United States Department of Commerce
National Institute of Standards and Technology

NIST Technical Note 1275

A Study of Meteorological Processes Important in the Degradation of Materials Through Surface Temperature

Sam C. Saunders, Mark A. Jensen, and Jonathan W. Martin

NIST Technical Note 1275

A Study of Meteorological Processes Important in the Degradation of Materials Through Surface Temperature

Sam C. Saunders
Mark A. Jensen

Scientific Consulting Services Inc.
P. O. Box 2118 C.S.
Pullman, WA 99165-2118

and

Jonathan W. Martin

Center for Building Technology
National Institute of Standards and Technology
Gaithersburg, MD 20899

April 1990

DEPARTMENT OF COMMERCE · UNITED STATES OF AMERICA

U.S. Department of Commerce
Robert A. Mosbacher, Secretary

National Institute of Standards and Technology
John W. Lyons, Director

National Institute of Standards and Technology
Technical Note 1275
Natl. Inst. Stand. Technol.
Tech. Note 1275
45 pages (Apr. 1990)
CODEN: NTNOEF

U.S. Government Printing Office
Washington: 1990

For sale by the Superintendent of Documents
U.S. Government Printing Office
Washington, DC 20402

Abstract

One of the greatest impediments in forecasting the service life of a material exposed outdoors is the uncontrolled and non-predictable nature of the ambient factors comprising its environment. This contributes to the difficulty of establishing the cause-and-effect relationship between these factors and the rate of material degradation. To surmount these difficulties it is necessary to characterize quantitatively each of the factors comprising the exposure environment which are thought to be important in the material's degradation. The selection and quantification of these factors must accommodate not only the periodicity of the diurnal cycle but its statistical fluctuation. The objective of this research is to develop a general mathematical model, through Fourier analysis, characterizing the diurnal variation in the primary factor, material temperature. This factor is felt to be important in the degradation of a wide range of materials and protective systems, including coated steel panels. The steps taken and problems encountered in developing such a model are outlined. It is concluded that the simulated data generated from this model display virtually the same stochastic behavior as does the real data and that, if appropriate meteorological records are available, it will be possible to characterize and reproduce the statistical behavior for any locality, season and panel orientation.

KEY WORDS; diurnal cycle, energy-balance equation, environmental factor, Fourier analysis, meteorological data, panel temperature, time series analysis

Contents

List of Figures

1. Introduction

The ultimate objective of the present analysis is to determine the magnitude of the influence of certain environmental processes despite these processes being obscured by statistical fluctuation. These processes are thought, from plausible physical reasons, to affect significantly the degradation of certain coating materials of interest. To perform the analysis some ancillary problems first must be addressed:

1. A mathematical model sufficiently general to encompass all the environmental processes of interest must be adopted;

2. The stochastic nature of all the environmental processes under investigation must be determined in the light of the mathematical model;

3. The distribution of the stochastic variables, which determine the sample environmental processes, must be estimated from the sample data;

4. By using the mathematical model with its derived procedure for composition, all the sample processes must be reconstructable to a degree of approximation which is adequate for the purpose intended;

5. The same procedure must be extendable to simulate the environmental processes, for extended periods of time, under different meteorological conditions;

6. All these mathematical procedures must then be encoded as machine software which can subsequently perform such analysis easily for different data sets taken under different conditions.

The mathematical model developed herein allows one to represent, through Fourier analysis, the diurnal variation seen in all the environmental processes which were observed. It is required that each simulated process, generated mathematically, exhibit periodicities and statistical fluctuation virtually indistinguishable from the observed time series from which it is modeled. Once the procedures for mathematical generation are in place one can, through repeated application, simulate all the environmental time series to be employed as independent variables. These can, in turn, be used to drive, under the appropriate conditions, any given deterministic steady-state equation (such as eq 1) which models the principal degradation phenomenon in question. Once the method has been implemented and validated then the analysis of meteorological data, obtainable from a given locality, can be used to generate a long term laboratory simulation of the effects of, say, environmental temperature on a particular sample panel of specified material, color and attitude as if it were actually exposed at that locality.

To this end, we bring the methodology of frequency-domain analysis to bear on the various time series which drive the model. Specifically, our analysis follows these lines:

1. Form a smoothed time series by taking a moving hourly average of the observed data;
2. Subdivide the smoothed data into subsets each comprising a sample series for a 1-day period;
3. For each smoothed *daily* series, find its Fast Fourier Transform in order to obtain the Fourier coefficients of the principal sinusoidal components of each day;
4. For each *harmonic*, find means and variances of the Fourier coefficients over the *total* number of days in the original sample.

The Fourier coefficients for the principal harmonics to be used in the *simulation* are now chosen from Gaussian distributions having the same means and variances as does the sample determined in item 4. above. To complete the actual simulation, we perturb the generated smoothed data by adding another sequence of dependent normally distributed random variables which reconstructs the quarter-hour variation removed in the smoothing. Full details and justifications for these assumptions are found in the sequel.

The *energy-balance equation for a horizontal plate* is:

$$AH_t\alpha = 2hA(T_p - T_a) - \epsilon A\sigma(T_s^4 - T_p^4) - \epsilon A\sigma(T_b^4 - T_p^4) \tag{1}$$

where

$$\begin{aligned}
\alpha &= \text{absorptivity} \\
h &= \text{convection coefficient, a function of wind speed} \\
H_t &= \text{total solar radiation} \\
A &= \text{surface area of the panel} \\
\sigma &= \text{Stephan Boltzmann constant: } 5.67 \times 10^{-8} \text{ watt/meter}^2\text{K}^4 \\
T_a &= \text{ambient temperature} \\
T_b &= \text{background temperature} \\
T_p &= \text{panel temperature} \\
T_s &= \text{sky temperature} \\
v &= \text{wind velocity} \\
\epsilon &= \text{panel emissivity.}
\end{aligned}$$

For this study we make the absorptivity α equal the panel emissivity ϵ, and let the total solar radiation H_t be zero at night. As a consequence of the design of the exposure apparatus, background temperature, T_b, was approximated by ambient temperature, T_a. The convection coefficient h is, on average, a linear function of the wind speed. Thus the regression coefficients for heat transfer due to the wind are, from [Watmult]

$$h = 2.8 + 3.0v.$$

and the table of emissivity values is:

ϵ	day	night
white	.7	.6
red	.96	.92
black	.98	.96

2. Theory and Assumptions

A periodic function is often studied using Fourier analysis. The form of the Fourier expansion of a function f periodic on $(-\pi, \pi)$ is

$$f(x) = \frac{a_0}{2} + \sum_{i=1}^{\infty}[a_i cos(ix) + b_i sin(ix)] \quad \text{for} \quad -\pi < x < \pi,$$

where

$$a_i = \frac{1}{\pi}\int_{-\pi}^{\pi} f(t)cos(it)\,dt \quad b_i = \frac{1}{\pi}\int_{-\pi}^{\pi} f(t)sin(it)\,dt.$$

We assume that the time series data for one of the driving variables, say

$$x_t \quad \text{for } t = 1, \ldots, nd,$$

is a realization of the stochastic process $X(t)$ for n days, with each day having d equally spaced observations. We further assume that there is an expected diurnal periodicity, so that in distribution

$$X(t) \sim X(t+d) \quad \text{for all } t = 1, \ldots, nd. \tag{2}$$

This is combined with unit-time stochastic influences which can be reduced by averaging each observation over the k forward and backward observations. Thus we have for all t

$$X(t) = Y(t) + e(t) \quad \text{where } Y(t) = \frac{1}{2k+1}\sum_{i=-k}^{k} X(t+i)$$

where $\{e(t) : t = 1, \ldots, nd\}$ is a sequence of (random) error variates with zero mean and $Y(t)$ is the smoothed process. In view of eq (2) it is reasonable to separate the n processes over as many days by altering the notation; we now write

$$Y(jd+t) = Y_j(t) \quad \text{for } j = 1. \ldots, n; \;\; t = 1, \ldots, d. \tag{3}$$

Thus we have n replications of the same daily (almost periodic) stochastic process; the jth one of which can be written in the stochastic form for $j = 1. \ldots, n$

$$Y_j(t) = A_{0,j} + \sum_{i=1}^{\infty}[A'_{i,j}cos(\frac{it\pi}{d}) + B'_{i,j}sin(\frac{it\pi}{d})] \quad \text{for } t = 1, \ldots, d.$$

This representation can be further simplified if we standardize it by eliminating from consideration the variation in the daily high, low and mean value of the process. Set

$$Z_j(t) = \frac{Y_j(t) - A_{0,j}}{M_j} \quad \text{for } j = 1, \ldots, n; \ t = 1, \ldots, d,$$

where $A_{0,j}$ is the mean and M_j is the half-range, i.e., half the maximum variation for the jth day. Thus, without loss of generality, we can write for the smallest value of m for which we obtain an approximation sufficient for our intended purposes

$$Z_j(t) \doteq \sum_{i=1}^{m} [A_{i,j} cos(\frac{it\pi}{d}) + B_{i,j} sin(\frac{it\pi}{d})] \quad \text{for } t = 1, \ldots, d, \tag{4}$$

where the $A_{i,j} = A'_{i,j}/M_j$ and $B_{i,j} = B'_{i,j}/M_j$ for $i = 1, \ldots, m$ are sets of random variables appropriate to the jth day environmental process.

Since the behavior of the process stays the same for each of the n days,[1] except for random influences, it follows that the vectors of the amplitudes of the ith harmonics $(A_{i,1}, \ldots, A_{i,n})$ and $(B_{i,1}, \ldots, B_{i,n})$ for $i = 1, \ldots, m$ each form a sequence of vectors of random variables. The n elements of each vector corresponding to a given harmonic i are independent and identically distributed (iid). Thus, it follows that the n stochastic processes, viz., $\{Z_1(t), \ldots, Z_n(t)\}$ for $t = 1, \ldots, d$, all have the same distribution. Hence the amplitude of each harmonic in each day's Fourier expansion has the same distribution, namely, for each $i = 1, \ldots, m$

$$A_{i,j} \sim F_i \quad \text{and} \quad B_{i,j} \sim G_i \quad \text{for all } j = 1, \ldots, n.$$

Thus it is now our problem to obtain the important harmonics in the Fourier expansion of $Z(t)$, where each day provides an iid replication. After finding the distribution of the random amplitudes from day to day in each of the driving environmental influences, for which we have data, one can integrate all these generated representations of environmental influences into a grand simulation of the panel temperature using the energy balance equation.

Once this integrated procedure is validated then it may be possible to find transformations which could reduce several time series, generated under widely varying parameters of emissivity or temperature extremes, to one series to which all are stochastically equivalent. This should be supported by tests-of-hypothesis about changes in the distributions of the magnitudes of the principal harmonics, as calculated using this mathematical model. The value in considering such transformations is that once verified it would be sufficient to use data sets generated in the laboratory using only white or black panels oriented in the flat position to make valid inferences about the degradation of, say, paints having different absorptivity over a wide range of material parameters.

3. Procedures

We elaborate here on our methods of data analysis. The following is a brief review of the nature of the raw data, a detailed outline of our data preparation and the development of simulation procedures, and a short discussion of some aspects of the simulation program.

[1] This assumption means that n cannot be large enough to entail seasonal changes

3.1 Data Attributes

The data with which we were provided were the concurrently observed time series for 10 variates, 6 of which describe temperature variation of the sample panels, the remaining 4 describing certain other measured environmental variables, namely, ambient temperature (T_a), sky temperature (T_s), incident radiation (H_t), and wind speed (v). We shall refer to these four factors collectively as the *driving variables.*

The data were acquired over approximately 50 days, with the observation vector being updated every 15 minutes. The attitude of the sample panels was modified every 10 days, so that during the 50-day period the panels had been tilted at 45°, and faced in each of the four cardinal directions as well as having lain flat. The partition of the data into files reflects this procedure.

A partial listing of the data appears in the accompanying table.

day/hr/min			T_a	T_s	$T_{p_{blk}}$	$T_{p_{red}}$	$T_{p_{w1}}$	$T_{p_{w2}}$	$T_{p_{w3}}$	$T_{p_{w4}}$	H_t	v
10	13	44	285.8	273.4	297.5	294.4	294.4	292.8	294.9	287.0	567.2	0.3
10	13	59	285.3	272.5	296.4	293.5	293.7	292.1	294.0	286.8	444.8	0.4
10	14	14	284.4	273.1	294.7	292.0	292.3	290.9	292.4	286.1	479.1	0.5
10	14	29	283.9	271.2	291.9	289.7	290.0	288.8	290.0	285.0	349.5	0.4
10	14	44	284.4	272.5	293.2	290.6	290.7	289.5	290.9	285.3	507.6	0.3
10	14	59	284.1	270.8	294.7	291.5	291.8	290.4	292.0	285.8	465.5	0.5
10	15	14	284.3	269.9	293.0	290.1	290.4	289.4	290.7	285.2	336.6	0.2
10	15	29	283.4	268.5	289.6	287.6	287.9	287.1	288.0	283.8	161.6	0.1
10	15	44	282.9	269.5	286.7	285.3	285.5	285.1	285.6	282.6	126.2	0.1
10	15	59	282.2	269.2	284.3	283.4	283.6	283.3	283.7	281.7	80.9	0.2
10	16	14	281.8	270.0	282.7	282.1	282.2	282.1	282.3	281.0	46.1	0.0

3.2 Statistical Protocol

The following protocol performs two functions:

1. It prepares raw data by smoothing and standardizing, after which the smoothed data are analyzed on a day-to-day basis using Fourier analysis. Statistical analysis is performed on the resulting sets of corresponding coefficients to estimate the means and variances of the underlying distributions for use in later simulation.

2. It analyzes the difference series formed by subtracting the smoothed data from the raw data, thus obtaining the *smoothing error* from the quarter-hour fluctuations for which the mean, variance, and autocorrelation are calculated.

The second function (2) allows us to tailor appropriately the random perturbations which were applied to the simulated time series of 10 days, (n = 10), each day containing 96, (d = 96), quarter-hour measurements which were smoothed appropriately.

In general, of course, the total sample period may be any number of days, say n, and each day may be arbitrarily divided into d different observations. Hence we are always provided with nd data points. Smoothing is over k records forward and backward, here $k = 2$ was arbitrarily chosen to give us an hourly mean, but any moderate value of k can be chosen for the smoothed process. Thus the protocol for handling any driving variables is as follows:

Let the given observed time series for a driving variable be denoted by

$$x_t, \quad \text{for } t = 1, \ldots, nd.$$

Then:

1. Smooth over k records back and k records forward at each time point, obtaining

$$y_t = \frac{1}{2k+1} \sum_{i=-k}^{k} x_{t+i} \quad \text{for} \quad t = 1, \ldots, nd.$$

   We make the data cyclic by taking all the indices modulo nd.

2. Calculate the single-record smoothing error fluctuation of actual minus smoothed data, call it

$$\epsilon_t = x_t - y_t, \quad \text{for } t = 1, \ldots, nd,$$

   and compute the mean, variance, and autocorrelation of the error process

$$\{\epsilon_t : t = 1, \ldots, nd\}.$$

3. Standardize the smoothed data for each driving variable by computing for each daily period the maximum, minimum, mean, and half-range of the smoothed series namely, for $j = 1, \ldots, n$

$$\begin{aligned} h_j &= \max_{t=1,\ldots,d} \{y_{d(j-1)+t}\} \\ l_j &= \min_{t=1,\ldots,d} \{y_{d(j-1)+t}\} \\ \mu_j &= \frac{1}{d} \sum_{t=1}^{d} y_{d(j-1)+t} \\ v_j &= \frac{h_j - l_j}{2}. \end{aligned}$$

   Then form n standardized day processes

$$z_{j,t} = \frac{y_{d(j-1)+t} - \mu_j}{v_j} \quad \text{for } j = 1, \ldots, n; \quad t = 1, \ldots, d.$$

4. Perform a Fourier analysis on the jth day's stationary process $\{z_{j,t} : t = 1, \ldots, d\}$ for $j = 1, \ldots, n$ for each of the four driving variables. Thus is obtained a sample of size n from the statistical distribution of the Fourier coefficients for the first several, say four, harmonics present with greatest intensity in each driving process.

By randomly choosing the coefficients of a Fourier expansion from a Gaussian distribution having the same mean and variance as each of the sample distributions of the principal harmonics of a particular smoothed process, we can generate a *simulated* standardized environmental factor process, with the help of the computer. This can be transformed by using the meteorological data of a particular location to scale the simulated process, thus obtaining a site-specific simulation possessing the statistical properties of the actual observed data. Say the local temperature extremes over the jth daily period are (l_j, h_j) with a mid-range of $m_j = (h_j + l_j)/2$, then with n mathematically generated time series of $\{z'_{j,t} : t = 1, \ldots, d\}$ for $j = 1, \ldots, n$ we transform them to obtain a simulated day process

$$y'_{d(j-1)+t} = m_j + \nu_j z'_{j,t} \quad \text{for } t = 1, \ldots, d \text{ and } j = 1, \ldots, n$$

We note that the above protocol does not take into account the long term effect on the distributions of the Fourier coefficients, if any, of the seasonal meteorological variation. Given the appropriate data it could be easily modified to do so, of course.

In order to be realistic, it is necessary to perturb the machine-generated smoothed hourly-averaged time series by adding a set of quarter-hour fluctuations by which to mimic, in all significant detail, the random variation that is actually encountered. We wish to generate for each day a simulated set of fluctuations, the autocorrelations of which closely approximate the autocorrelations of actual observed variation. To do this, we first generate a column d-vector of independent, normally distributed random variates each with variance unity, label it $\mathbf{w}$. We write the transpose

$$\mathbf{w}' = (w_1, \ldots, w_d).$$

We then scale the series by the actual standard deviation, call it σ_j, of the observed fluctuations for the jth day. Finally, we induce the desired dependence between successive fluctuations by computing the $d \times d$ autocovariance matrix of the daily-average observed fluctuations, call it Σ. Since it must be positive definite we can find the square-root matrix $\mathbf{C}$ such that $\Sigma = \mathbf{C}'\mathbf{C}$. Then pre-multiplying the scaled pseudo-random series by the matrix $\mathbf{C}$ we obtain the simulated, dependent smoothing error for the jth day as the column d-vector within the new $n \times d$ matrix of fluctuation data,

$$\mathcal{E} = (\mathbf{e}_1, \ldots, \mathbf{e}_n) = (\sigma_1, \ldots, \sigma_n)\mathbf{C}\mathbf{w}. \tag{5}$$

N.B. The observed daily-average fluctuation, label it $\bar{\mathbf{e}}$, is obtained from this matrix of fluctuation data by averaging across the rows. It is the time series represented by the column d-vector of smoothed errors, the index of which is the time level within the daily period of observations, averaged over the n days of observation:

$$\bar{\mathbf{e}} = \frac{1}{n} \sum_{j=1}^{n} \mathbf{e}_j, \tag{6}$$

where

$$\bar{\mathbf{e}}_i = \frac{1}{n} \sum_{j=1}^{n} e_{i,j} \quad \text{for } i = 1, \ldots, d. \tag{7}$$

We compute the autocorrelation matrix of this average in order that we may use a single matrix for the simulated error for each day, rather than computing separate matrices from the observed data for every day. This provides satisfactory results while reducing computation time considerably.

3.3 Data Compression

The nature of the incident radiation data necessitated a slight digression from the above protocol. Measuring as it does sunlight (in essence), the radiation time series must go to zero during the night, approximately one-half of each daily period. To employ mechanically the protocol on the raw data when we wish to model only the variate's *daytime* behavior would clearly lead to anomalous results, e.g., unrealistically low daily means. To overcome this difficulty, the program (SIMUL) makes use of a data compression subroutine (COMPRESS) which essentially removes all nil observations from the radiation time series and returns a shortened series upon which the protocol operates. Sending the *reduced* time series accompanied by the number of *virtual* observations (i.e., the number of daily observations d less the average daily number of observations removed by compression) through the protocol provides us with the simulated process we desire.

This simulated series must be subsequently "decompressed"; i.e., using the original series as a template, we introduce nil entries into the simulated series. The resulting decompressed series will have zero entries at locations corresponding to zero entries in the original data, and locations corresponding to non-zero entries in the original series will contain the simulated observations. The subroutine DECOMPRESS performs this task.

3.4 Program SIMUL

Program SIMUL (see appendix), written to perform the analysis according to the above protocol, has been made available in a machine-readable version on a DSDD floppy disk under MS-DOS.[2]

Although the above protocol provides specific values, the number of days in the total sample and number of records over which the series is smoothed can be easily modified via the relevant PARAMETER statements.

SIMUL makes extensive use of the IMSL STAT/LIBRARY. Graphical results were obtained using the PGPLOT device-independent library.

SIMUL was written in VAX FORTRAN and run on a Digital Equipment Corporation VAX 11/750.

[2] Interested researchers can contact Dr. Martin for further information.

4. Results

We now retrace the general steps and accomplishments of the analysis:
In figure 1 is a graphical presentation of the original data for a period of approximately 10 days, with observations made every quarter hour (yielding 960 observations), of the driving (environmental) variables *ambient temperature, sky temperature, incident radiation* and *wind speed*. These are depicted respectively in graphs A, B, C, and D.

A Fourier analysis is performed on the smoothed hourly data for each variable and the harmonics of its decomposition are obtained. (The mathematical details are presented in sec. 2.) The increasing goodness of approximation of the first few terms of the partial Fourier expansion to the actual time series of the driving variable, sky temperature, is exhibited in figure 2 as a function of the number of harmonics that are used in the approximation. The number of terms, i.e., Fourier coefficients, that is deemed adequate is a matter of engineering judgment but 8 harmonics seems to give a very good fit to all the driving variables. Of course 4 or 6 may be considered adequate for many applications.

In figure 3 is a comparison of the original time series *vs* the same series reconstructed to the degree of approximation chosen and then perturbed. We show how closely the addition of randomly generated, correlated quarter-hour fluctuations to the time series generated by the partial Fourier expansion of the smoothed hourly environmental variable (sky temperature), will approximate the original data itself.

In figure 4 we depict the empirical sample distributions of the first four cosine coefficients for the sky temperature based on the 10-day observations. For comparison the fitted normal distribution, having the same mean and variance as the data, is drawn on the graph of the empirical distribution along with the Kolmogorov-Smirnov 90% confidence limits. It is seen that with this data the hypothesis that the cosine coefficients are normally distributed cannot be rejected at the 90% confidence level.

Figure 5 compares the autocorrelations of simulated and actual smoothing error for sky temperature. There is virtually no detectable difference in behavior.[3]

Figure 6 gives the time series for the original sky-temperature data in kelvin vs. time and then presents, for comparison, another one that is independently simulated by using only the first four Fourier coefficients.

Figures 7-9 show the effect of compressing the actual data of incident radiation by eliminating from consideration the necessity of fitting the periods of null radiation during the night. This simplifies the Fourier expansion since constant functions require a large number of terms to approximate well.

Figure 8 shows a simulated time series, in graph A, of incident radiation generated mathematically in the compressed state and then decompressed by interposing the periods of null radiation appropriately, exhibited in graph B.

[3]It should be observed that the smoothing error is only correlated with its immediate neighbors. It is thought that this dependence may, in the case of panel temperature, be a result of the use of the steady-state equation for energy balance rather than a more complex dynamic equation.The latter would contain some lag in adjustment of temperature to new conditions. Thus this aspect of nature is compensated for in the sequence of dependent errors from smoothing. In other cases the dependent quarter-hour fluctuation may be a fact of nature.

Figure 9 compares the original incident-radiation data and the corresponding, simulated (mathematically generated) series, utilizing four Fourier coefficients.

Figure 10 compares the original panel-temperature data *vs* the steady-state model eq (1) driven by the 4 simulated variates, using 4 principal harmonics for each time series. Part of the disparity in the variation is due to the use of the steady-state equation, rather than a dynamic one, for panel temperature.

5. Conclusions

The conclusions that can be reached as a result of this investigation, are now presented, along with an opinion as to the degree of certitude with which they can be held. These conclusions at this juncture can be only tentative; however, their validity will certainly increase with the amount of data available. Since all the driving variables exhibit a diurnal periodicity, with differing amounts of stochastic variation, their representation by Fourier series with harmonic amplitudes which are random variables is a natural mathematical model.

The method of Fourier analysis of the time series of the driving variables, along the lines that are pursued here, is probably the only feasible technique of analysis, regardless of how successful or unsuccessful it is in this instance. There are several reasons for this:

- It makes possible the generation of suitable diurnal stress cycles, exhibiting virtually the same statistical behavior as the real data, from virtually any locality and season of the year if the appropriate meteorological records are available.
- It makes possible an optimal judgment as to balancing a sufficiently accurate approximation of the weather record of a driving variable, through the inclusion of a large number of terms in its Fourier series, against the additional complexity and difficulty of generating the large number of random Fourier coefficients necessary to simulate a weather record for an extended period.

A tentative conclusion, which would need additional data to verify, is that unless some influence such as duration-of-wetness is necessary as an environmental variable there is no need to simulate more than one panel attitude or color since these effects can be compensated for mathematically.

This same analysis can be performed for other continuous variables which are not in this study, such as humidity, but damage, or the initiation of damage, occurring at discrete times due to the occurrence of rare events cannot be accounted for easily with this method.

Using independent normal variates (and pair-wise uncorrelated normal variates are independent) to supply the quarter-hour variation will not simulate the driving variate of interest with sufficient accuracy. The point is that a quarter hour is too short a time period to effect independence in the error variates; the dependence is likely greater for shorter time periods and less for longer. The consequences of completing the simulated data by the addition of independent normal variates, rather than using the dependent ones, is not clear because it depends upon usage of the results. Such an assessment of adequacy can only be made by other comparisons with actual results. This dependence in the error variates is also influenced by the lag in the dynamic response of the plate temperature to changes in the driving variates. The present model utilizes only the steady-state equation, which does not incorporate the time-dependence through which lag-time effects could be ascertained. This formula excludes, by definition, the transient behavior in which the dependence would be manifested.

Acknowledgment: The authors acknowledge the assistance of Dr. Hunter Fanney of CBT-NIST in providing the steady-state energy-balance eq (1) for an exposed horizontal plate.

Figures

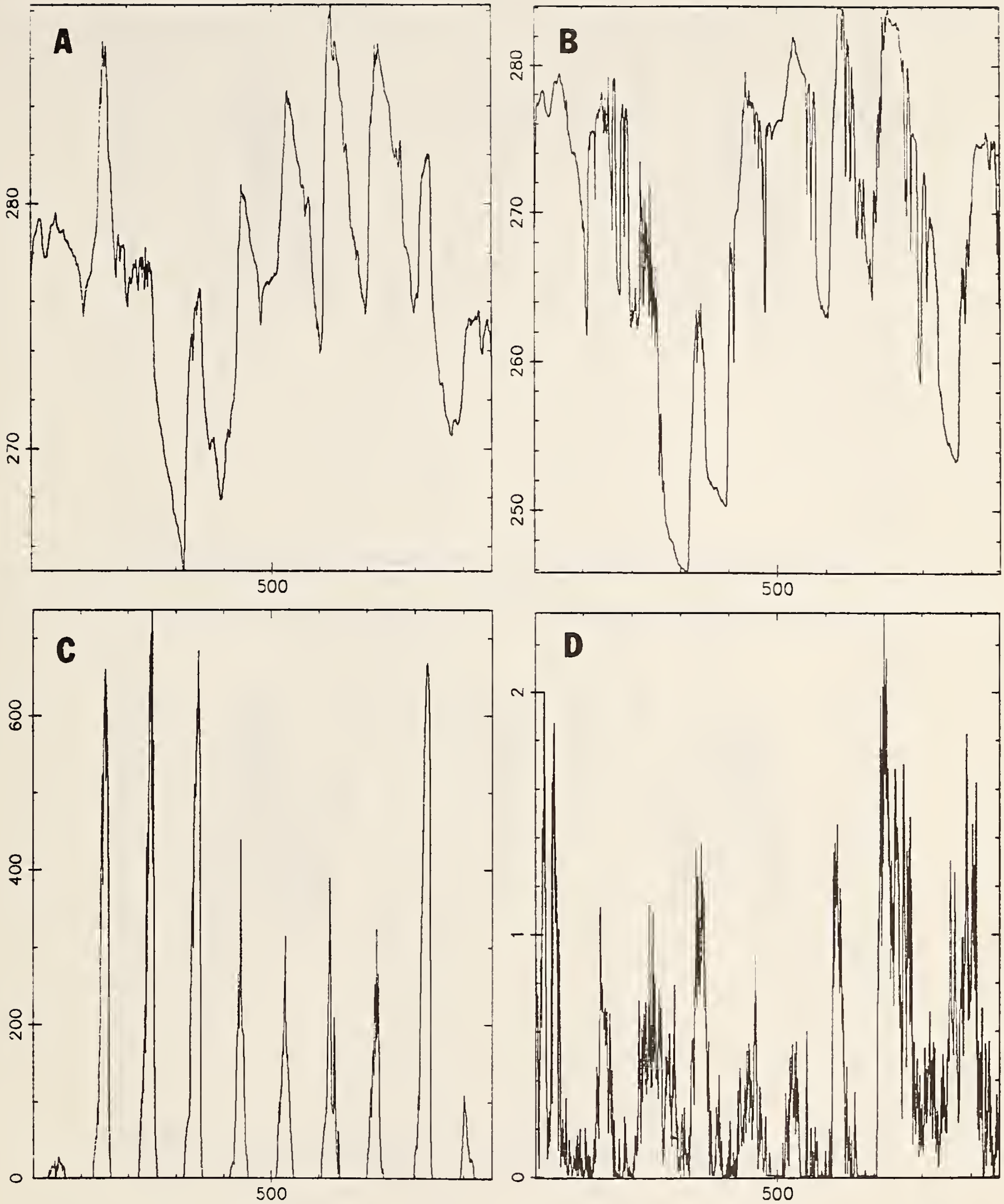

Figure 1 Examples of original data for each driving variable. A. Ambient temperature: time in quarter hours vs. kelvin. B. Sky temperature: time in quarter hours vs. kelvin. C. Incident radiation: time in quarter hours vs. langleys. D. Wind speed: time in quarter hours vs mph.

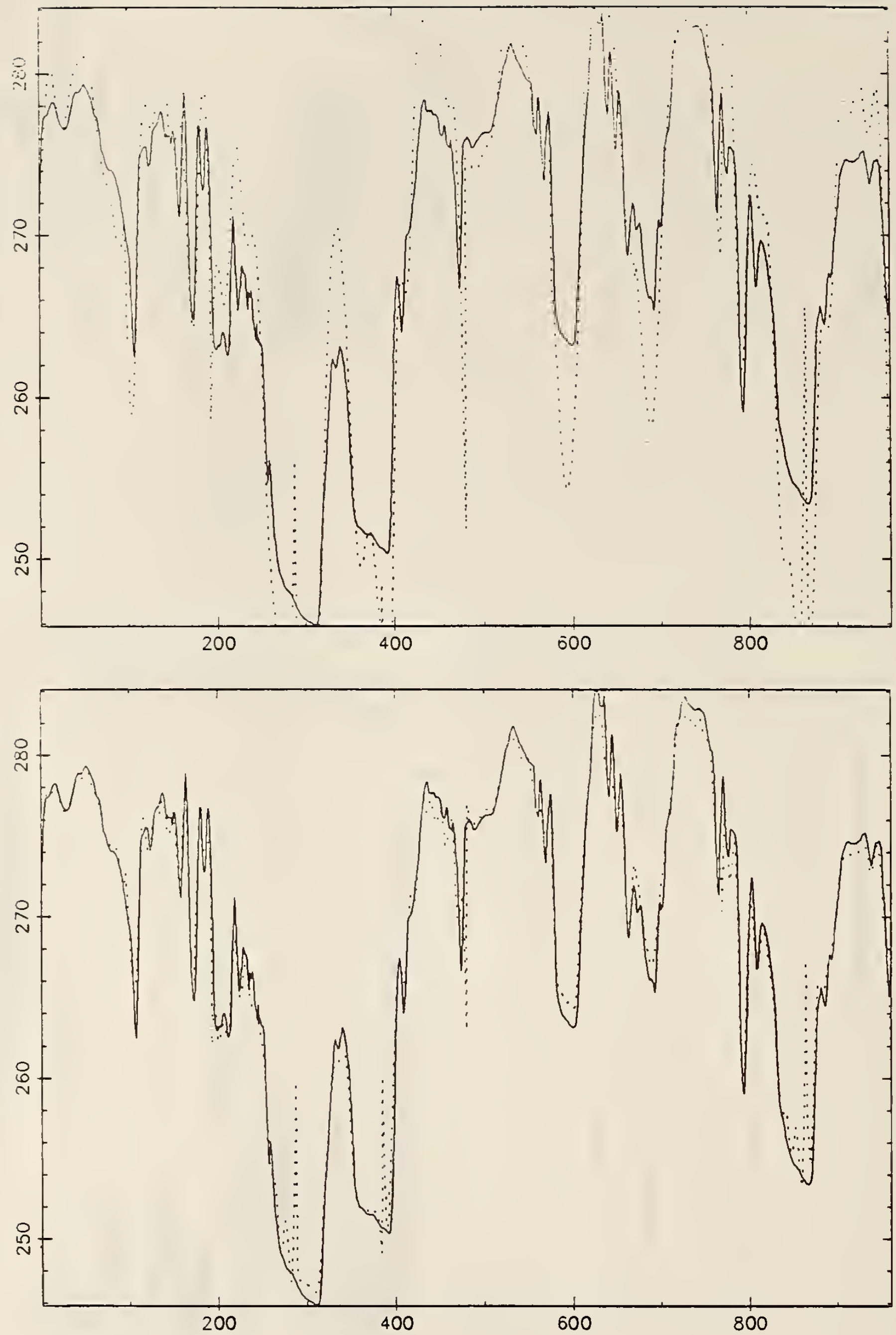

Figure 2 Smoothed sky temperature vs. Fourier reconstruction: time in quarter hours vs. kelvin. A. Solid line: smoothed data; dotted line: data generated using first 4 harmonics present in greatest intensity in Fourier decomposition. B. Solid line: smoothed data; dotted line: data generated using first 8 harmonics.

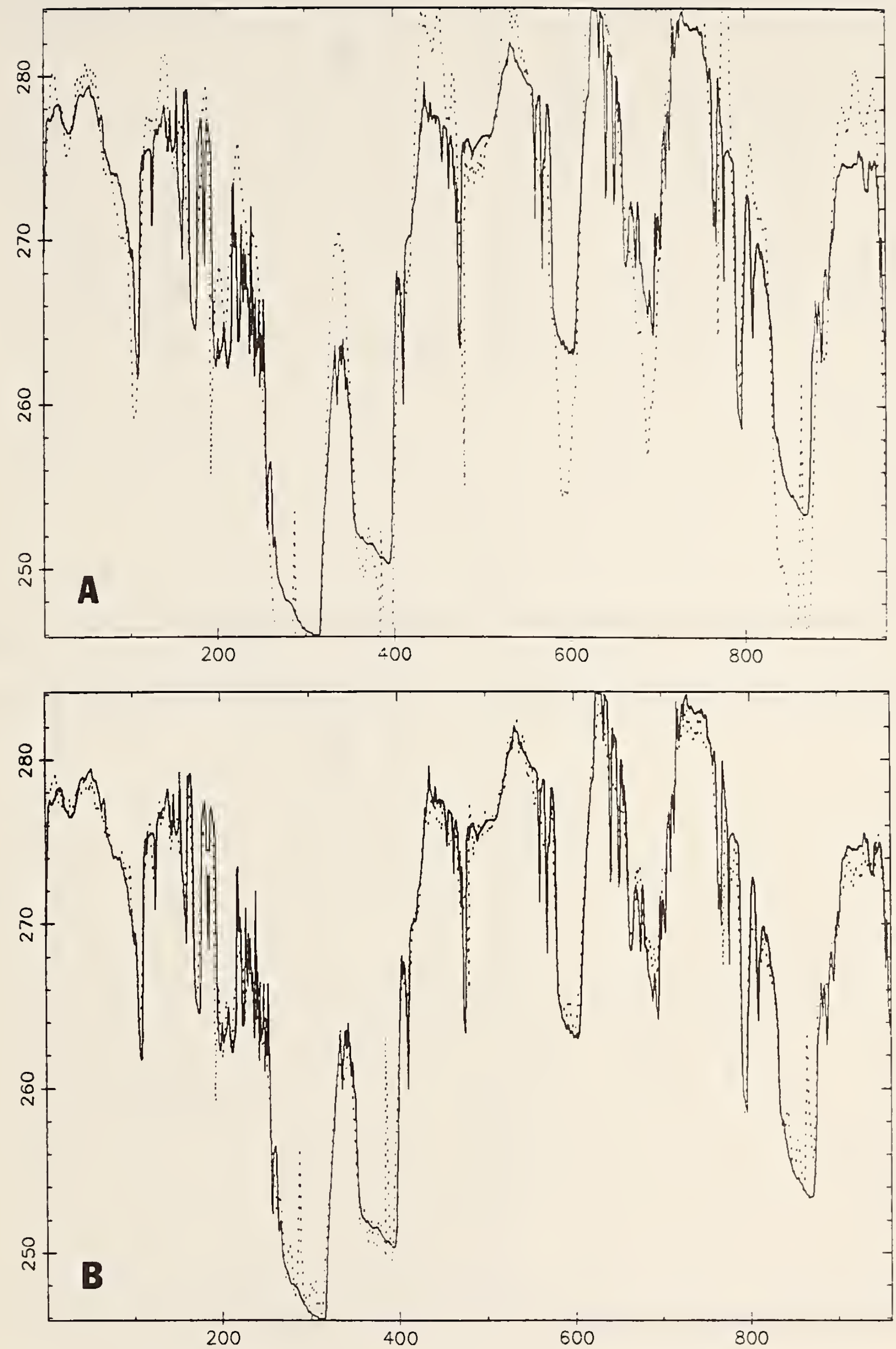

Figure 3 Original sky-temperature data vs. reconstructed then perturbed data. A. Solid line: original data; dotted line: reconstructed then perturbed data (4 Fourier components). B. As above, using 8 Fourier components. Time in quarter hours vs. kelvin.

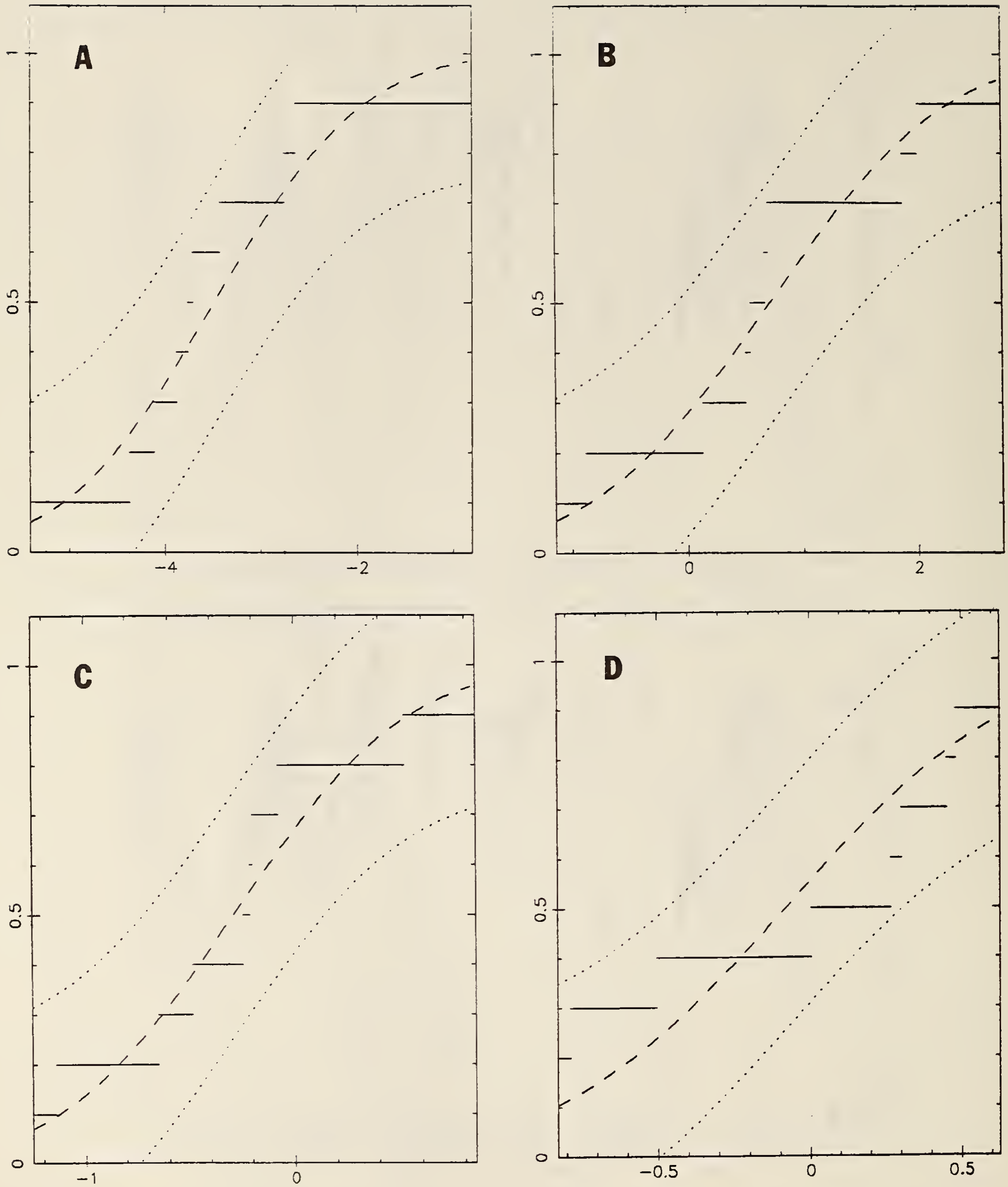

Figure 4 Distributions of first 4 cosine coefficients. Solid: sample distribution; dashed: normal distribution with same mean and variance as sample; dotted: Kolmogorov-Smirnov 90% goodness-of-fit limits.

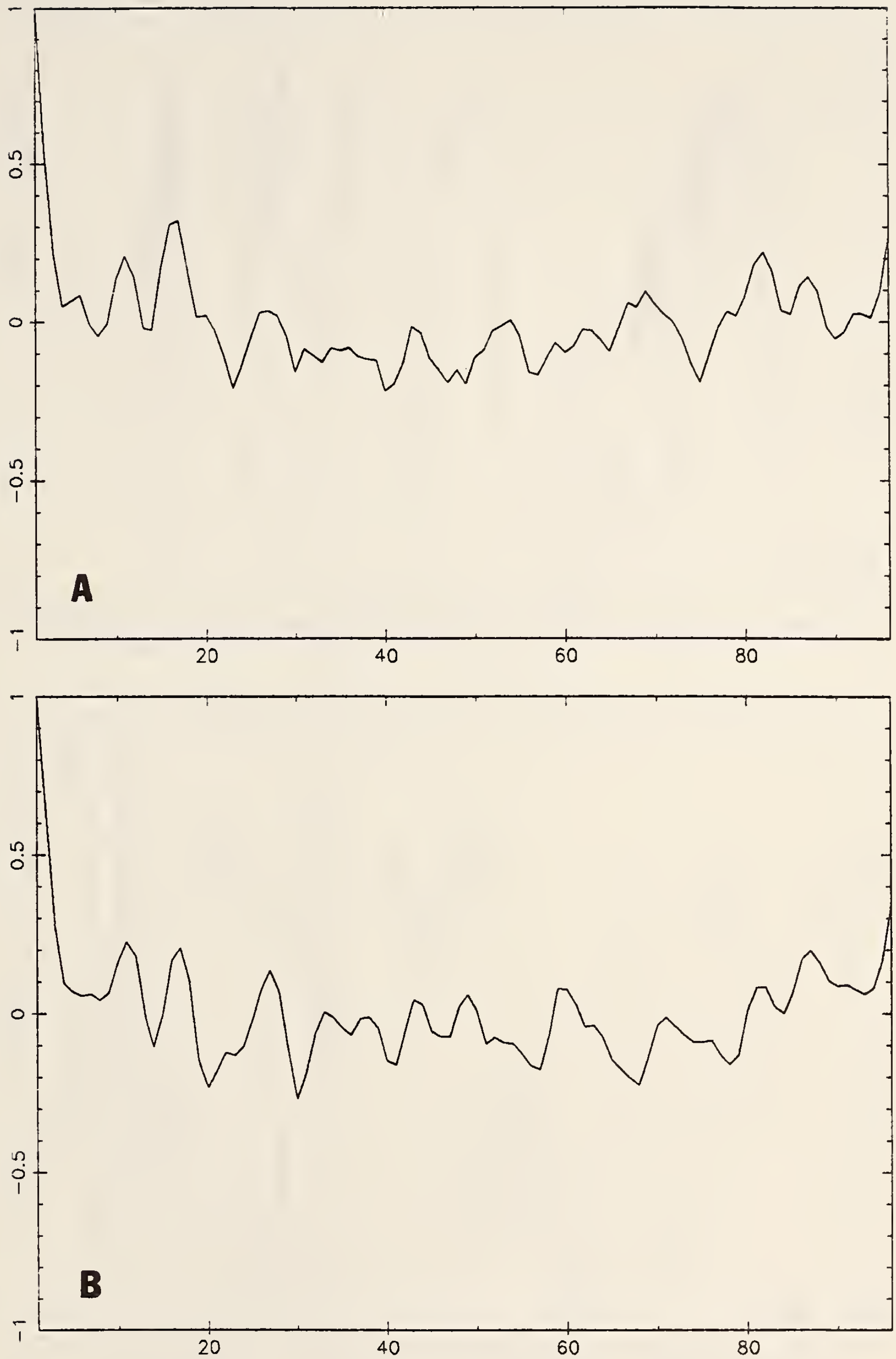

Figure 5 Autocorrelation of smoothing error for sky-temperature data. Autocorrelation of averaged daily smoothing error (A.) vs. that of simulated error: time in quarter hours vs. correlation.

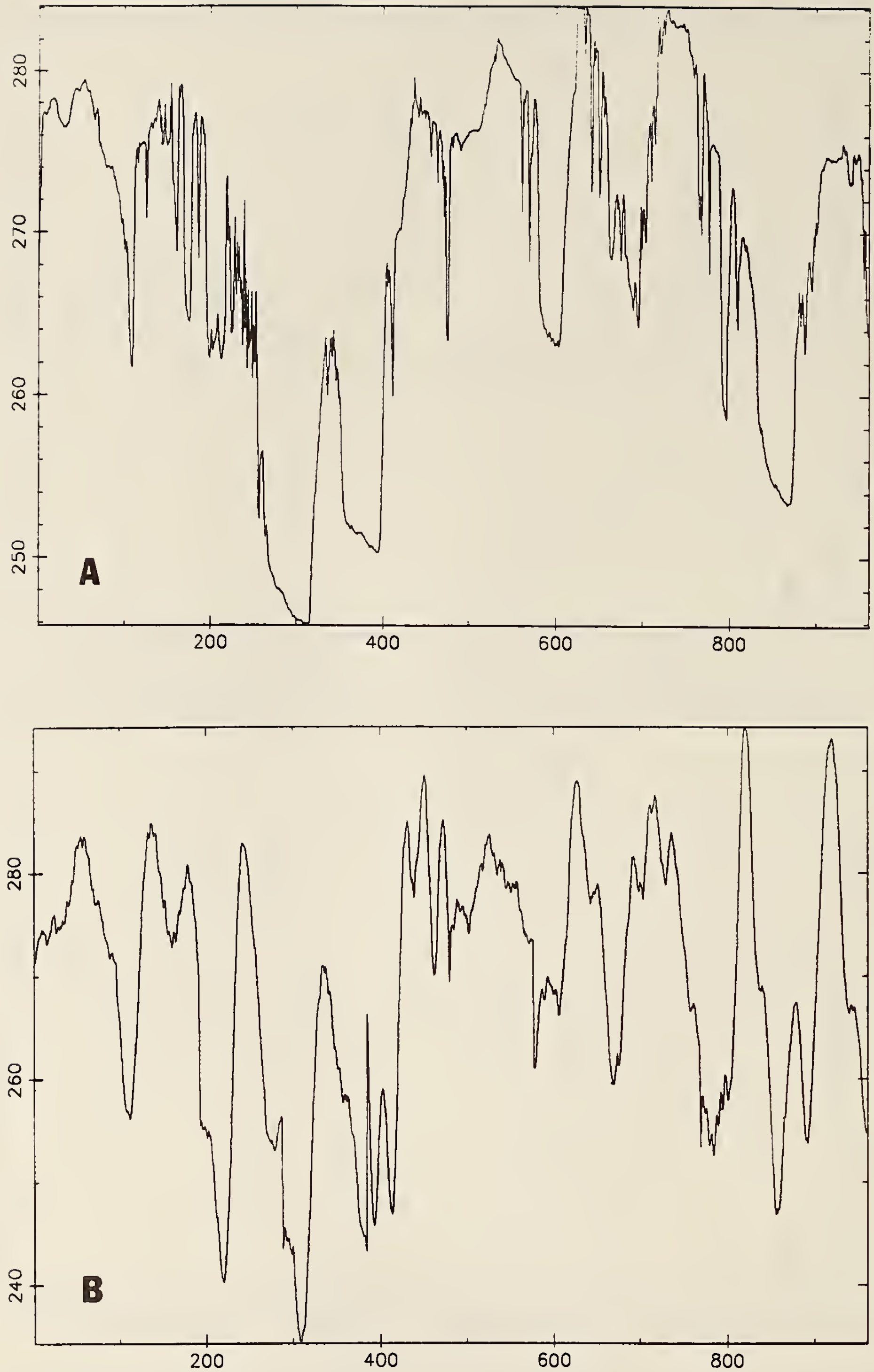

Figure 6 Original sky temperature data (A.) vs. simulated series: time in quarter hours vs. langleys.

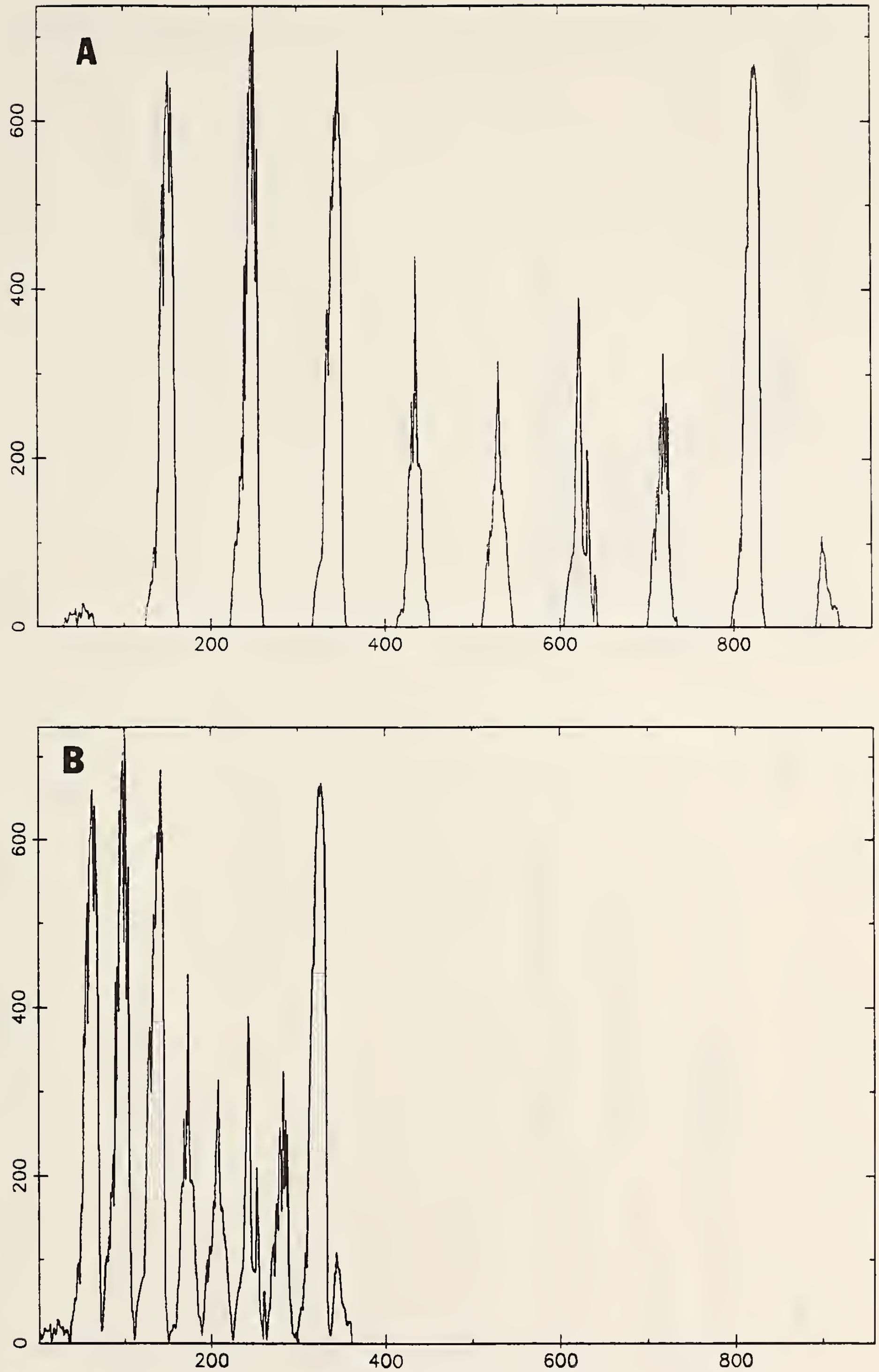

Figure 7 Incident radiation data: time in quarter hours vs. langleys. Original (A.) vs. compressed.

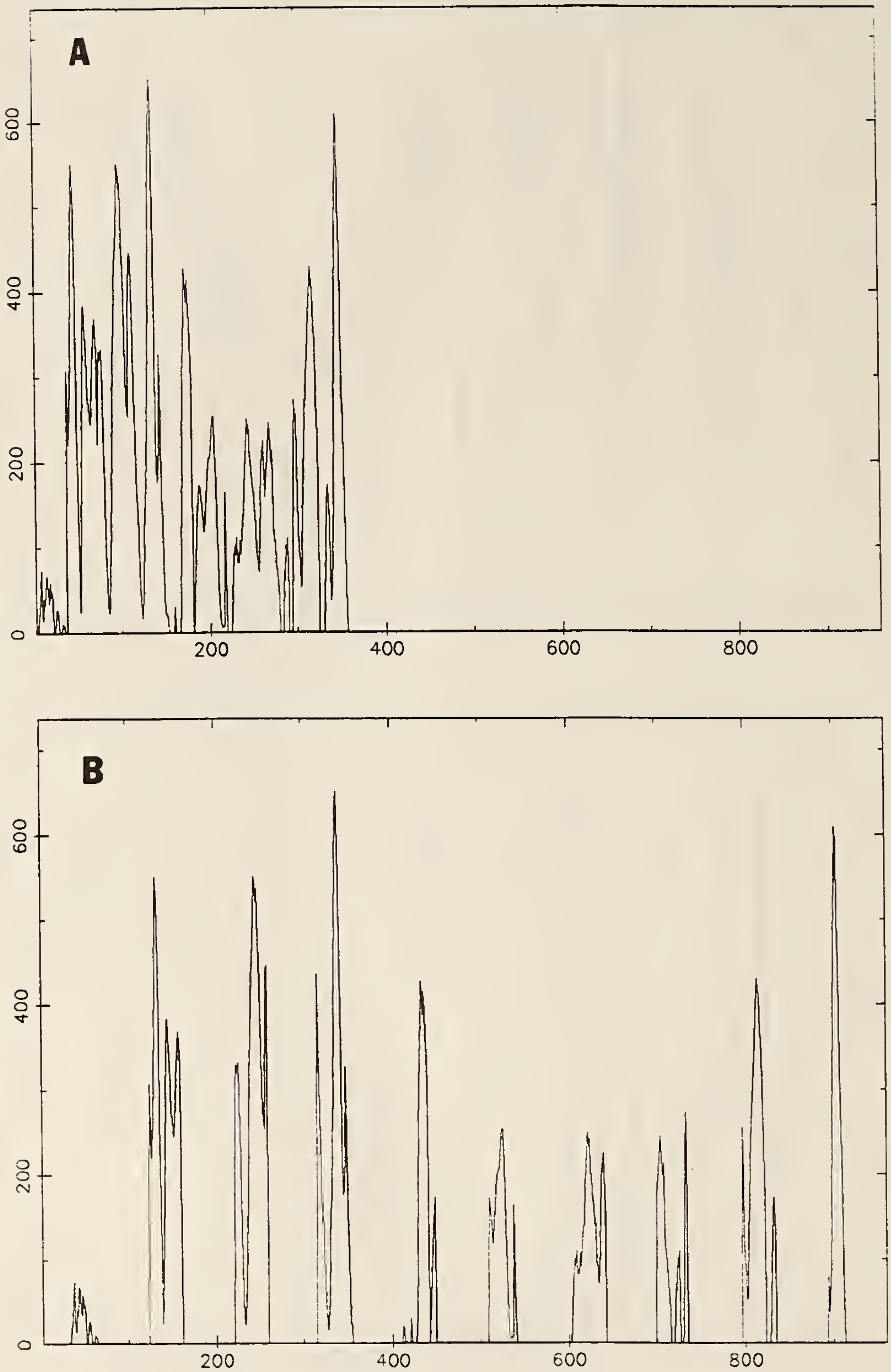

Figure 8 Simulated incident radiation data (A.) vs. same "decompressed": time in quarter hours vs. langleys.

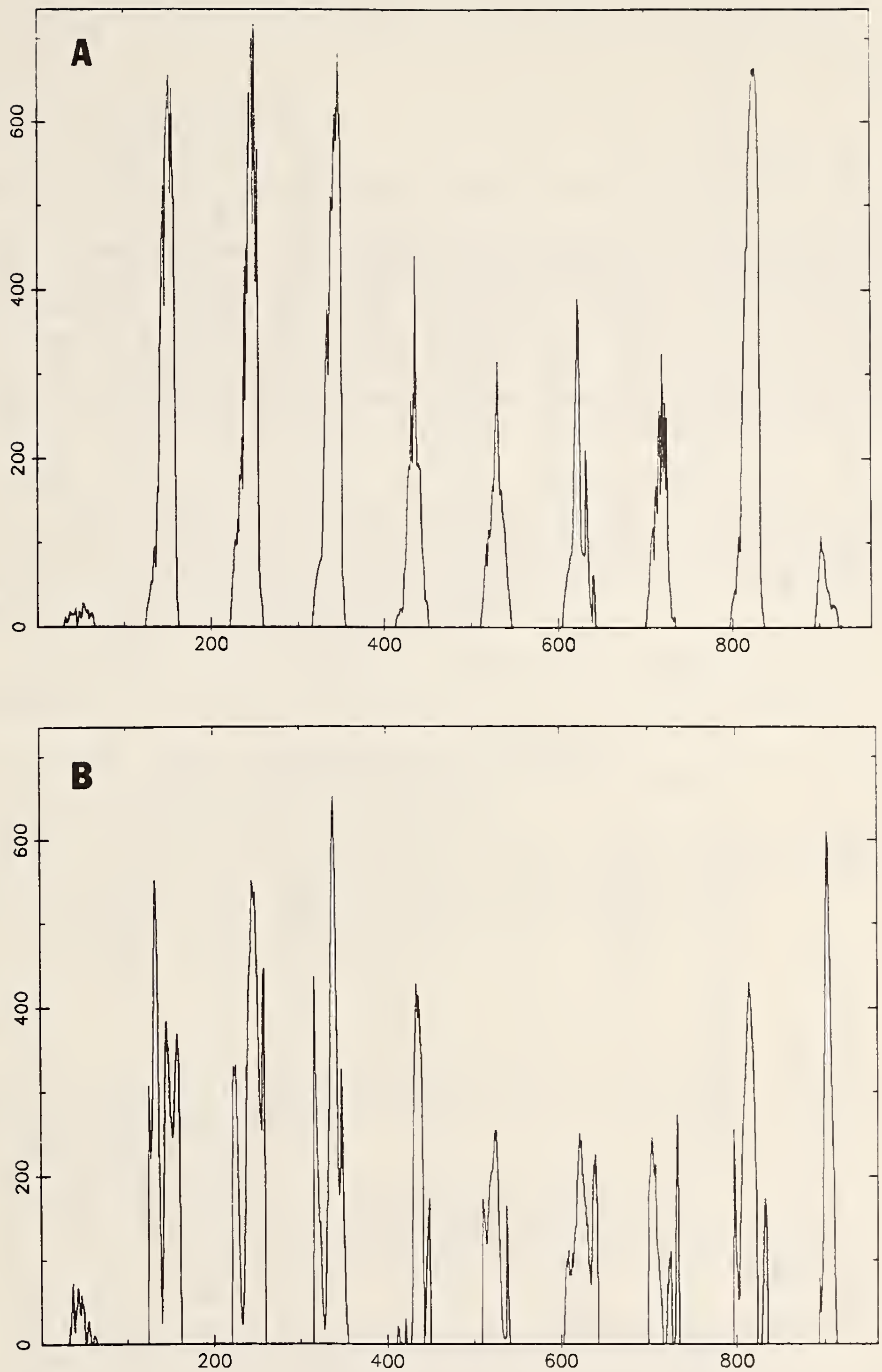

Figure 9 Original incident radiation data (A.) vs. simulated series: time in quarter hours vs. langleys.

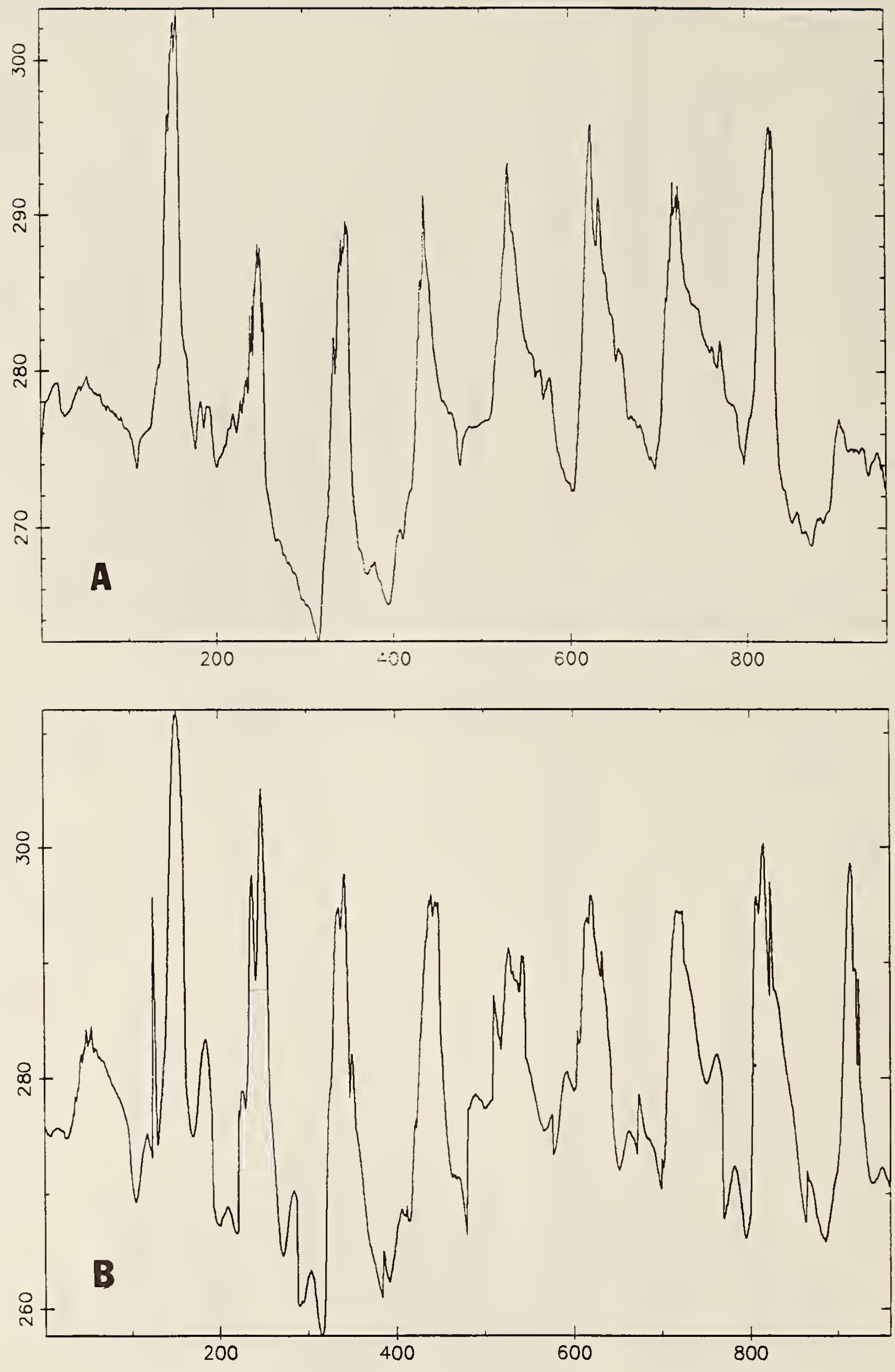

Figure 10 Original panel-temperature data (A.) vs. steady-state model eq driven by simulated variates: time in quarter hours vs. kelvin.

APPENDIX A: PROGRAM SIMUL

```
*************************************************************
*
* Analyzes time series of observation vectors by smoothing
* component variate series, performing Fourier decomposition
* upon component variates, and fitting a normal distribution
* to the sample coefficient distributions. These distributions
* are used to govern the random selection of Fourier coefficients
* for a smooth simulated process. The smooth simulation vector
* time series is modified by a simulated error vector time series.
*
* This error series is formed by first obtaining the mean,
* variance, and autocorrelation matrix of the one-day averaged
* error vector time series, and imposing those quantities upon
* a normally distributed random vector series. Full details can
* be found in the accompanying report.
*
*
* PARAMETERS:
*
* NDOBS: no. of daily observations recorded
* NDAYS: no. of days over which sample was take
* LEN: total observations in sample
* NTERMS: no. of Fourier harmonics to be included in simulation
* NDSIM no .of days of simulated observations desired
* LENSIM: total no. of observations in simulation
* NDRV: no. of driving variables
* TPI: 2. * pi
*
* ARRAYS:
*
* X(LEN): raw data for each variate
* XSM(LEN): smoothed data
* XDIFF: smoothing error
* ADAY: daily means
* VDAY: daily variations ( (high-low)/2. )
* XNRM: normalized data partitioned into daily subseries
* CSQR: square root of autocorrelation matrix of average daily
*       smooothing error
* BUMP: simulated smooting error over one day's observations
* XSIM: simulated observations of current variate
* FMEAN: means of daily sample Fourier coefficients
* FSDEV: standard deviation of daily sample Fourier coefficients
* SIMCOF: sample Fourier sine and cosine coefficients
* VSIM: simulated vector time series
* WK*: workspace arrays for subroutines
*
* NVAR: index array for input routine, contains variate cardinals
```

```
*
* SIMPLE VARIABLES:
*
* SDEV: standard deviation of averaged daily smoothing error
* NVOBS: virtual no. of daily observations
* SUMS: simulation accumulator, accumulates over Fourier
     components

*****************************************************************

      parameter (ndobs = 96, ndays = 10, len = ndobs*ndays,
       nterms = 4, ndsim = 10, ndrv = 4,lensm = ndobs*ndsim,
       tpi = 2.*3.141592654)

      real x(len), xsm(len), vday(ndays), aday(ndays),
       xdiff(len),  csqr(ndobs,ndobs),xnrm(ndobs,ndays),bump(ndobs),
       xsim(lensm),fmean(2, nterms), fsdev(2,nterms),
       simcof(2,nterms),wk0(ndobs), wk1(ndobs),wk2(ndobs),
       wk3(ndobs,ndobs),wk4(ndobs/2+1, 5, ndays), wk5(ndays),
       vsim(lensm, ndrv)

      integer nvar(ndrv)

      data nvar / 1, 2, 9, 10 /

      do 1000 n = 2,2


      call getx (ndays*ndobs, nvar(n), x)

      ncmp = 0

      if (nvar(n).eq.9) call compress (ndays, ndobs, x,
           ncmp, lennil, lenful, nphase, wk0)

      nvobs = ndobs-ncmp

      call smooth (ndays, nvobs, x, xsm, xdiff)

      call getsamp (ndays, nvobs, xsm, aday, vday)

      call normize (ndays, nvobs, xsm, aday, vday, xnrm)

      call getbump (ndays, nvobs, xdiff, sdev, csqr, wk0, wk1,
           wk2, wk3)

      call getdist (ndays, nvobs, nterms, xnrm, fmean, fsdev, wk4,
           wk5)

      do 30 j = 1, ndsim
         call getsim (nvobs, nterms, fmean, fsdev, simcof, sdev,
           bump, csqr, wk1)
```

```
      do 30 i = 1, nvobs
           ci = sngl( i )
           sums = 0.0
           do 35 k = 1, 2
           do 35 l = 1, nterms
              f = tpi*( sngl(l) / sngl(nvobs) )
              if (k.eq.1) then
                sums = sums + simcof(k, l)*cos( f*ci )
              else
                sums = sums + simcof(k, l)*sin( f*ci )
              endif
35            continue
           ii = nvobs * (j-1) + i
           sums = sums / sngl(nterms)
           xsim (ii) = sums*vday(j) + aday(j) + bump(i)
30         continue

         if (nvar(n).eq.9) then
           call decompress (ndsim, ndobs, x, xsim, ncmp,
     1                      lennil, lenful, nphase)
           do 40 i = 1, lensm
              if (x(i).lt.0.) x(i) = 0.
              xsim(i) = x(i)
40            continue
         endif

      do 1000 i = 1, lensm
         vsim(i,n) = xsim(i)

1000  continue

      do 2000 i = 1, lensm
         t(i) = sngl(i)
2000     continue

      call simwrite (ndrv, ndsim, ndobs, vsim)

      end
```

```
***********************************************************************

     subroutine getsamp( ndays, ndobs, x, aday, vday)

***********************************************************************
*
* Obtains vectors of daily means and variations.
*
* ARRAYS:
*
* X: raw data (input)
* ADAY: daily average temps (output)
* VDAY: daily average variations ( (hi-lo)/2. ) (output)


*
* SIMPLE VARIABLES:
*
* NDAYS: no. of days in sample (input)
* NDOBS: no. of observations/day (input)
*
* EXTERNAL ROUTINES:
*
* SSUM: computes the sum of elements of a given array (IMSL
     Library)
***********************************************************************
     real x(ndays*ndobs), aday(ndays), vday(ndays)

     len = ndays*ndobs
     dmax = x(1)
     dmin = dmax
     j = 0

     do 100 i = 1, len

        if (dmax.lt.x(i)) dmax = x(i)
        if (dmin.gt.x(i)) dmin = x(i)

        if ( mod(i,ndobs).eq.0 ) then
          j = j+1
          vday(j) = (dmax - dmin)/2.
          dmax = x(i+1)
          dmin = dmax
          endif
100       continue
     do 200 i = 1, ndays
        aday(i) = ssum(ndobs,x((i-1)*ndobs + 1),1) / sngl(ndobs)
200     continue

        return
        end
```

```
*********************************************************************

     subroutine getx (nobs, nvar, x)

*********************************************************************

* Reads raw data from input file (logical name: 'NP'). Format of
     the read is determined by data -- see accompanying report,
     table on pg. 5.
*
* ARRAYS:
*
* X: contains raw data (output)
* A: input variable, reads line of data (10 variates)
*
* SIMPLE VARIABLES:
*
* NOBS: total no. of observations
* NVAR: selects variate to be read
*
*********************************************************************


     real x(nobs), a(10)

     open (10, file='np', status='old')

     do 100 i = 1, nobs

        read (10, *) ii, jj, kk, (a(j),j=1,10)

        x(i) = a(nvar)

100     continue

     close (10)

     return
     end
```

```
***********************************************************************

      subroutine smooth (ndays, ndobs, x, xsm, xdiff)

************************************************************************
* Performs moving average calculation on input data in order to
* smooth it. Also returns error array.
*
* PARAMETER:
*
* NUM: diameter of moving average
*
* ARRAYS:
*
* X: raw data (input)
* XSM: smoothed data (output)
* XDIFF: X-XSM, smoothing error (output)
*
* SIMPLE VARIABLES:
*
* LEN: total no. of observations
* SUM: accumulator
*
**********************************************************************

      parameter (num = 2)

      real x(ndays*ndobs), xsm(ndays*ndobs), xdiff(ndays*ndobs)

      len = ndays*ndobs

      do 200 i = 1, len
         sum = 0.
         do 100 j = -num, num

! mod() causes data to wrap around...

            sum = sum + x( mod( i+j+len, len )+1 )
100         continue

         xsm(i) = sum / (2.*sngl(num) + 1.)
         xdiff(i) = x(i) - xsm(i)

200      continue

      return
      end
```

```
*****************************************************************

     subroutine normize (nd, ndobs, xsm, aday, vday, xnrm)

*******************************************************************
*
* Forms standardized (zero mean, range [-1, +1]), partitioned
* (into daily subseries) series of observations from smoothed
* data.
*
* ARRAYS:
*
* XSM: smoothed data (input)
* ADAY: daily mean temps (input)
* VDAY: daily variations (input)
* XNRM: standardized partitioned data (output)
*

* SIMPLE VARIABLES:
*
* NDOBS: no. of daily observations
* NDAYS: no. of days in sample
*
*******************************************************************

     real xsm(ndobs*nd), xnrm(ndobs, nd), aday(nd), vday(nd)

     do 100 j = 1, nd
        do 100 i = 1, ndobs
          xnrm(i,j) = (xsm(ndobs*(j-1)+i) - aday(j) ) / vday(j)
100       continue

     return
     end
```

```
*****************************************************************

     subroutine getbump (ndays, ndobs, xdiff, sdev, csqr, ac,
          acv,xdavg, corm)

*****************************************************************

* Obtains average daily smoothing error and computes its standard
* deviation and autocorrelation matrix for use in producing
     simulated
* smoothing error.
*
* ARRAYS:
*
* XDIFF: actual smoothing error (input)
* CSQR: square root of autocorrelation matrix (output)
* CORM: autocorrelation matrix (input as workspace)
* AC: vector of autocorrelations (input as workspace)
* ACV: vector of autocovariances (input as workspace)
* XDAVG: contains two copies of average daily smoothing error
*  (two copies to make the data look cyclic to the autocorrelation
     routine)
*  (input as workspace)
* STAT: contains univariate statistics on daily average error
     series
*  (input as workspace)
*
* SIMPLE VARIABLES:
*
* SDEV: standard deviation of average daily error (output)
* SUM: accumulator
*
* EXTERNAL ROUTINES:
*
* UVSTA: computes several univariate statistics of given time
     series
*  (IMSL Library)
* ACF: computes autocorrelation and autocovariance functions of
     given time
*  series (IMSL Library)
* LFTDS: computes Cholesky decomposition of symmetric, positive
     definite
*  matrix (the autocorrelation matrix, in this case) (IMSL Library)

*
*****************************************************************
     real xdiff(ndays*ndobs), xdavg(ndobs*2), ac(ndobs),
     acv(ndobs), stat(15), corm(ndobs, ndobs), csqr(ndobs, ndobs)

! compute average daily error series...
```

```
      do 10 i = 1, ndobs
         sum = 0.0
         do 15 j = 1, ndays
           sum = sum + xdiff( ndobs*(j-1) + i )
15         continue
         xdavg(i) = sum / sngl(ndays)
         xdavg(i+ndobs) = xdavg(i)
10       continue

! compute univariate statistics...

      call uvsta( 0, ndobs, 1, xdavg, ndobs, 0, 0, 1, 95., 95., 0,
      stat, 15, nm)

      sdev = stat(3)

! compute autocorrelation over lags up to ndobs...

      call acf( ndobs*2, xdavg, 0, 0, 0, 0.0, ndobs-1, acv, ac,
      seac)

! recover autocorrelation matrix...

      do 20 i = 1, ndobs
      do 20 j = 1, ndobs
         corm(i, j) = ac(j-i+1)
20       continue

! get its square root...

      call lftds (ndobs, corm, ndobs, csqr, ndobs)

      return
      end
```

```
*****************************************************************

     subroutine getdist (ndays, ndobs, nterms, xnrm, fmean,fsdev,
          pm, dat)

*****************************************************************
*
* Computes the fast Fourier transform of the standardized data,
     obtaining
* means and variances of ten first NTERM coefficient sets over
     NDAYS of
* sample subseries.
*
* ARRAYS:
*
* XNRM: standardized, partitioned data (input)
* FMEAN: means of coefficient samples  (output)
* FSDEV: std. deviations of coefficient samples (output)
* PM: periodogram matrix, contains frequencies, periods,
     intensities,
*  cosine coefficients, sine coefficients (input as workspace)
* STAT:  univariate statistics on coefficient samples (input as
     workspace)
* DAT: holds one coefficient sample at a time for processing
* SIMPLE VARIABLES:
*
* LDPM: leading dimension of PM (input)
*
* EXTERNAL ROUTINES:
*
* PFFT: computes FFT in form of a periodogram for sample time
     series
*  (IMSL Library)
* UVSTA: computes several univariate statistics for a given time
     series
*  (IMSL Library)
*****************************************************************
     real xnrm(ndobs, ndays), fmean(2, nterms), fsdev(2, nterms),
     pm(ndobs/2+1, 5, ndays), dat(ndays), stat(15)

     ldpm = ndobs/2 + 1

     do 10 j = 1, ndays

! compute fast fourier transform for each daily subseries...

        call pfft( ndobs,xnrm(1,j),0,0.0,0,1,0,pm(1,1,j),ldpm)
10      continue

! k index: 1: cosine, 2: sine
```

```
      do 15 k = 1, 2
      do 15 l = 1, nterms
         do 20 j = 1, ndays

! add right amount to indices to pick out correct columns of PM...

            dat(j) = pm(l+1, k+3, J)
20          continue

! compute univariate stats for coefficient sample...

         call uvsta(0,ndays,1,dat,ndays,0,0,1,95.,95.,0,stat,15,nm)

         fmean(k, l) = stat(1)
         fsdev(k, l) = stat(3)
15       continue
      return
      end
```

```
*****************************************************************

     subroutine getsim(ndobs,nterms,fmean,fsdev,simcof,sdev,bump,
     csqr, xrnd)

******************************************************************
*
* Randomly chooses Fourier coefficients for one-day simulation
* according to normal distribution determined by GETDIST;generates
* a one-day dependent series of random variates for use in
* unsmoothing simulation.
*
* ARRAYS:
*
* FMEAN: means of daily Fourier coefficient distributions (input)
* FSDEV: std. deviations of daily Fourier coefficient distributions
      (input)
* SIMCOF: randomly selected Fourier coefficients (output)
* BUMP: dependent random series to simulate smoothing error
      (output)
* CSQR: square root of autocorrelation of average daily smoothing
     error,
*  used to induce proper correlations onto random number series
      (input)
* XRND: vector of normally distributed random numbers (input as
     workspace)
*
* SIMPLE VARIABLES:
*
* NDOBS: no. of daily observations
* NTERMS: no. of Fourier components to retain
*
* EXTERNAL ROUTINES:
*
* RNNOF: generates single normally distributed random number
* RNNOR: generates series of nnormally distributed random numbers
*  (IMSL Library)
* SSCAL: scales an array by a scalar (IMSL Library)
* MURRV: premultiplies a real matrix by a vector (IMSL Library)
*
******************************************************************

     real fmean(2, nterms), fsdev(2, nterms), simcof(2, nterms),
         bump(ndobs), csqr(ndobs, ndobs), xrnd(ndobs)

     do 10 k = 1, 2
     do 10 l = 1, nterms

! generate simulation coefficients...

        simcof(k, l) = rnnof()*fsdev(k ,l) + fmean(k, l)
10      continue
```

```
! generate normally distributed random vector...

      call rnnor(ndobs, xrnd)

! scale it by std. dev. of actual daily error...

      call sscal(ndobs, sdev, xrnd, 1)

! multiply it by (autocorrelation matrix)**0.5 to induce correct
      correlations...

      call murrv(ndobs,ndobs,csqr,ndobs,ndobs,xrnd,1,ndobs,bump)

      return
      end
```

```
*************************************************************

      subroutine simwrite (ndrv, ndsim, ndobs, vsim)

*************************************************************
      real vsim(ndsim*ndobs, ndrv)

      open (10, file='sim.dat', status = 'new')

      do 10 i = 1, ndsim*ndobs
         write(10, *) (vsim(i, j), j = 1, ndrv)
10       continue

      close (10)

      return
      end
```

```
***********************************************************************
      subroutine compress(ndays,ndobs,x,ncmp,lennil,lenful,nphase,
           cx)

***********************************************************************
* Removes nil observations from incident radiation data, returns
      compressed
* series of observations and information on arrangement of nil
      observations.
*
* ARRAYS:
*
* X: input radiation data; output compressed data
* CX: intermediate storage array (input as workspace)
*
* SIMPLE VARIABLES:
*
* NDOBS: no. of daily observations (input)
* NDAYS: no. of days in sample (input)
* NCMP: average daily total no. of nil observations (output)
* LENNIL: average daily no. of consecutive nil observations
      (output)
* LENFUL: average daily no. of consecutive non-zero observations
      (output)
* NPHASE: length of first series of consecutive zero (non-zero)
      observations:
*  abs(NPHASE): length, NPHASE < 0 => first record was zero,
*     NPHASE > 0 => first record was non-zero (output)
* NFLAG: if set (1), last record read was non-zero; if clear (0),
      last record
*  read was zero
* NFIRST: if -1, first record was zero; if +1, first record was
      non-zero
* MLN: counts number of zero record subseries
* MLF: counts number of non-zero record subseries
* J: running index of compressed series; incremented only when
      series is updated
***********************************************************************

      real x(ndays*ndobs), cx(ndays*ndobs)

      len = ndays*ndobs

      j = 1
      ncmp = 0

      mlf = 0
      mln = 0

      if ( x(1).eq.0.0 ) then
         nfirst = -1
```

```
        nflag = 0
     else
        nfirst = 1
        nflag = 1
     endif

     ln = 0
     lf = 0

     do 100 i = 1, len

        if ( x(i).eq.0.0 ) then
          ncmp = ncmp+1

          if (nflag.eq.1) then
             mlf = mlf+1
             if((mlf.eq.1).and.(nfirst.gt.0))nphase=nfirst*lenful
                     nflag = 0
          endif

        else
          cx(j) = x(i)
          j = j+1
          lenful = lenful + 1

          if (nflag.eq.0) then
             mln = mln+1
             if((mln.eq.1).and.(nfirst.lt.0))nphase = nfirst*ncmp
                     nflag = 1
          endif

        endif
100     continue

     if ( (x(len).ne.0.0).and.(nfirst.lt.0) ) mlf = mlf+1
     if ( (x(len).eq.0.0).and.(nfirst.gt.0) ) mln = mln+1

     lennil = ncmp / mln
     lenful = lenful / mlf

     do 200 i = 1, len-ncmp
        x(i) = cx(i)
200     continue

     ncmp = ncmp / ndays

     return
     end
```

```
**********************************************************************
      subroutine decompress(ndays, ndobs, x, cx, ncmp, lennil,
      lenful, nphase)

**********************************************************************
*
* Introduces nil observations into compressed simulated incident
      radiation
* data according to information provided by COMPRESS.
*
* ARRAYS:
*
* X: input simulated compressed data; output simulated decompressed
      data
* CX: intermediate storage array (input as workspace)
*
* SIMPLE VARIABLES:
*
* NDOBS: actual no. of daily observations (input)
* NDAYS: no. of days in sample (input)
* NCMP: average daily total no. of nil observations (input)
* LENNIL: average daily no. of consecutive nil observations in
      uncompressed
*  data(input)
* LENFUL: average daily no. of consecutive non-zero observations in

*  uncompressed data (input)
* NPHASE: length of first series of consecutive zero (non-zero)
      observations
*  in uncompressed data:
*  abs(NPHASE): length, NPHASE < 0 => first record was zero,
*     NPHASE  > 0 => first record was non-zero (input)
* NPH: abs(NPHASE)
* J: running index for compressed data array, incremented only when
      array is read
**********************************************************************
      real x(ndays*ndobs), cx(ndays*(ndobs-ncmp))
      nph = abs(nphase)

      do 5 i = 1, ndays*ndobs
         x(i) = 0.
5        continue

      j = 0

      do 100 k = 1, ndays
         kn = (k-1)*ndobs

         if (nphase.gt.0) then

! fill first NPHASE records of array...
```

```
          do 10 i = kn+1, kn+nph
             x(i) = cx(j)
             j = j+1
10           continue

        endif
        if (nphase.lt.0) then

! skip first NPHASE records of array (i.e. fill first NPHASE
records
! with zeros) and fill next LENFUL records...

          do 20 i = kn+nph+1, kn+nph+1+lenful
             x(i) = cx(j)
             j = j+1
20           continue

        endif

        if (nphase.gt.0) then

! skip next LENNIL records and fill to end of day...

          do 30 i = kn+nph+lennil+1, kn+ndobs
             x(i) = cx(j)
             j = j+1
30           continue

        endif

100     continue

      return
      end
```

NIST-114A (REV. 3-89)	U.S. DEPARTMENT OF COMMERCE NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY	1. PUBLICATION OR REPORT NUMBER NIST/TN-1275
		2. PERFORMING ORGANIZATION REPORT NUMBER
	BIBLIOGRAPHIC DATA SHEET	3. PUBLICATION DATE April 1990

4. TITLE AND SUBTITLE

A Study of Meteorological Processes Important in the Degradation of Materials Through Surface Temperature

5. AUTHOR(S)

Sam C. Saunders, Mark A. Jensen, and Jonathan W. Martin

6. PERFORMING ORGANIZATION (IF JOINT OR OTHER THAN NIST, SEE INSTRUCTIONS)

U.S. DEPARTMENT OF COMMERCE
NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY
GAITHERSBURG, MD 20899

7. CONTRACT/GRANT NUMBER

8. TYPE OF REPORT AND PERIOD COVERED

Final

9. SPONSORING ORGANIZATION NAME AND COMPLETE ADDRESS (STREET, CITY, STATE, ZIP)

Same as item #6

10. SUPPLEMENTARY NOTES

[ ] DOCUMENT DESCRIBES A COMPUTER PROGRAM; SF-185, FIPS SOFTWARE SUMMARY, IS ATTACHED.

11. ABSTRACT (A 200-WORD OR LESS FACTUAL SUMMARY OF MOST SIGNIFICANT INFORMATION. IF DOCUMENT INCLUDES A SIGNIFICANT BIBLIOGRAPHY OR LITERATURE SURVEY, MENTION IT HERE.)

One of the greatest impediments in forecasting the service life of a material exposed outdoors is the uncontrolled and non-predictable nature of the ambient factors comprising its environment. This contributes to the difficulty of establishing the cause-and-effect relationship between these factors and the rate of material degradation. To surmount these difficulties it is necessary to characterize quantitatively each of the factors comprising the exposure environment which are thought to be important in the material's degradation. The selection and quantification of these factors must accommodate not only the periodicity of the diurnal cycle but its statistical fluctuation. The objective of this research is to develop a general mathematical model, through Fourier analysis, characterizing the diurnal variation in the primary factor, material temperature. This factor is felt to be important in the degradation of a wide range of materials and protective systems, including coated steel panels. The steps taken and problems encountered in developing such a model are outlined. It is concluded that the simulated data generated from this model display virtually the same stochastic behavior as does the real data and that, if appropriate meterolocial records are available, it will be possible to characterize and reproduce the statistical behavior for any locality, season and panel orientation.

12. KEY WORDS (6 TO 12 ENTRIES; ALPHABETICAL ORDER; CAPITALIZE ONLY PROPER NAMES; AND SEPARATE KEY WORDS BY SEMICOLONS)

diurnal cycle; environmental factor; energy-balance equation; Fourier analysis; meteorological data; panel temperature.

13. AVAILABILITY

[X] UNLIMITED
[ ] FOR OFFICIAL DISTRIBUTION. DO NOT RELEASE TO NATIONAL TECHNICAL INFORMATION SERVICE (NTIS).
[X] ORDER FROM SUPERINTENDENT OF DOCUMENTS, U.S. GOVERNMENT PRINTING OFFICE, WASHINGTON, DC 20402.
[X] ORDER FROM NATIONAL TECHNICAL INFORMATION SERVICE (NTIS), SPRINGFIELD, VA 22161.

14. NUMBER OF PRINTED PAGES

45

15. PRICE

ELECTRONIC FORM

NIST Technical Publications

Periodical

Journal of Research of the National Institute of Standards and Technology—Reports NIST research and development in those disciplines of the physical and engineering sciences in which the Institute is active. These include physics, chemistry, engineering, mathematics, and computer sciences. Papers cover a broad range of subjects, with major emphasis on measurement methodology and the basic technology underlying standardization. Also included from time to time are survey articles on topics closely related to the Institute's technical and scientific programs. Issued six times a year.

Nonperiodicals

Monographs—Major contributions to the technical literature on various subjects related to the Institute's scientific and technical activities.

Handbooks—Recommended codes of engineering and industrial practice (including safety codes) developed in cooperation with interested industries, professional organizations, and regulatory bodies.

Special Publications—Include proceedings of conferences sponsored by NIST, NIST annual reports, and other special publications appropriate to this grouping such as wall charts, pocket cards, and bibliographies.

Applied Mathematics Series—Mathematical tables, manuals, and studies of special interest to physicists, engineers, chemists, biologists, mathematicians, computer programmers, and others engaged in scientific and technical work.

National Standard Reference Data Series—Provides quantitative data on the physical and chemical properties of materials, compiled from the world's literature and critically evaluated. Developed under a worldwide program coordinated by NIST under the authority of the National Standard Data Act (Public Law 90-396). NOTE: The Journal of Physical and Chemical Reference Data (JPCRD) is published quarterly for NIST by the American Chemical Society (ACS) and the American Institute of Physics (AIP). Subscriptions, reprints, and supplements are available from ACS, 1155 Sixteenth St., NW., Washington, DC 20056.

Building Science Series—Disseminates technical information developed at the Institute on building materials, components, systems, and whole structures. The series presents research results, test methods, and performance criteria related to the structural and environmental functions and the durability and safety characteristics of building elements and systems.

Technical Notes—Studies or reports which are complete in themselves but restrictive in their treatment of a subject. Analogous to monographs but not so comprehensive in scope or definitive in treatment of the subject area. Often serve as a vehicle for final reports of work performed at NIST under the sponsorship of other government agencies.

Voluntary Product Standards—Developed under procedures published by the Department of Commerce in Part 10, Title 15, of the Code of Federal Regulations. The standards establish nationally recognized requirements for products, and provide all concerned interests with a basis for common understanding of the characteristics of the products. NIST administers this program as a supplement to the activities of the private sector standardizing organizations.

Consumer Information Series—Practical information, based on NIST research and experience, covering areas of interest to the consumer. Easily understandable language and illustrations provide useful background knowledge for shopping in today's technological marketplace.

Order the **above** *NIST publications from: Superintendent of Documents, Government Printing Office, Washington, DC 20402.*

Order the **following** *NIST publications—FIPS and NISTIRs—from the National Technical Information Service, Springfield, VA 22161.*

Federal Information Processing Standards Publications (FIPS PUB)—Publications in this series collectively constitute the Federal Information Processing Standards Register. The Register serves as the official source of information in the Federal Government regarding standards issued by NIST pursuant to the Federal Property and Administrative Services Act of 1949 as amended, Public Law 89-306 (79 Stat. 1127), and as implemented by Executive Order 11717 (38 FR 12315, dated May 11, 1973) and Part 6 of Title 15 CFR (Code of Federal Regulations).

NIST Interagency Reports (NISTIR)—A special series of interim or final reports on work performed by NIST for outside sponsors (both government and non-government). In general, initial distribution is handled by the sponsor; public distribution is by the National Technical Information Service, Springfield, VA 22161, in paper copy or microfiche form.

U.S. Department of Commerce
National Institute of Standards and Technology
(formerly National Bureau of Standards)
Gaithersburg, MD 20899