AEROSPACE REPORT NO.
TR-92(2508)-1

Forecasting the Relativistic Electron Flux at Geosynchronous Orbit

Prepared by

H. C. KOONS and D. J. GORNEY
Space and Environment Technology Center
Technology Operations

15 April 1992

Prepared for

SPACE SYSTEMS DIVISION
AIR FORCE SYSTEMS COMMAND
Los Angeles Air Force Base
P. O. Box 92960
Los Angeles, CA 90009-2960

Programs Group

THE AEROSPACE CORPORATION
El Segundo, California

APPROVED FOR PUBLIC RELEASE;
DISTRIBUTION IS UNLIMITED

This report was submitted by The Aerospace Corporation, El Segundo, CA 90245-4691, under Contract No. F04701-88-C-0089 with the Space Systems Division. P. O. Box 92960, Los Angeles, CA 90009-2960. It was reviewed and approved for The Aerospace Corporation by A. B. Christensen, Principal Director, Space and Environment Technology Center. Capt. Fisher was the project officer.

This report has been reviewed by the Public Affairs Office (PAS) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nationals.

This technical report has been reviewed and is approved for publication. Publication of this report does not constitute Air Force approval of the report's findings or conclusions. It is published only for the exchange and stimulation of ideas.

TRYON K. FISHER, Capt, USAF
MOIE Project Officer

QUANG BUI, Lt, USAF
MOIE Program Manager

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION Unclassified	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution unlimited
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S) TR-92(2508)-1	5. MONITORING ORGANIZATION REPORT NUMBER(S) SSD-TR-92-29
6a. NAME OF PERFORMING ORGANIZATION The Aerospace Corporation Technology Operations 6b. OFFICE SYMBOL *(If applicable)*	7a. NAME OF MONITORING ORGANIZATION Space Systems Division
6c. ADDRESS *(City, State, and ZIP Code)* El Segundo, CA 90245-4691	7b. ADDRESS *(City, State, and ZIP Code)* Los Angeles Air Force Base Los Angeles, CA 90009-2960
8a. NAME OF FUNDING/SPONSORING ORGANIZATION 8b. OFFICE SYMBOL *(If applicable)*	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER F04701-88-C-0089
8c. ADDRESS *(City, State, and ZIP Code)*	10. SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO. | PROJECT NO. | TASK NO. | WORK UNIT ACCESSION NO.

11. TITLE *(Include Security Classification)*

Forecasting the Relativistic Electron Flux at Geosynchronous Orbit

12. PERSONAL AUTHOR(S)

Koons, Harry C., Gorney, David J.

13a. TYPE OF REPORT	13b. TIME COVERED FROM ______ TO ______	14. DATE OF REPORT *(Year, Month, Day)*	15. PAGE COUNT
		1992 April 15	33

16. SUPPLEMENTARY NOTATION

17. COSATI CODES			18. SUBJECT TERMS *(Continue on reverse if necessary and identify by block number)*
FIELD	GROUP	SUB-GROUP	Neural network, Geosynchronous orbit, Relativistic electrons, Space forecast

19. ABSTRACT *(Continue on reverse if necessary and identify by block number)*

A neural network, developed to model the temporal variations of relativistic (> 3 MeV) electrons at geosynchronous orbit, has been used to make reasonably accurate day-ahead forecasts of the relativistic electron flux at geosynchronous orbit. This model can be used to forecast days when internal discharges might occur on geosynchronous satellites or satellites operating within the outer Van Allen radiation belt. The neural network (in essence, a nonlinear prediction filter) consists of three layers of neurons, containing 10 neurons in the input layer, 6 neurons in a hidden layer, and 1 output neuron. The network inputs consist of ten consecutive days of the daily sum of the planetary magnetic index, ΣKp. The output is a prediction of the daily averaged electron flux for the tenth day. The neural network model, together with projections of ΣKp based on its historical behavior, can be used to make the day-ahead forecasts of the relativistic electron flux at geosynchronous orbit. A significantly better forecast is obtained by modifying the network to include one additional input, the measured daily averaged electron flux for the day prior to the forecast day, and one more neuron in the hidden layer. Both models are described in this report.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT	21. ABSTRACT SECURITY CLASSIFICATION
[X] UNCLASSIFIED/UNLIMITED [ ] SAME AS RPT. [ ] DTIC USERS	Unclassified

22a. NAME OF RESPONSIBLE INDIVIDUAL	22b. TELEPHONE *(Include Area Code)*	22c. OFFICE SYMBOL

DD FORM 1473, 84 MAR

83 APR edition may be used until exhausted.
All other editions are obsolete.

SECURITY CLASSIFICATION OF THIS PAGE

UNCLASSIFIED

PREFACE

The authors thank R. Klebesadel of the Los Alamos National Laboratory, Principal Investigator for the SEE instrument, for the electron data. D. Baker and J. B. Blake generated the edited data set of daily averages used in this study. The authors would also like to acknowledge helpful discussions with M. Schulz.

CONTENTS

INTRODUCTION....5

THE NEURAL NETWORK....7

FORECASTS....11

UNNORMALIZED MODEL....11

NORMALIZED MODEL....16

SUMMARY....21

REFERENCES....23

APPENDIX A - Program: FORECAST.C....A-1

APPENDIX B - Program: FORECAST2.C....B-1

FIGURES

1. Electron flux prediction network 7
2. Probability density function for ΣKp for Day 0 plotted parametrically for eight ranges of ΣKp for the previous day, Day -1 12
3. Application of forecasting technique to produce one-day forecasts for the electron flux for 60 days from January 22 through March 22, 1985 13
4. Comparison of measured flux to forecast for one-day forecasts for the electron flux for 60 days from January 22 through March 22, 1985 13
5. Sample output from FORECAST.EXE for 2 July 1984 14
6. Error distribution for the unnormalized model (FORECAST.C) for the entire data set from April 19, 1982 to June 4, 1988 15
7. Sample output from FORECST2.EXE for 2 July 1984 16
8. Error distribution for the normalized model (FORECST2.C) for the entire data set from April 19, 1982 to June 4, 1988 17
9. Error distribution for the normalized model (FORECST2.C) for the 100 days with the largest measured flux from April 19, 1982 to June 4, 1988 18

TABLE

1. Weight matrices and neuron thresholds required to evaluate the electron flux 9

INTRODUCTION

The flux of relativistic (~ MeV) electrons at geosynchronous altitudes shows a strong temporal dependence on epoch relative to the onset of geomagnetic storms (Baker et al., 1979, 1986, 1987; Nagai, 1987, 1988). This electron population has attracted significant attention in recent years, partly because electrical discharges caused by these energetic particles have resulted in anomalous behavior in satellite operations in geosynchronous orbit (Reagan et al., 1983; Vampola, 1987) and on spacecraft operating within the outer Van Allen radiation belt. Quantitative modelling of the temporal behavior of the electron flux can be used as an estimator of the electron flux when direct measureinents are required but are not available. A tenable and accurate forecasting technique would be an especially valuable application. Linear prediction filter techniques (e.g., Nagai, 1988) have shown considerable promise for applying time series of geomagnetic indices as proxy data for the electron flux. The linear techniques provide a simple tool for identifying the times of flux enhancements or dropouts, but they lack the ability to track the magnitude of the electron flux accurately enough for practical applications. We have developed a simple and accurate neural network model (essentially a nonlinear prediction filter) that we have used to study the temporal behavior of the electrons (Koons and Gorney, 1991).

During extended intervals of geomagnetic activity, large fluxes of energetic electrons develop in the outer magnetosphere. After the storm, they diffuse inward enhancing the flux at geosynchronous orbit and in the outer Van Allen radiation belt. These penetrating electrons can become embedded within dielectrics such as printed circuit boards and cable insulation on satellites, building up electrical potentials over time that can exceed the breakdown potential of the dielectric (Meulenberg, 1976; Vampola 1987). Theoretical and experimental results (Wenaas, 1977; Beers, 1977) have shown that breakdowns occur when the fluence of penetrating electrons exceeds ~ 10^{12} cm^{-2} in time periods shorter than the leakage time scales of the dielectric (typically several hours to a few days). Often, these fluence levels are exceeded in geosynchronous orbit several days after major geomagnetic storms. A quantitative forecast of the daily fluence of penetrating electrons at geosynchronous orbit would be quite valuable to the operators of these vehicles.

Superposed epoch analyses have revealed a clear, repeatable pattern in the behavior of the flux of relativistic electrons at geosynchronous orbit. Nagai (1988) showed the dependence of energetic electron flux on geomagnetic activity as measured by the Kp and Dst magnetic indices. The first feature is a rapid decrease in the flux at the onset of a geomagnetic storm. This decrease has been attributed to the combined effects of the geomagnetic field becoming highly distorted (i.e., tail-like) and the convection electric field becoming enhanced at the onset of a geomagnetic storm. The second observed feature is a flux enhancement extending from one to five days following the storm onset, and the final feature is an eventual return to "background" values about ten days after the storm. Nagai (1988) produced a linear prediction model of ~ MeV electron flux based on Kp.

Nagai (1988) related the daily sum of Kp (ΣKp) for 20 consecutive days to the logarithm of the average electron flux (> 2 MeV) for the 20th day. This simple linear scheme proved quite successful in reproducing the general features of the electron flux variations described above. The errors in the logarithm of the flux for this technique were less than 0.5 for about half of the days for which measurements were available. As a characteristic of the linearity of the scheme, the prediction errors tended to be largest (~ one order of magnitude) for the more intense events. Since these events are of most practical interest, an improved prediction procedure using a neural network was developed by Koons and Gorney (1991).

They took a new approach toward modeling and forecasting the flux of energetic electrons in geosynchronous orbit based on input values of Kp. They produced a neural network that successfully reproduces electron flux values based on ten consecutive values of ΣKp. The neural network was developed using BrainMaker, neural network simulation software from California Scientific Software. Neural networks can be trained iteratively to recognize complex and nonlinear patterns in data. The neural network model provides higher accuracy than the linear techniques, especially for large events where quantitative results are of the most practical benefit. Although it is fundamentally more complex than linear prediction filters, the neural network still is simple enough to be implemented on a small personal computer.

THE NEURAL NETWORK

The neural network used for this application consists of three layers of neurons as shown in Figure 1. The 10 neurons comprising the first layer are connected to the input, consisting of the values of ΣKp for ten consecutive days. A ten-day span was chosen because the impulse function obtained by Nagai [1988] from the GMS-3 electron data became essentially zero at a time lag of 10 days. Day 0 is defined to be the day for which the electron flux is calculated. The second layer of neurons, often referred to as the hidden layer, consists of 6 neurons. It is common to construct neural networks such that the number of hidden neurons is half the sum of the number of inputs and outputs. A single neuron is connected to the output, which represents the logarithm of the average flux of electrons for Day 0.

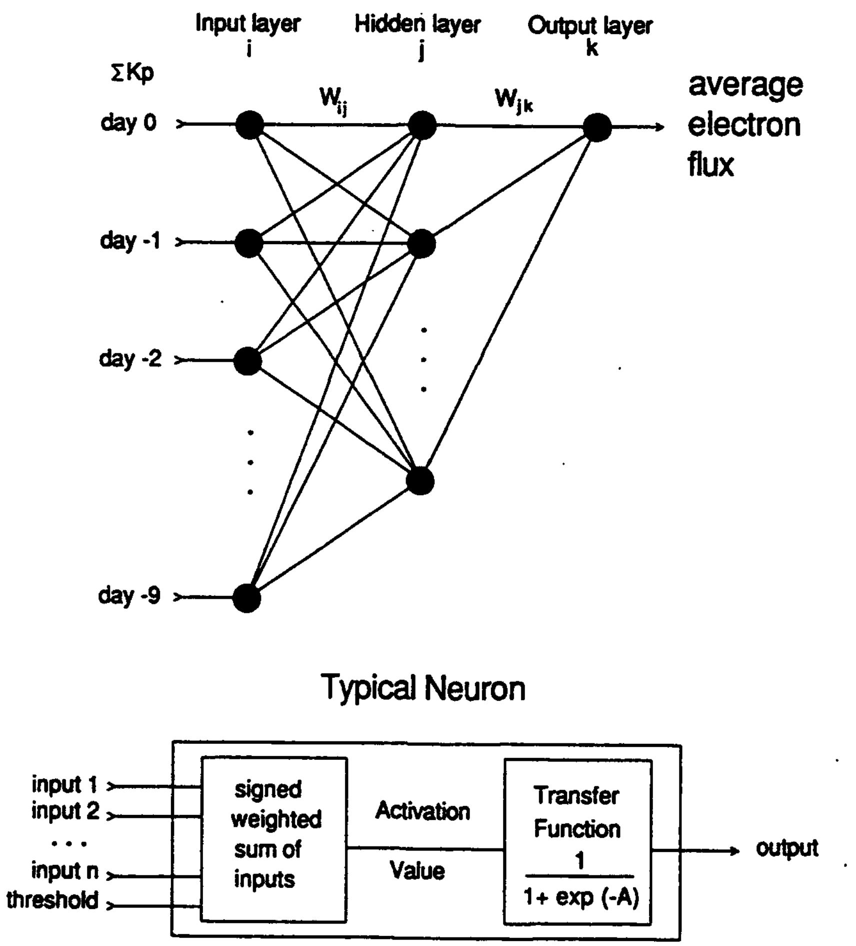

Figure 1. Electron flux prediction network. Diagram shows the structure of the neural network used for predicting the geosynchronous energetic electron flux based on input values of ΣKp. The values W_{ij} and W_{jk} represent weight matrices that couple input and output values to the hidden layer of neurons. The bottom diagram represents the internal function of a typical neuron.

Connections only exist between any single neuron and the neurons in the previous layer of the network. Neurons within a given layer do not connect to each other and do not receive inputs from subsequent layers. For example, neurons in layer 1 send outputs to layer 2, and neurons in layer 2 take inputs from layer 1 and send outputs to layer 3. The connection strengths between any two layers constitute the elements of a real-valued matrix (W). The elemental values W_{ij} represent the connection strength or weight between neuron i (in one layer) to neuron j (in the next higher layer). The weight matrices are modified by training using actual data, and these matrices ultimately contain all of the information relating the input (ΣKp) to the output (the logarithm of the electron flux).

The electron data used to develop the neural network were collected by a SEE (Spectrometer for Energetic Electrons) instrument. The SEE sensor was designed and built by the Los Alamos National Laboratory. For a description of the instrument see Baker et. al. [1986]. This design has flown aboard a number of geostationary satellites. An edited data set covering the period from 19 April 1982 to 4 June 1988 from one spacecraft, 1982-019, was provided for our use. The data set consisted of daily average count rates with background (consisting mainly of galactic cosmic rays) removed.

The network was trained using count rates from the high energy (> 3 MeV) electron channel. The results have been converted to flux using a geometric factor of 0.08 and an efficiency of 0.3 for the 3 MeV channel (J. B. Blake, private communication, 1989). The training data set consisted of 62 days of data from 1 July 1984 to 31 August 1984. The training interval was selected on the basis of data continuity and the occurrence of several discrete flux enhancements within the chosen interval. In order to obtain convergence in the neural network, the training criteria was set at 10% of the complete range of output, corresponding to ~0.5 for the logarithm of the flux or, equivalently, about a factor of three. Training required 2652 passes through the 62 patterns in the training set. The 62 patterns were processed by the network in chronological order. This is not a requirement. A random order might converge more rapidly if there are systematic trends in the data. The calculations were performed on a 16-MHz Compaq Deskpro 386 personal computer in 72 minutes. Once the network is trained, many cases can be run through the network quickly by simply evaluating the functional relationship, which can be written in closed form as (Koons and Gorney, 1991)

$$O_1^3 = \left(1 + \exp\left[-\left\{\Sigma_j 1/\left(1 + \exp\left[-\left\{\Sigma_i I_i^1 W_{ij}^{12} + T_j^2\right\}\right]\right) W_{j1}^{23} + T_1^3\right\}\right]\right)^{-1}$$

The appropriate weight matrices and threshold neuron values are given in Table 1.

Table 1. Weight matrices and neuron thresholds required to evaluate the electron flux from the neural network model given by Eq. 5.

W^{12} =

2.374	-0.639	1.889	1.842	1.216	4.204
0.868	-0.264	2.198	-0.723	-1.853	-1.111
0.790	-2.876	-1.457	0.141	-2.302	-3.078
-1.060	0.605	1.482	-1.812	-2.802	2.245
-1.061	-1.293	-0.649	-0.689	-1.999	-2.245
-0.756	-0.489	2.684	-1.255	-3.711	2.609
4.986	0.369	1.885	-1.571	-2.256	-1.377
-1.358	-0.916	1.143	-1.196	-0.759	-3.052
-2.553	-0.588	-0.197	-2.524	-0.155	-0.903
-0.028	0.723	-3.071	-2.401	-2.857	1.131

T^2 = {0.818   4.236   -0.797   2.582   7.999   -1.890}

W^{23} =

-2.019
1.929
2.464
4.248
-4.000
-5.139

T^3 = 0.077

FORECASTS

UNNORMALIZED MODEL.

The neural network model, together with projections of ΣKp based on its historical behavior, can be used to make day-ahead forecasts of the relativistic electron flux at geosynchronous orbit. This is possible because ΣKp is not a truly random variable and because the electron flux is strongly dependent on recent (1 to 3 days) magnetic activity. We have examined the time series of ΣKp from 1932 to 1988 and we find that there is a strong tendency for quiet and moderately disturbed periods to persist and for violently disturbed periods to be followed by moderately disturbed periods. This behavior is shown in Figure 2 where the probability density function for ΣKp for a given day, here called Day 0, is shown parametrically for ΣKp for the previous day, Day -1. The overall probability that the value of ΣKp on Day 0 is within its most probable bin is 42%, and the overall probability that the value of ΣKp on Day 0 is within ± 1 bin of the most probable value is 86%.

Figure 3 shows a simple application of this forecasting technique to a 60-day period in early 1985. For each day, a set of calculations was performed using the actual values of ΣKp for the preceding 9 days and 10 values of ΣKp from 0 to 72 in steps of 8 for the day to be forecast. The three curves in Figure 3 are, in order from top to bottom, the highest forecast flux, the most probable flux (from the most probable value for ΣKp for Day 0), and the lowest forecast flux. An interesting characteristic of the forecast is that the most probable flux tends to be quite close to the highest forecast flux. The lowest flux is always forecast for a day on which ΣKp has its highest possible value, 72.

Figure 4 compares the flux measured by the SEE instrument for this time period with the most probable flux forecast by this technique. The agreement is excellent. In particular, the most probable flux obtained from the forecast matches or slightly exceeds the measured flux at the peaks, which are the times of most concern to spacecraft operators.

It is fortunate that large magnetic storms (which can not yet be forecast) produce the lowest flux levels on the day they occur. Thus, the time periods of largest error in the forecast are those of least hazard to spacecraft from these relativistic electrons. The neural network model should thus serve as a useful forecasting tool for the large flux levels that are of primary concern to spacecraft operators.

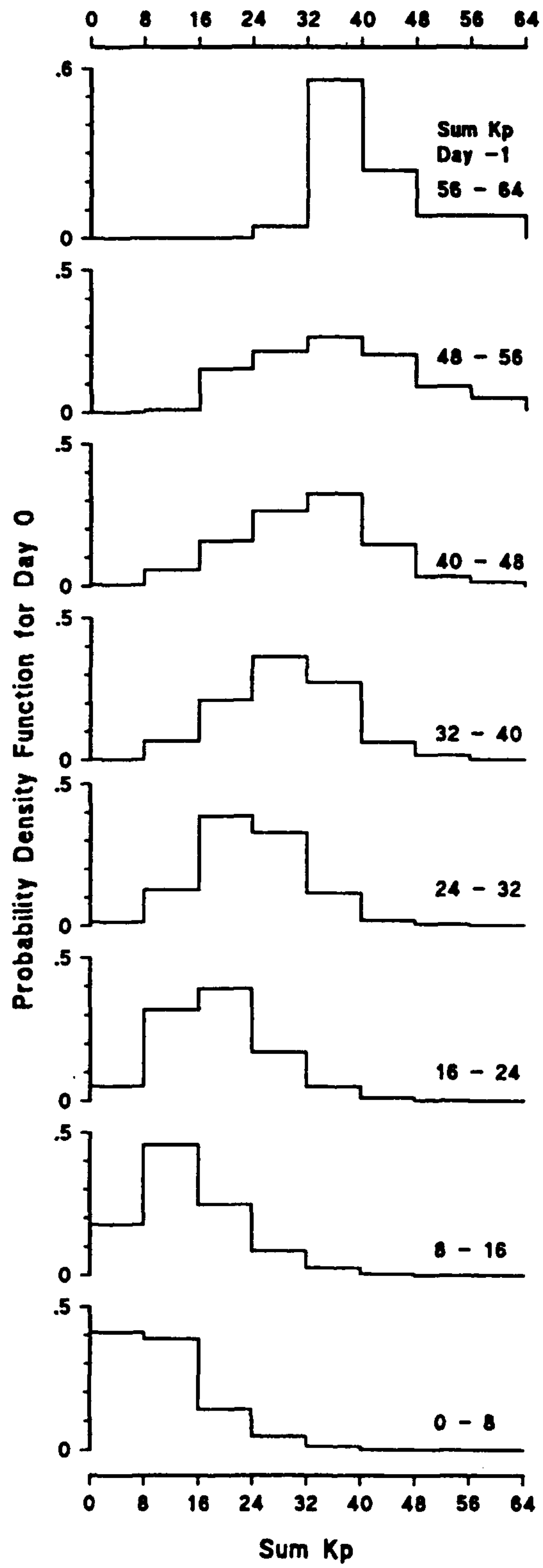

Figure 2. Probability density function for ΣKp for Day 0 plotted parametrically for eight ranges of ΣKp for the previous day, Day -1. These statistics were obtained for the period from January 1, 1932 through June 30, 1988.

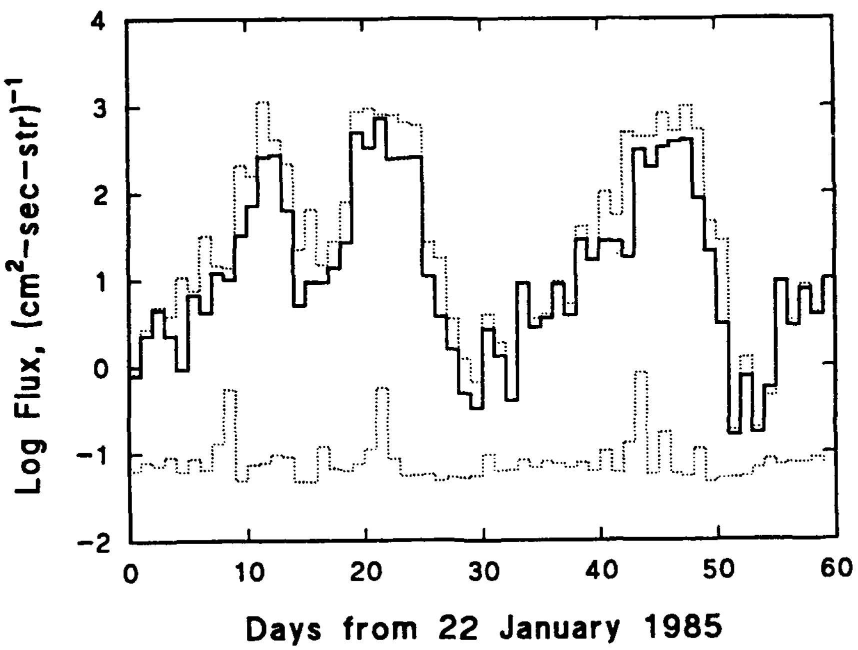

Figure 3. Application of forecasting technique to produce one-day ahead forecasts for the electron flux for 60 days from January 22 through March 22, 1985. The top curve is plotted for the highest flux predicted. The middle curve is the flux predicted for the most probable value of ΣKp for the day of the forecast. The lower curve is the lowest flux predicted. The lowest flux normally results for $\Sigma Kp = 72$ on the day of the forecast.

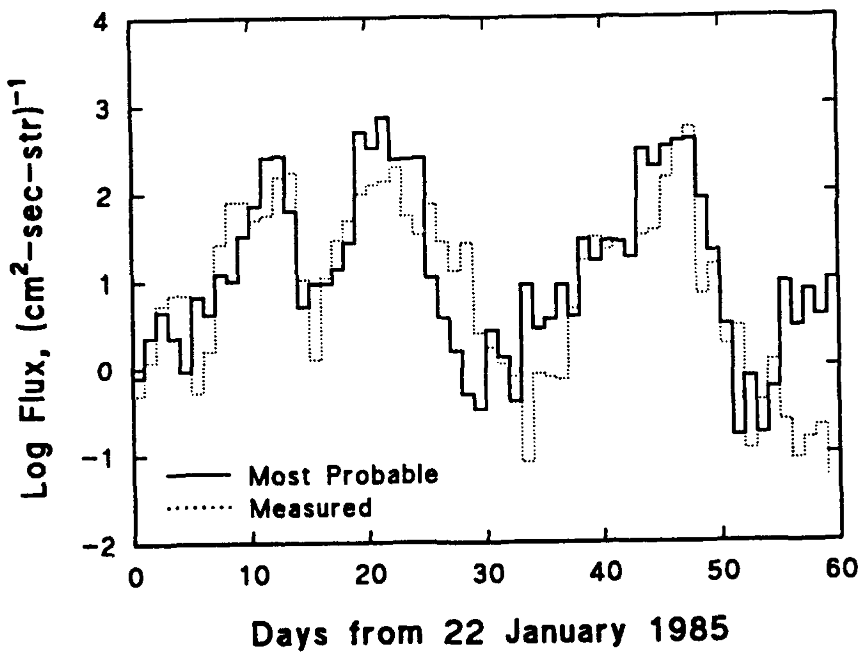

Figure 4. Comparison of measured flux to forecast for one-day ahead forecasts for the electron flux for 60 days from January 22 through March 22, 1985. The solid curve is the flux predicted for the most probable value of ΣKp for the day of the forecast. The dashed curve is the flux obtained from the SEE observations.

This model has been coded for IBM-compatible personal computers. The source code, FORECAST.C, is provided in Appendix A, and a sample output forecasting the flux for 2 July 1984 by the command line:

```
FORECAST 21.0 21.0 20.7 27.9 16.4 17.3 19.6 23.9 15.7 12.4 > PRN
```

is shown in Figure 5.

```
              The Aerospace Neural Network Model for
        Relativistic Electron Flux at Geosynchronous Orbit

                  by H. C. Koons and D. J. Gorney
                     Space Sciences Laboratory
                     The Aerospace Corporation
                       El Segundo, California

Input values (Sum Kp): 21.0 21.0 20.7 27.9 16.4 17.3 19.6
23.9 15.7 12.4

      Log Electron Flux (> 3 MeV) for Day 0 = 0.880468

Predicted Log Electron Flux (> 3 MeV) for Day +1:
Sum Kp =   0 -   8  Log Flux =  1.059  Probability = 0.052
Sum Kp =   8 -  16  Log Flux =  0.911  Probability = 0.319
Sum Kp =  16 -  24  Log Flux =  0.764  Probability = 0.392
Sum Kp =  24 -  32  Log Flux =  0.583  Probability = 0.170
Sum Kp =  32 -  40  Log Flux =  0.314  Probability = 0.050
Sum Kp =  40 -  48  Log Flux = -0.082  Probability = 0.013
Sum Kp =  48 -  56  Log Flux = -0.539  Probability = 0.002
Sum Kp =  56 -  64  Log Flux = -0.903  Probability = 0.001
Sum Kp =  64 -  72  Log Flux = -1.104  Probability = 0.000

      Very High (>2.7) Probability =   0 %
        High (1.7-2.7) Probability =   0 %
Intermediate (0.7-1.7) Probability =  76 %
           Low (< 0.7) Probability =  24 %
```

Figure 5. Sample output from FORECAST.EXE for 2 July 1984.

The overall performance of the network was measured by comparing the model outputs with measured fluxes over the entire ~6-year period from April 19, 1982, to June 4, 1988. The distribution of errors is shown in Figure 6. The distribution is somewhat skewed with a preponderance of cases for which the calculated value from the network model exceeds the measured value. The average log error is -0.41 and the RMS log error is 1.08.

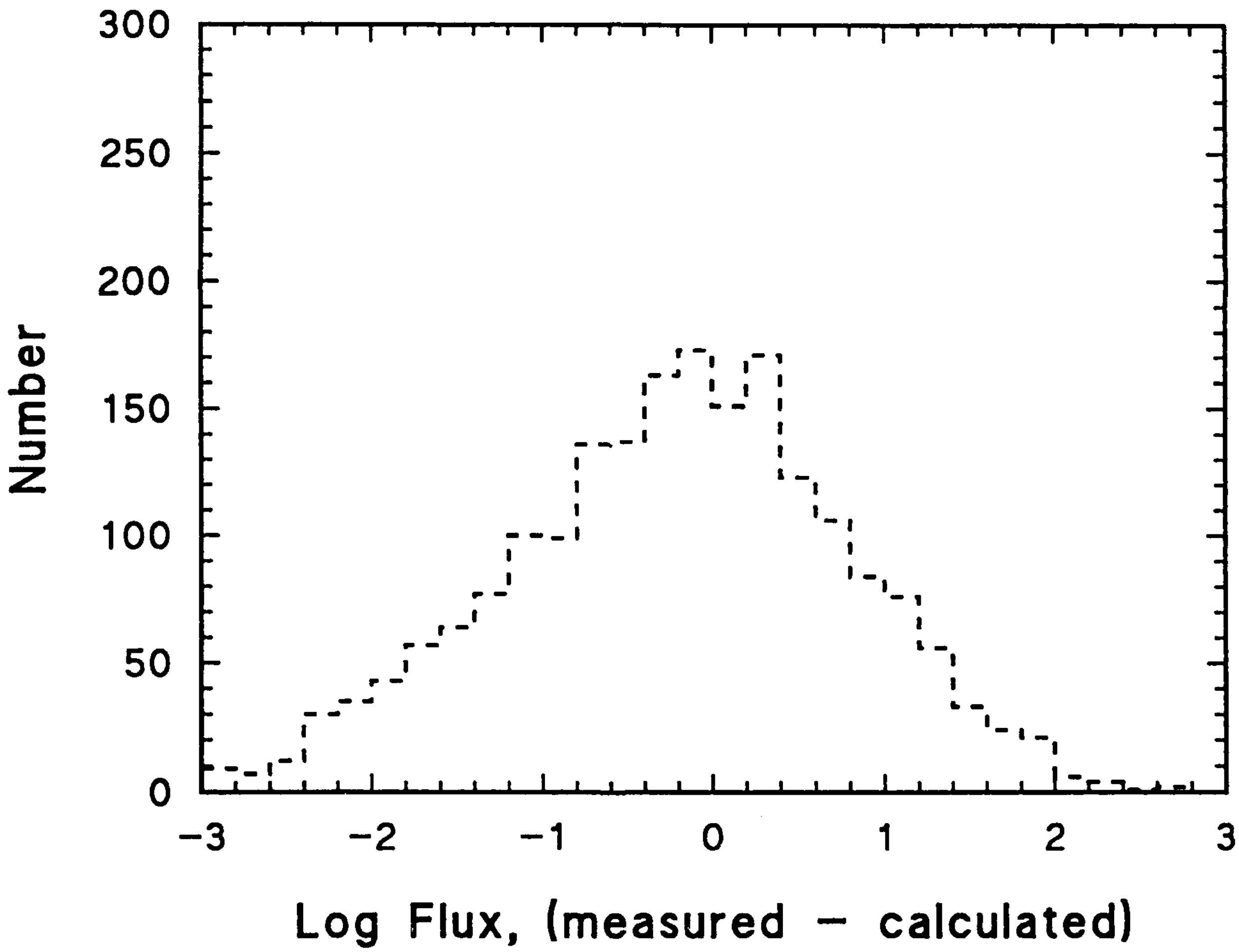

Figure 6. Error distribution for the unnormalized model (FORECAST.C) for the entire data set from April 19, 1982 to June 4, 1988.

NORMALIZED MODEL.

The model described above makes a forecast based solely on a time series of ΣKp without any information about the actual flux levels. Since measurements of the relativistic electron fluxes at geosynchronous orbit are routinely made by the GOES spacecraft, we have developed a second neural network model that includes as an additional input, the measured daily averaged flux of > 3 MeV electrons on Day 0. Note that the GOES fluxes must be scaled to > 3 MeV for this application. This network has 11 input neurons, 7 hidden neurons (one more than the unnormalized network), and one output neuron. The network was trained using the same training set from July and August 1984. This model has also been coded for IBM-compatible personal computers. The source code, FORECST2.C, is provided in Appendix B and a sample output forecasting the flux for 2 July 1984 by the command line:

```
FORECST2 07/02/84 0.08 0.47 21.0 21.0 20.7 27.9 16.4 17.3 19.6
23.9
15.7 12.4 > PRN
```

is shown in Figure 7

```
                The Aerospace Neural Network Model for
        Relativistic Electron Flux at Geosynchronous Orbit

                        Version 2.0 7/11/91

                    by H. C. Koons and D. J. Gorney
              Space and Environment Technology Center
                     The Aerospace Corporation
                       El Segundo, California

Input values:
     {Day 0}: 07/01/84
     Flux for {Day - 1}: 0.080
     Flux for {Day 0}: 0.470
     Sum Kp: 21.0 21.0 20.7 27.9 16.4 17.3 19.6 23.9 15.7
12.4

Log Electron Flux (> 3 MeV) for Day 0 = 0.136922

Predicted Log Electron Flux (> 3 MeV) for {Day + 1}:
Sum Kp =   0 -   8  Log Flux =  1.461  Probability = 0.052
Sum Kp =   8 -  16  Log Flux =  1.061  Probability = 0.319
Sum Kp =  16 -  24  Log Flux =  0.657  Probability = 0.392
Sum Kp =  24 -  32  Log Flux =  0.271  Probability = 0.170
Sum Kp =  32 -  40  Log Flux = -0.078  Probability = 0.050
Sum Kp =  40 -  48  Log Flux = -0.377  Probability = 0.013
Sum Kp =  48 -  56  Log Flux = -0.621  Probability = 0.002
Sum Kp =  56 -  64  Log Flux = -0.813  Probability = 0.001
Sum Kp =  64 -  72  Log Flux = -0.959  Probability = 0.000

           Very High (>2.7) Probability =   0 %
             High (1.7-2.7) Probability =   0 %
     Intermediate (0.7-1.7) Probability =  37 %
                Low (< 0.7) Probability =  63 %
```

Figure 7. Sample output from FORECST2.EXE for 2 July 1984.

The overall performance of the network was also measured by comparing the model outputs with measured fluxes over the entire ~6-year period from April 19, 1982, to June 4, 1988. The distribution of errors is shown in Figure 8. The average log error is -0.16, and the RMS log error is 0.69. This distribution is significantly narrower than the error distribution for the unnormalized model shown in Figure 6. Thus, the normalized model should be used for forecasting when reliably measured fluxes are available.

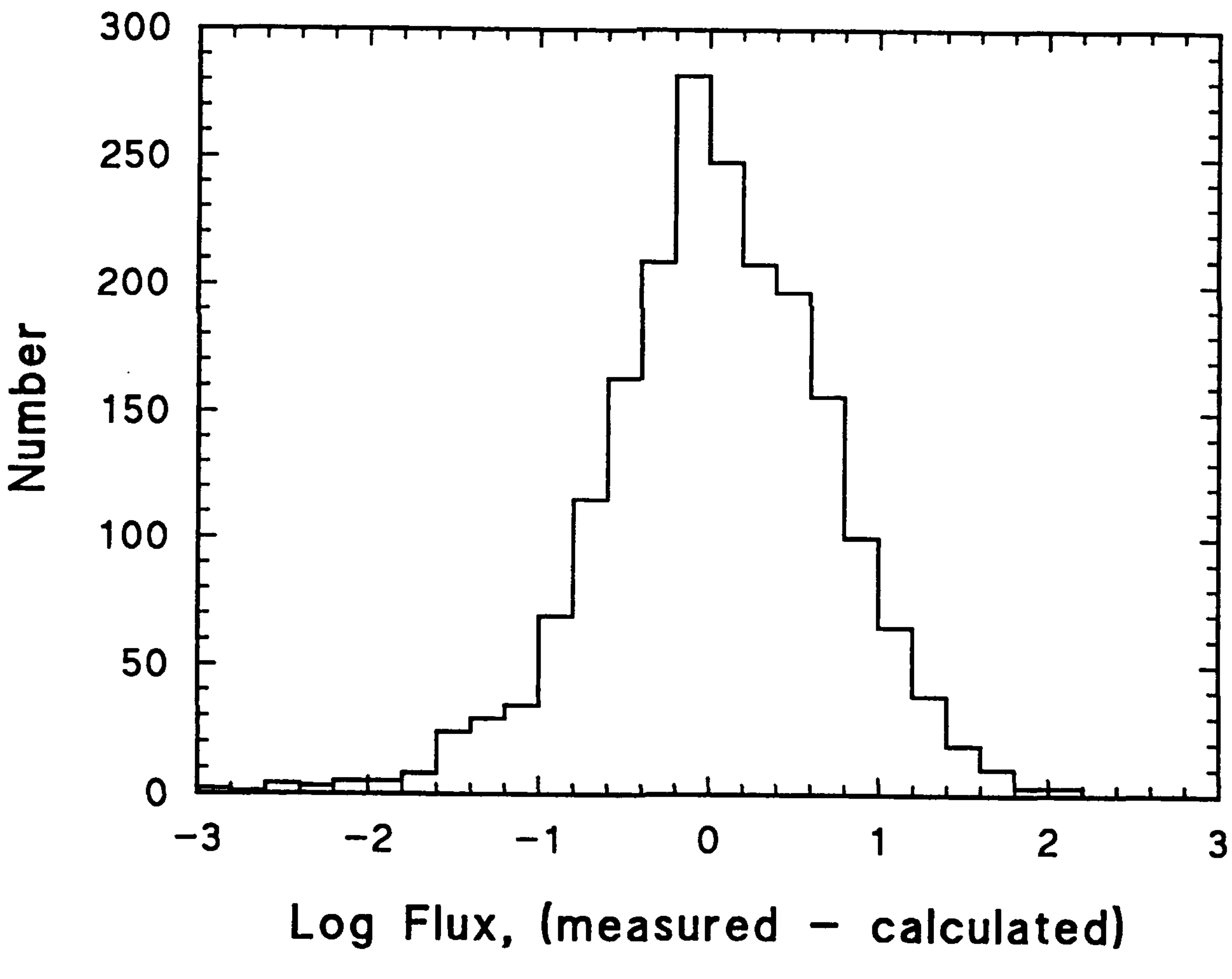

Figure 8. Error distribution for the normalized model (FORECST2.C) for the entire data set from April 19, 1982 to June 4, 1988.

The distribution of errors for the normalized model for the 100 days with the highest measured flux from April 19, 1982 to June 4, 1988 is shown in Figure 9. The distribution is very narrow, indicating that the neural network model is particularly effective in estimating the flux during periods when it is enhanced.

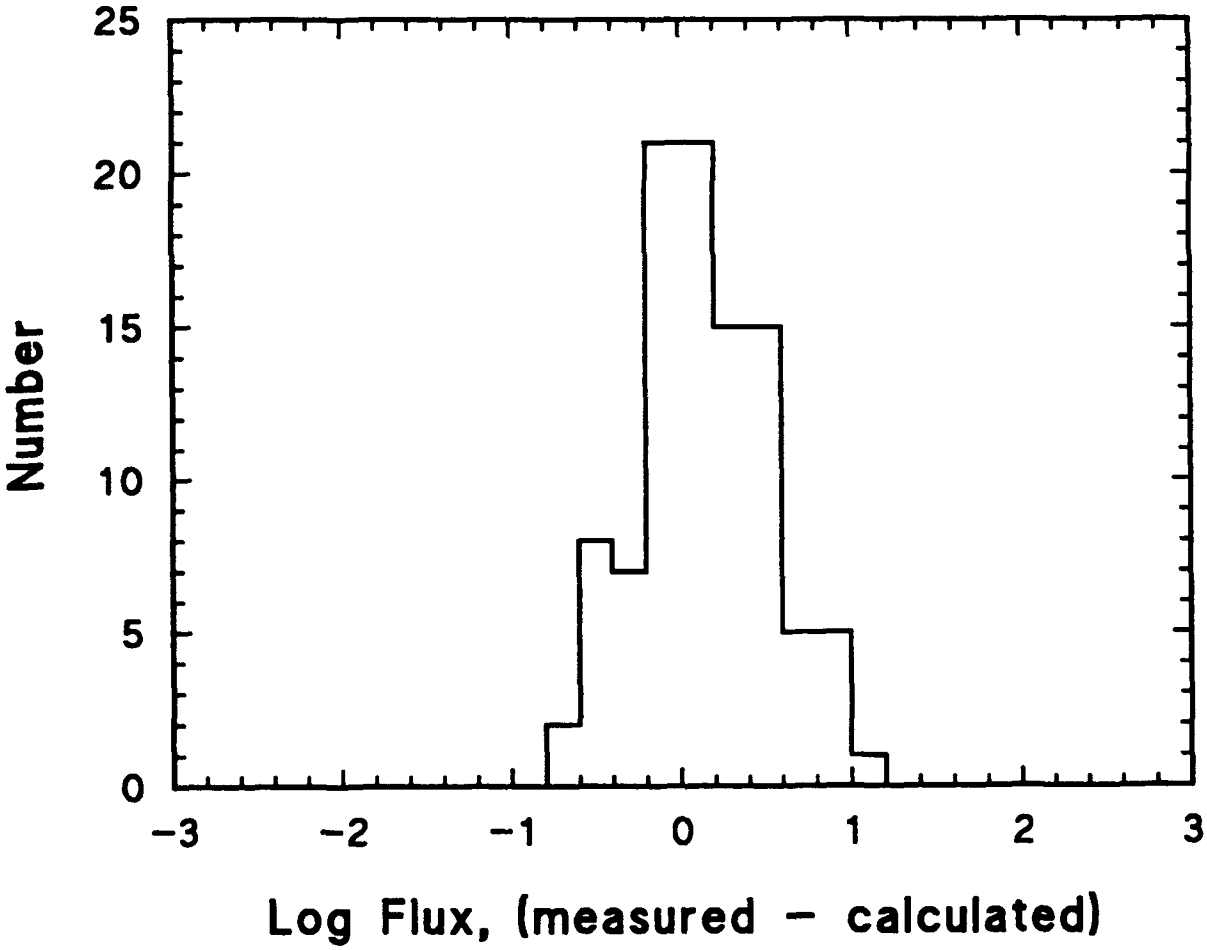

Figure 9. Error distribution for the normalized model (FORECST2.C) for the 100 days with the largest measured flux from April 19, 1982 to June 4, 1988.

The normalized model can also be used with flux measurements having thresholds other than 3 MeV provided a simple arithmetic conversion is applied to the data input. The energy dependence of the electron flux at geosynchronous orbit tends to have a simple exponential dependence given by

$$dJ/dE = (dJ/dE)_0 \exp(-E / E_0) \quad (cm^2/s\text{-}ster\text{-}keV),$$

where E_0, the spectral e-folding energy, is about 0.6 MeV (Baker et. al., 1987). Then the integral flux J_E above some threshold E1 can be written

$$J_E = \int_0^{4\pi} \int_{E_1}^{\infty} (dJ/dE)_0 \exp(-E/E_0) d\Omega dE .$$

The ratio of the flux above energy E_2 to that above energy E_1 is then

$$J_{E2} / J_{E1} = \exp [(E_1 - E_2) / E_0].$$

Taking the logarithm to the base 10

$$\log_{10} J_{E2} = \log_{10} J_{E1} + [(E_1 - E_2) / E_0] \log_{10} e.$$

For $E_2 = 3$ MeV,

$$\log_{10} J_{E2} = \log_{10} J_{E1} + [(E_1 - 3.0) / 0.6] * 0.43429.$$

Example: Suppose a measurement of the integral flux above 2 MeV is available, and the value of the logarithm to the base 10 of the daily average flux is 2.64. What value should be used as the model input? Here E1 = 2.0 MeV, and $\log_{10} J_{E1} = 2.64$. From the last equation above the model input value, $\log_{10} J_{E2}$ is given by

$$\log_{10} J_{E2} = 2.64 + [(2.0 - 3.0) / 0.6] * 0.43429 = 1.916.$$

SUMMARY

We have developed two neural network models that provide practical, day-ahead forecasts of the relativistic electron flux at geosynchronous orbit. The less accurate model uses only the daily sum of the planetary magnetic index, Kp, as its input. The more accurate model is normalized by using the known flux from the preceding day as an additional input.

The models are sufficiently accurate to serve as forecasting tools for times of high relativistic electron flux. This should be especially useful to operators of geosynchronous and other high-altitude spacecraft that may be susceptible to anomalies caused by these electrons.

REFERENCES

Baker, D. N., P. R. Higbie, R. D. Belian and E. W. Hones, "Do Jovian electrons influence the terrestrial outer radiation zone?" *Geophys. Res. Lett.*, 6, 531, 1979.

Baker, D. N., J. B. Blake, R. W. Klebesadel and P. R., Higbie, "Highly relativistic electrons in the Earth's outer magnetosphere, 1. Lifetimes and temporal history 1979-1984," *J. Geophys. Res.*, 91, 4265, 1986.

Baker, D. N., J. B. Blake, D. J. Gorney, P. R. Higbie, R. W. Klebesadel, and J. H. King, "Highly relativistic magnetospheric electrons: A role in coupling to the middle atmosphere?" *Geophys. Res. Lett.*, 14, 1027, 1987.

Beers, B. L., "Radiation-induced signals in cables," *IEEE Trans. Nucl. Sci.*, NS-24, 2429, 1977.

Koons, H. C., and D. J. Gorney, "A neural network model of the relativistic electron flux at geosynchronous orbit," *J. Geophys. Res.*, 96, 5549-5556, 1991.

Meulenberg, A., Jr., "Evidence for a new discharge mechanism for dielectrics in plasmas," *Progress in Astronautics and Aeronautics: Spacecraft Charging by Magnetospheric Plasmas*, Volume 47, edited by A. Rosen, AIAA, New York, pp. 236-247, 1976.

Nagai, T., "Solar variability observed with GMS/SEM," *Pap. Meteorol. Geophys.*, 38, 157, 1987.

Nagai, T., "Space weather forecast: Prediction of relativistic electron intensity at synchronous orbit," *Geophys. Res. Lett.*, 15, 425, 1988.

Reagan, J. B., R. E. Meyerott, E. E. Gaines, R. W. Nightingale, P. C. Filbert and W. L. Imhof, "Space charging currents and their effects on spacecraft systems," *IEEE Trans. Electr. Insul., EI-18*, 354, 1983.

Vampola, A. L., "Thick dielectric charging on high-altitude spacecraft," *Journal of Electrostatics, 20*, 21, 1987.

Wenaas, E. P., "Spacecraft charging effects by the high energy natural environment," *IEEE Trans. Nucl. Sci., NS-24*, 2281-2284, 1977.

APPENDIX A

```
/*      Program: FORECAST.C

     Copyright 1990 by H. C. Koons
     Space Sciences Laboratory
     The Aerospace Corporation
     El Segundo, CA

     Revised: 2/16/90

     The program predicts the flux of ~3 MeV electrons at
     geosynchronous orbit from ten values of the daily sum
     of the planetary magnetic index Kp.

     The algorithm is based on the neural network algorithm
     in BrainMaker, neural network software, from California
     Scientific Software.

     The weight matricies from ENET1.MTX are used.

     Input:  10 values of (Sum Kp) for 10 consecutive days.
          (day zero is the day of the prediction)

     Output: The logarithm of the electron flux for day zero
          for electrons with energies greater than 3 MeV.

          The predicted log of the electron flux for day
          one as a function of (Sum Kp)

          The probability that the flux will exceed certain
          values related to bulk charging.
*/

#include <conio.h>
#include <stdio.h>
#include <stdlib.h>
#include <math.h>

#define CR_TO_FLUX 1.62
#define NUM_IN 10
#define NUM_HID 6
#define NUM_OUT 1
```

```
double w12[][NUM_HID] = {
      { 2.374,  -.639,  1.889,  1.842,  1.216,  4.204},
      {  .868,  -.264,  2.198,  -.723, -1.853, -1.111},
      {  .790, -2.876, -1.457,   .141, -2.302, -3.078},
      {-1.060,   .605,  1.482, -1.812, -2.802,  2.245},
      {-1.061, -1.293,  -.649,  -.689, -1.999, -2.245},
      { -.756,  -.489,  2.684, -1.255, -3.711,  2.609},
      { 4.986,   .369,  1.885, -1.571, -2.256, -1.377},
      {-1.358,  -.916,  1.143, -1.196,  -.759, -3.052},
      {-2.553,  -.588,  -.197, -2.524,  -.155,  -.903},
      { -.028,   .723, -3.071, -2.401, -2.857,  1.131}
      };

double w12th[] = {  .818,  4.236,  -.797,  2.582,  7.999,
-1.890};

double w23[] = {-2.019, 1.929, 2.464, 4.248, -4.000,
-5.139};

double w23th = 0.077;

double prob[10][10] = {
      {.410, .387, .142, .047, .012, .002, .001, .000, .000,
.000},
      {.178, .457, .248, .085, .026, .005, .001, .000, .000,
.000},
      {.052, .319, .392, .170, .050, .013, .002, .001, .000,
.000},
      {.014, .127, .387, .328, .115, .020, .007, .002, .000,
.000},
      {.003, .067, .211, .362, .273, .064, .017, .002, .001,
.000},
      {.005, .057, .158, .262, .324, .147, .033, .014, .000,
.000},
      {.001, .010, .153, .214, .265, .204, .092, .051, .010,
.000},
      {.000, .000, .000, .040, .560, .240, .080, .080, .000,
.000},
      {.000, .000, .000, .000, .667, .000, .333, .000, .000,
.000}
      };

char *msg[4] = {"Low (< 0.7)", "Intermediate (0.7-1.7)",
"High (1.7-2.7)", "Very High (>2.7)"};

double enet(double a[]);
```

```
/*-------------------------------------*/
void main(int argc, char *argv[])
{
     register int bin;
     register int k, l;
     double bin_prob[4];
     double day_zero;
     double flux;
     double input[10];
     double probability;

     if (argc != 11) {
          clrscr();
          fprintf(stdout, "\n*** Input Error ***\n");
          fprintf(stdout, "Enter 10 values of Sum Kp on the
command line with Day 0 first\n");
          fprintf(stdout, "followed by Day -1 etc. For
example, enter 37.7 for 38-\n");
          fprintf(stdout, "38 for 38 or 38.3 for 38+\n");
          exit(0);
     }

     clrscr();
     printf("                     The Aerospace Neural
Network Model for\n");
     printf("               Relativistic Electron Flux at
Geosynchronous Orbit\n\n");
     printf("                         by H. C. Koons and D.
J. Gorney\n");
     printf("                            Space Sciences
Laboratory\n");
     printf("                            The Aerospace
Corporation\n");
     printf("                              El Segundo,
California\n");

     printf("\n   Input values (Sum Kp): ");
     for (k = 0; k < NUM_IN; k++)
          printf("%4.1f ", atof(argv[k + 1]));

     /* scale input into the range from 0.0 to 1.0 */
     for (k = 0; k < NUM_IN; k++)
          input[k] = atof(argv[k+1]) / 50.0;

     /* save day zero value of Sum Kp */
     day_zero = atof(argv[1]);

     flux = enet(input) + CR_TO_FLUX;

     printf("\n\n");
     printf("         Log Electron Flux (> 3 MeV) for Day 0
= %f\n", flux);
```

```
     /* move inputs by one day */
     for (k = NUM_IN - 1; k > 0; k--)
          input[k] = input[k - 1];

     for (bin = 0; bin < 4; bin++)
          bin_prob[bin] = 0;

     printf("\n");
     printf("Predicted Log Electron Flux (> 3 MeV) for Day
+1:\n");
     for (l = 0; l < 9; l++)  {
          input[0] = ((double) l * 80.0 + 40.0 )/ 500.0;

          /* for (k = 0; k < NUM_IN; k++)
               printf("%f ", input[k]);
          */

          flux = enet(input) + CR_TO_FLUX;

          probability = prob[(int) (day_zero / 8.0)][l];

          printf("Sum Kp = %3d - %3d  Log Flux = %6.3f"
              "  Probability = %5.3f", l * 8, (l + 1) * 8,
flux,
              probability);

          if (flux >= 2.7)
               bin_prob[3] += probability;
          else if (flux >= 1.7 && flux < 2.7)
               bin_prob[2] += probability;
          else if (flux >= 0.7 && flux < 1.7)
               bin_prob[1] += probability;
          else
               bin_prob[0] += probability;

          if (flux >= 2.7)
               printf("  *** Warning ***\n");
          else
               printf("\n");


     }
          printf("\n");
          for (bin = 3; bin > -1; bin--)
               printf("%22s Probability = %3d %\n",
msg[bin], (int) (bin_prob[bin] * 100.0 + 0.5));
}
```

```
/*---------------------------------------*/
double enet(double input[])
{
     register int i, j;
     double activation2[NUM_HID];
     double activation3;
     double output2[NUM_HID];
     double output3;

     /* calculate output from hidden neurons */
     for (j = 0; j < NUM_HID; j++)   {
          activation2[j] = 0.0;
          for(i = 0; i < NUM_IN; i++)   {
               activation2[j] += input[i] * w12[i][j];
          }


          activation2[j] += w12th[j];
          output2[j] = 1.0 / (1.0 + exp(-activation2[j]));
     }

     /* calculate output from output neuron */
     activation3 = 0.0;
     for (j = 0.0; j < NUM_HID; j++)
          activation3 += output2[j] * w23[j];

     activation3 += w23th;
     output3 = 1.0 / (1.0 + exp(-activation3));

     return 5.0 * output3 - 3.0;
}
```

APPENDIX B

```
/*      Program: FORECST2.C

     Copyright 1991 by H. C. Koons
     Space and Environment Technology Center
     The Aerospace Corporation
     El Segundo, CA

     1.00
     1.01 Revised: 2/16/90
     2.00 Derived from FORECAST.C

     The program predicts the flux of ~3 MeV electrons at
     geosynchronous orbit from ten values of the daily sum
     of the planetary magnetic index Kp.

     The algorithm is based on the neural network algorithm
     in BrainMaker, neural network software, from California
     Scientific Software.

     The weight matricies from ENET11A.MTX are used.

     Input:  The date for Day 0.

          The logarithm of the electron flux for Day - 1
          and Day 0 for electrons with energies greater than
3 MeV.

          10 values of (Sum Kp) for 10 consecutive days.
          (Day 0 is the day of the prediction)

     Output: The logarithm of the electron flux for day zero
          for electrons with energies greater than 3 MeV.

          The predicted log of the electron flux for day
          one as a function of (Sum Kp)

          The probability that the flux will exceed certain
          values related to bulk charging.
*/

#include <conio.h>
#include <stdio.h>
#include <stdlib.h>
#include <math.h>

#define CR_TO_FLUX 1.62
#define NUM_IN 11
#define NUM_HID 7
#define NUM_OUT 1
#define NUM_KP 10

double w12[][NUM_HID] = {
```

```
     { 2.991, -1.361,  1.698,  1.355, -2.283, -1.441,
-2.960},
     {-1.106, -0.233, -0.292,  0.405,  0.439, -0.418,
-0.564},
     { 1.775,  1.973, -1.366, -3.321,  3.331,  2.332,
0.383},
     {-2.269, -1.343,  4.283,  0.303, -0.469, -0.874,
-1.649},
     { 1.365, -0.458,  0.536,  0.833, -0.823, -1.896,
-0.955},
     {-0.650,  0.827,  1.806, -0.811, -0.095, -1.604,
-1.626},
     { 2.831, -1.391,  0.178, -0.574,  0.827, -2.436,
-2.112},
     { 0.999,  1.957, -2.351, -3.139,  0.705,  3.033,
-1.552},
     {-4.199,  1.196,  0.976, -0.670, -0.026,  0.338,
-0.022},
     {-0.429,  1.563, -0.594,  2.302, -2.182,  1.466,
-1.000},
     {-0.328,  1.718, -0.600, -1.065, -0.657, -3.181,
3.874}
     };

double w12th[] = { 0.849, -0.841, -1.792,  2.996,  1.404,
0.385,  4.998};

double w23[] = {3.479,  0.319,  4.526, -3.983, -3.765,
3.867,  0.549};

double w23th = -1.070;

double prob[10][10] = {
     {.410, .387, .142, .047, .012, .002, .001, .000, .000,
.000},
     {.178, .457, .248, .085, .026, .005, .001, .000, .000,
.000},
     {.052, .319, .392, .170, .050, .013, .002, .001, .000,
.000},
     {.014, .127, .387, .328, .115, .020, .007, .002, .000,
.000},
     {.003, .067, .211, .362, .273, .064, .017, .002, .001,
.000},
     {.005, .057, .158, .262, .324, .147, .033, .014, .000,
.000},
     {.001, .010, .153, .214, .265, .204, .092, .051, .010,
.000},
     {.000, .000, .000, .040, .560, .240, .080, .080, .000,
.000},
     {.000, .000, .000, .000, .667, .000, .333, .000, .000,
.000}
     };
```

```
char *msg[4] = {"Low (< 0.7)", "Intermediate (0.7-1.7)",
"High (1.7-2.7)", "Very High (>2.7)"};

double enet(double a[]);

/*--------------------------------------*/
void main(int argc, char *argv[])
{
     register int bin;
     register int k, l;
     double bin_prob[4];
     double day_zero;
     double flux;
     double input[NUM_IN];
     double probability;

     if (argc != 14) {
          clrscr();
          fprintf(stdout, "\n*** Input Error ***\n");
          fprintf(stdout, "Enter the date for {Day 0} as
mm/dd/yy and\n");
          fprintf(stdout, "enter 1 value for Log Flux for
{Day - 1} and\n");
          fprintf(stdout, "enter 1 value for Log Flux for
{Day 0} and\n");
          fprintf(stdout, "enter 10 values of Sum Kp on the
command line with {Day 0} first\n");
          fprintf(stdout, "followed by Day -1 etc. For
example, enter 37.7 for 38-\n");
          fprintf(stdout, "38 for 38 or 38.3 for 38+\n");
          exit(0);
     }

     clrscr();
     printf("                     The Aerospace Neural
Network Model for\n");
     printf("               Relativistic Electron Flux at
Geosynchronous Orbit\n\n");
     printf("                               Version 2.0
7/11/91\n\n");
     printf("                        by H. C. Koons and D.
J. Gorney\n");
     printf("                Space and Environment
Technology Center\n");
     printf("                              The Aerospace
Corporation\n");
     printf("                                El Segundo,
California\n\n");

     printf("Input values:\n");
     printf("  {Day 0}: %s\n", argv[1]);
     printf("  Flux for {Day - 1}: %5.3f\n", atof(argv[2]));
     printf("  Flux for {Day 0}: %5.3f\n", atof(argv[3]));
     printf("  Sum Kp: ");
```

```
    for (k = 0; k < NUM_KP; k++)
         printf("%4.1f ", atof(argv[k + 4]));

    /* scale count rate for day -1 */
    input[0] = ((atof(argv[2]) - CR_TO_FLUX) - ( -3.0)) /
5.0;

    /* scale input into the range from 0.0 to 1.0 */
    for (k = 0; k < NUM_KP; k++)
         input[k + 1] = atof(argv[k+4]) / 50.0;

    /* save day zero value of Sum Kp */
    day_zero = atof(argv[4]);

    flux = enet(input) + CR_TO_FLUX;

    printf("\n\n");
    printf("            Log Electron Flux (> 3 MeV) for Day 0
= %f\n", flux);

    /* move inputs by one day */
    for (k = NUM_KP; k > 0; k--)
         input[k+1] = input[k];

    for (bin = 0; bin < 4; bin++)
         bin_prob[bin] = 0;

    printf("\n");
    printf("Predicted Log Electron Flux (> 3 MeV) for {Day
+ 1}:\n");
    input[0] = ((atof(argv[3]) - CR_TO_FLUX) - (-3.0)) /
5.0;
    for (l = 4; l <= 68; l += 8)  {
         input[l] = (double) l / 50.0;

         /* for (k = 0; k < NUM_IN; k++)
          *      printf("%5.2f ", input[k]);
          */

         flux = enet(input) + CR_TO_FLUX;

         probability = prob[(int) (day_zero / 8.0)][(int)
(l / 8)];

         printf("Sum Kp = %3d - %3d  Log Flux = %6.3f"
              "  Probability = %5.3f", l - 4, l + 4, flux,
             probability);
```

```
        if (flux >= 2.7)
            bin_prob[3] += probability;
        else if (flux >= 1.7 && flux < 2.7)
            bin_prob[2] += probability;
        else if (flux >= 0.7 && flux < 1.7)
            bin_prob[1] += probability;
        else
            bin_prob[0] += probability;

        if (flux >= 2.7)
            printf("  *** Warning ***\n");
        else
            printf("\n");


    }
        printf("\n");
        for (bin = 3; bin > -1; bin--)
            printf("     %22s Probability = %3d %\n",
msg[bin], (int) (bin_prob[bin] * 100.0 + 0.5));

        printf("\f");
}

/*---------------------------------------*/
double enet(double input[])
{
    register int i, j;
    double activation2[NUM_HID];
    double activation3;
    double output2[NUM_HID];
    double output3;

    /* calculate output from hidden neurons */
    for (j = 0; j < NUM_HID; j++)  {
        activation2[j] = 0.0;
        for(i = 0; i < NUM_IN; i++)  {
            activation2[j] += input[i] * w12[i][j];
        }


        activation2[j] += w12th[j];
        output2[j] = 1.0 / (1.0 + exp(-activation2[j]));
    }

    /* calculate output from output neuron */
    activation3 = 0.0;
    for (j = 0.0; j < NUM_HID; j++)
        activation3 += output2[j] * w23[j];

    activation3 += w23th;
    output3 = 1.0 / (1.0 + exp(-activation3));
    return 5.0 * output3 - 3.0;
}
```

TECHNOLOGY OPERATIONS

The Aerospace Corporation functions as an "architect-engineer" for national security programs, specializing in advanced military space systems. The Corporation's Technology Operations supports the effective and timely development and operation of national security systems through scientific research and the application of advanced technology. Vital to the success of the Corporation is the technical staff's wide-ranging expertise and its ability to stay abreast of new technological developments and program support issues associated with rapidly evolving space systems. Contributing capabilities are provided by these individual Technology Centers:

Electronics Technology Center: Microelectronics, solid-state device physics, VLSI reliability, compound semiconductors, radiation hardening, data storage technologies, infrared detector devices and testing; electro-optics, quantum electronics, solid-state lasers, optical propagation and communications; cw and pulsed chemical laser development, optical resonators, beam control, atmospheric propagation, and laser effects and countermeasures; atomic frequency standards, applied laser spectroscopy, laser chemistry, laser optoelectronics, phase conjugation and coherent imaging, solar cell physics, battery electrochemistry, battery testing and evaluation.

Mechanics and Materials Technology Center: Evaluation and characterization of new materials: metals, alloys, ceramics, polymers and their composites, and new forms of carbon; development and analysis of thin films and deposition techniques; nondestructive evaluation, component failure analysis and reliability; fracture mechanics and stress corrosion; development and evaluation of hardened components; analysis and evaluation of materials at cryogenic and elevated temperatures; launch vehicle and reentry fluid mechanics, heat transfer and flight dynamics; chemical and electric propulsion; spacecraft structural mechanics, spacecraft survivability and vulnerability assessment; contamination, thermal and structural control; high temperature thermomechanics, gas kinetics and radiation; lubrication and surface phenomena.

Space and Environment Technology Center: Magnetospheric, auroral and cosmic ray physics, wave-particle interactions, magnetospheric plasma waves; atmospheric and ionospheric physics, density and composition of the upper atmosphere, remote sensing using atmospheric radiation; solar physics, infrared astronomy, infrared signature analysis; effects of solar activity, magnetic storms and nuclear explosions on the earth's atmosphere, ionosphere and magnetosphere; effects of electromagnetic and particulate radiations on space systems; space instrumentation; propellant chemistry, chemical dynamics, environmental chemistry, trace detection; atmospheric chemical reactions, atmospheric optics, light scattering, state-specific chemical reactions and radiative signatures of missile plumes, and sensor out-of-field-of-view rejection.