregor, TX (Engineering Services Manager)
1 Hughes Aircraft Company, Canoga Park, CA (Missile Systems Division)
1 Hughes Aircraft Company, El Segundo, CA (Technical Document Center, B. W. Campbell, Bldg. E1, MS E 110)
2 Johns Hopkins University, Applied Physics Laboratory, Laurel, MD (Chemical Propulsion Information Agency, Code ML, R. D. Brown)
1 Lockheed Missiles & Space Company, Sunnyvale, CA (Technical Information Center 52-50, Palo Alto, DCS-L201, Reports Acquisition)
1 Martin Marietta Aerospace, Orlando, FL (MP-30, Engineering Library)
1 McDonnell Douglas Corporation, Huntington Beach, CA (A3-135, Library Services)
1 McDonnell Douglas Corporation, St. Louis, MO (Technical Library)
1 Olin Corporation, Marion, IL (Marion Works, R. D. Altekruse)
1 Rockwell International Corporation, Canoga Park, CA (TIC, D/586-103, BA29)
1 Rockwell International Corporation, Downey, CA (TIC-D096-400-AJ01)
1 Rohm and Haas Company, Huntsville Defense Contract Office, Huntsville, AL
1 SRI International, Menlo Park, CA (D. Ross)
1 Sundstrand Corporation, Rockford, IL
3 TRW, Inc., Redondo Beach, CA
  Technical Information Center
    Document Services (2)
    R. C. Reeve (1)
1 Talley Industries of Arizona, Inc., Mesa, AZ (Library)
1 Textron, Inc., Bell Aerospace Textron Division, Buffalo, NY (Technical Library)
1 The Aerospace Corporation, Los Angeles, CA (Library Acquisitions Group)
1 The Boeing Company, Seattle, WA (Kent Library, 8K-38, Library Proc. Supervisor)
1 The Marquardt Company, Van Nuys, CA (Library)
1 Thiokol Chemical Corporation, Elkton, MD
1 Thiokol Chemical Corporation, Huntsville Division, Huntsville, AL (Technical Library)
2 Thiokol Chemical Corporation, Wasatch Division, Brigham City, UT (Technical Library)
1 United Technologies Corporation, Chemical Systems Division, Sunnyvale, CA (Technical Library)
1 United Technologies Research Center, East Hartford, CT (Document Control Station)
1 Vought Corporation, Systems Division, Dallas, TX (Lib-2-58010)

50 Naval Weapons Center, China Lake

- Code 003 (1)
- Code 01 (1)
- Code 012 (1)
- Code 03A (1)
- Code 03A4 (1)
- Code 03T, C. J. Thelan (1)
- Code 12 (1)
- Code 31 (1)
- Code 32 (1)
- Code 3205 (1)
- Code 3205, R. Hoffer (1)
- Code 3241 (1)
- Code 3242 (1)
- Code 3242, T. A. Hicks (1)
- Code 3243 (1)
- Code 3244 (1)
- Code 3246 (1)
- Code 326 (1)
- Code 3263 (1)
- Code 327 (1)
- Code 327B, R. Vetter (5)
- Code 3272 (1)
- Code 3273 (1)
- Code 3274 (1)
- Code 3275 (1)
- Code 33 (1)
- Code 338 (1)
- Code 343 (4) (3 plus archive copy)
- Code 35 (1)
- Code 351 (1)
- Code 36 (1)
- Code 366 (1)
- Code 3662, S. D. Benson (1)
- Code 3686, R. Lloyd (1)
- Code 38 (1)
- Code 385 (1)
- Code 3858
  - A. H. Lepie (1)
  - H. P. Richter (1)
- Code 388 (1)
- Code 3881 (1)
- Code 3882, C. Dinerman (1)
- Code 39 (1)
- Code 621 (1)

DATE

ILME

-8