Report No. DOT-TSC-HW913-PM-79-6
February 1979

PREDICTION ALGORITHMS FOR URBAN TRAFFIC CONTROL

P. Mengert

P. Brown

L. Yuan

PROJECT MEMORANDUM

U.S. DEPARTMENT OF TRANSPORTATION

RESEARCH AND SPECIAL PROGRAMS ADMINISTRATION
Transportation Systems Center
Cambridge MA 02142

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

ABSTRACT

The objectives of this study are to 1) review and assess the state-of-the-art of prediction algorithms for urban traffic control in terms of their accuracy and application, and 2) determine the prediction accuracy obtainable by examining the performance of general time series analysis methods with actual data. This report is divided into two parts. Part I discusses the review and assessment, while Part II examines general time series analysis methods.

Accurate prediction algorithms are necessary for the effective operation of computerized traffic responsive control systems. These systems offer the potential for reducing traffic congestion and improving operational efficiency in the existing urban roadway system.

Although a number of prediction algorithms have been proposed and studied for urban traffic control, two algorithms are dominant: the Second Generation and the Third Generation predictors of the Urban Traffic Control System (UTCS). Both predictors are based on single-location traffic measurements. Both algorithms use the linear combination of residues (differences between traffic measurements and either historical data or smoothed traffic data) as the basic feature for prediction. The Second Generation predictor requires historical data as the reference. The Third Generation predictor does not require historical data and makes predictions based on current traffic measurements only.

In the review, (i.e., Part I of this report) test results showed that the predicted values of both the Second and Third

Generation Predictors tracked the trend of the actual values of the volume measurements, and both algorithms improved the prediction compared to using the current measurement as the predicted value. However, in both predictors the predicted values time-lagged the actual measurements. Also, the Second Generation Predictor worked consistently better than the Third Generation Predictor. This implies that the urban street traffic pattern does have considerable repeatability, so that the historical volume data is very desirable even for short-term traffic predictions.

In Part II of this report a general technique was developed for time series prediction and was applied to the problem of determining optimum predictors for traffic volume. The technique was developed for optimizing ARIMA (autoregressive integrated moving average - see Box and Jenkins Time Series Analysis and related predictors). Second and third generation UTCS predictors were within the general ARIMA framework and a systematic probe of a very general class of predictors tentatively showed the simple second and third generation predictor forms to be optimal within a large class including ARIMA and certain non-linear adaptive extensions. The best parameters to use are discussed. All empirical observations were based on actual traffic volume data collected on streets in Toronto.

PART I

REVIEW AND ASSESSMENT OF PREDICTION ALGORITHMS FOR URBAN TRAFFIC CONTROL

1.0 BACKGROUND

In general, there are two types of urban roadway control systems; namely, the street network control systems and the freeway control systems. The subjects which are studied here concentrate on street network control. The prediction algorithms in urban street network control are designed to asses's the short term variations of the traffic based on current and/or past traffic measurements, so that the available traffic data can be used as the basis for traffic control action determination. Essentially, the main intended purpose of the prediction algorithm is to compensate for the time lag between the traffic measurement and the traffic control action so that the effectiveness of the traffic responsive control actions can be fully realized.

In urban freeway control, incident detection algorithms are counterparts of the prediction algorithms. The main purpose of the incident detection algorithms is to reduce the time delay between the occurrence of an incident and the control action for removing the effect caused by the incident. Although incident detection was not the major subject under review, the concepts used in incident detection, such as traffic data smoothing and the correlation between traffic measurements and special events were quite relevant and useful in assessing the traffic prediction algorithms.

2.0 INTRODUCTION

In a real-time traffic-responsive urban traffic control system, the optimum signal timing is a function of the traffic in the network. The control actions are derived from the traffic measurements. In most real-time control systems, there is an inherent time lag between the system (traffic) measurement and the control action. This time lag prevents the real-time control algorithm from realizing its full effectiveness. Furthermore, time lags lead to potential oscillation - both in the control action itself and the resulting traffic flow. Methodologies which incorporate short-time traffic prediction into control algorithms therefore are very promising in the enhancement of the effectiveness of real-time control.

A functional representation of a typical real-time traffic control system is shown in Figure 1. In this system, the traffic data are gathered by the detectors on the road. These data are then processed to determine the traffic state (e.g., the traffic volume) of the system. The control commands are then generated based on the traffic state. These control commands are finally implemented (in terms of signal timing patterns) on the street network to regulate the traffic flow in a desired fashion. Starting from raw data gathering to control command implementation constitutes the traffic control loop.

In this traffic control loop, the time lag between data gathering and control command implementation stems from three places. First of all, time is needed to process and smooth the raw traffic data so that a meaningful traffic state can be obtained. Then, a computational time is needed to translate the traffic state into control commands (e.g. timing pattern). Finally, a so-called transition period is needed to make a smooth transition from the current signal timing pattern to the new timing pattern. This transition period is necessary to eliminate the undersirable disruptions (and thus potential traffic congestion) between timing pattern changes, and to comply with safety considerations (e.g., minimum amber time, minimum green interval). The time lag process from traffic measurement to new timing pattern implementation is represented schematically in Figure 2. This process usually takes 5 to 15 minutes. In other words, it takes at least 5 to 15 minutes for a new timing pattern to become effective. It is therefore desirable to predict traffic volume this far into the future so that the timing pattern to be used is based on the traffic volume which will exist at the time the resulting timing pattern is in effect. The mathematical procedures which are designed to make real-time, short-term traffic predictions based on current and/or previous traffic measurements are the "prediction algorithms".

2.1 MEASURES OF EFFECTIVENESS

The effectiveness of the prediction algorithms can be defined as the ability of compensating the time lag as mentioned before, and the ability of achieving prediction accuracy. In general, the time lag also shows up as the prediction error (i.e. the inaccuracy in prediction) and thus, in actual practice, the prediction error is usually used as the major measure of effectiveness of prediction algorithms.

For the purpose of numerical comparison, two aggregated measures are usually used to define the predictor effectiveness (Reference 1). They are the mean square error and the mean absolute error of prediction which are defined as follows:

mean square error

$$= \frac{1}{N} \Sigma \text{ (measured value - predicted value)}^2$$

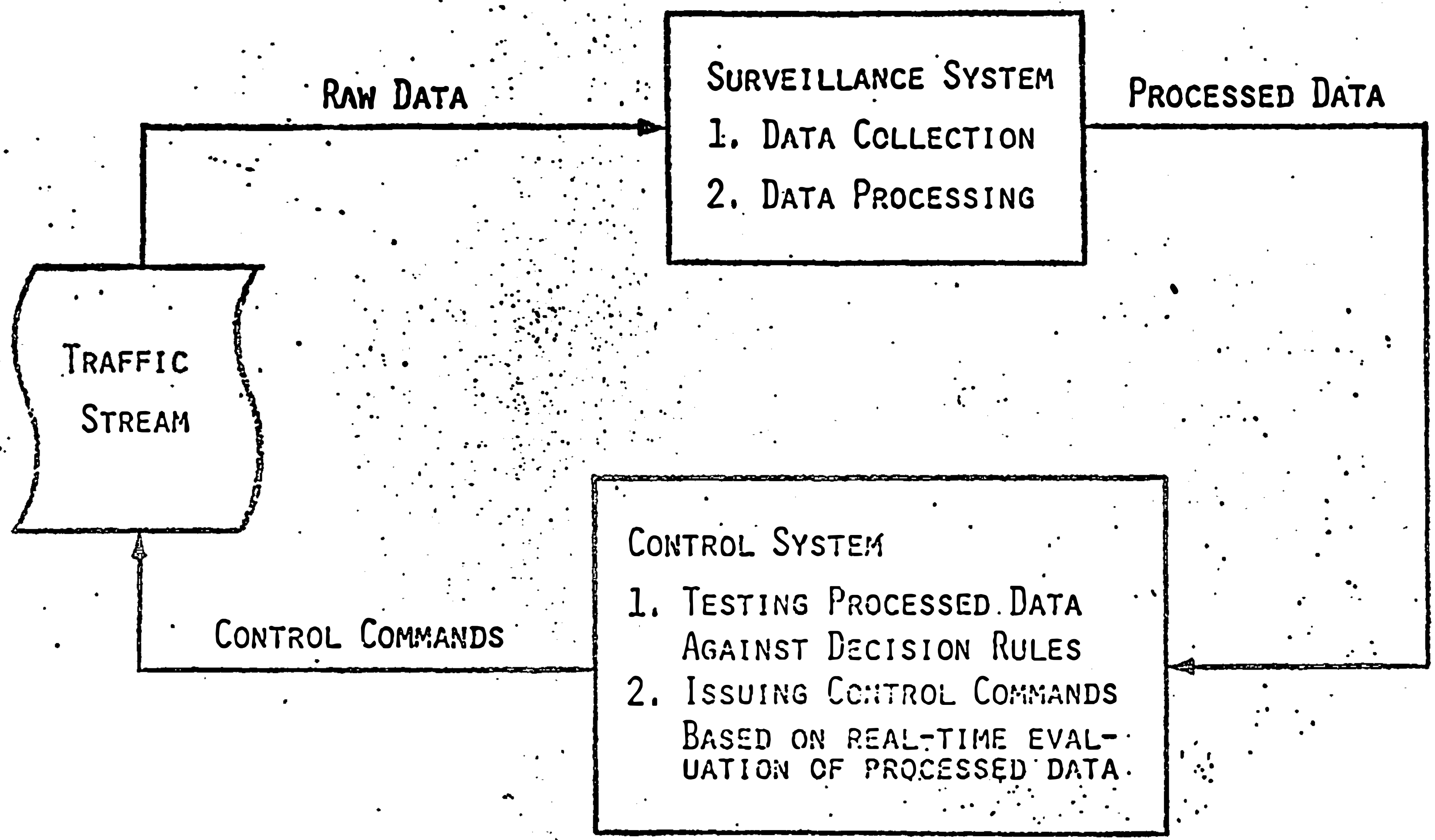

FIGURE 1. A TYPICAL TRAFFIC CONTROL SYSTEM

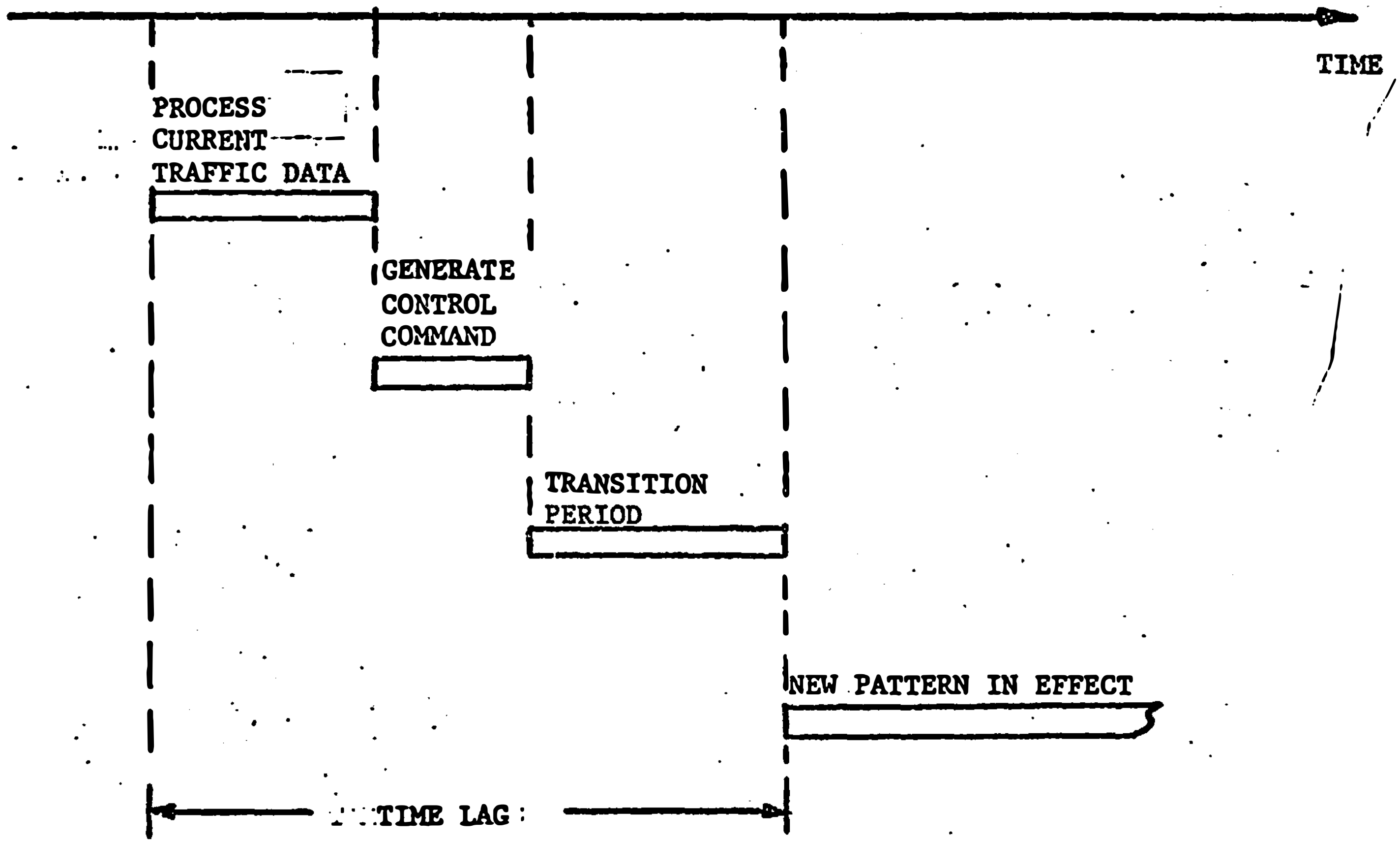

FIGURE 2. THE INHERENT TIME LAG

mean absolute error

$$= \frac{1}{N} \Sigma \; |\text{measured value} - \text{predicted value}|$$

where N is the total number of predictions made. The mean square error indicates the presence or absence of some frequent large errors in prediction, and the mean absolute error gives an idea of the error magnitude one might typically expect. In addition to the aggregated measures, prediction error distribution is another measure which is particularly useful in determining the bounds of the errors and the biases.

2.2 IMPACTS ON REAL-TIME TRAFFIC CONTROL

The usefulness of the predictor in the traffic responsive control system depends heavily on the sensitivity of the computed timing pattern with respect to the changes in traffic volume. If the traffic control systems were to use rather long time average volume as the basis for the timing pattern computation (i.e. the traffic fluctuations were not used in timing pattern generation so that short term variations were not considered), or if the traffic variations have only little effect on the timing pattern computation, the prediction algorithms and their prediction accuracy would have little value or consequence on traffic control. On the other hand, if the timing pattern generation of the traffic control system is highly responsive to the traffic variations, the prediction algorithms and their prediction accuracy would have a major impact on the computed signal timing pattern and thus the resulting traffic flow. Therefore, in actual applications, not only the predicting accuracy but also the associated impact of this accuracy on the real-time traffic control have to be considered.

3.0 CLASSIFICATION OF EXISTING ALGORITHMS

Prediction algorithms may be classified into the following categories:

- algorithms which require more than one measurement location versus those which require the traffic measurement from a single location,
- algorithms which require historical traffic data as a reference versus those which rely on current data only
- algorithms which use the linear combinations of traffic measurements for prediction, versus those based on non-linear combinations or parameters
- algorithms which are adaptive to the underlying process (e.g., the parameters of the algorithm vary with traffic deviations) versus those that are not.

3.1 MULTIPLE-LOCATION VERSUS SINGLE LOCATION MEASUREMENT

In an attempt to obtain more prediction accuracy, multiple location measurements may be used. The additional measurement locations may be upstream from the point of the prediction and/or on different traffic lanes. The practical implication of the multiple measurement location requirement is more detectors, communications, and processing. These additions will increase the cost of the traffic control system significantly. Previous studies (Ref. 2, 3, 4) showed that conflicting results can be obtained on the benefit to be derived from the additional measurements. More investigation in this area is thus needed in order to determine under what conditions and to what extent multiple location measurements will provide increased benefits over single location measurements. The advantage of multiple measurement locations can be examined using the methods developed in Part II of this report.

3.2 HISTORICAL DATA VERSUS CURRENT DATA ONLY

Another consideration in prediction algorithm classification is the historical data requirement. If the traffic flow has some regularity (or repeatability from day to day) the historical data should be very helpful in predicting future traffic flow even on the short term basis. This is because the next interval flow of the historical pattern is probably the best information for compensating the time lag between the predicted and the actual traffic flow. However, the implementation of a prediction algorithm using historical data would require a large data base. This historical data base would require several stratifications (i.e., by detectors, by week days, and by control intervals) and would thus be considerable. This historical data base not only would occupy large core space but also would require considerable effort to gather and to process before a traffic control system could be set up. In addition, since the historical data would be site specific, the prediction algorithm would not be readily transferable to other sites.

3.3 LINEAR VERSUS NON-LINEAR

All the existing algorithms with implementation experience which have been reviewed used the linear combination of residues for traffic prediction. The residue is defined as either the difference between actual measured traffic flow and the historical data, or the difference between the actual measured and the smoothed traffic flow data.

3.4 ADAPTIVE VERSUS NON-ADAPTIVE

One factor which determines a good predictor is its ability to model the underlying process of interest (e.g., the traffic flow in this case). If the characteristics of the process under prediction change with time (as most of the traffic flow does), the mathematical model of the process should also be changed in order to provide an accurate representation from which the predictions can be obtained. Making the model automatically adapative to the changing underlying process would therefore seem to be a desirable feature. This is discussed further in Sections 4.1.4 and 4.2.2 and examined analytically with actual data in Part II of this report.

3.5 SUPPLEMENTAL INPUTS TO THE PREDICTORS

The prediction algorithm classification delineation up to this point assumed that the traffic status was measured by detectors in the road. This assumption is a valid one because most existing traffic predictors used the detector measurement as the only input data. However, inputs from other sources may have profound effect on the performance of the predictor (e.g., the a-priori knowledge of special events affecting traffic such as ball games or adverse weather conditions).

An example of using supplemental information that could be beneficial in traffic prediction is the identification of specific vehicles. This information could be obtained from an automatic vehicle identifier (AVI). The AVI (under development) is a small passive device (either electronic or optical) which can be installed on the vehicle. The AVI can carry information of the vehicle such as the vehicle identification number. This information can be picked-up by special wayside detectors while the vehicle is passing by. The real-time information -- time, location, and identification -- is transmitted to the control center, through communication link, for processing.

The real-time AVI information would be very valuable in supplementing the traffic status obtained from the regular system detectors in predicting downstream future traffic demands and in computing the traffic control commands. AVI is an advanced concept that offers a potential for enhancing the effectiveness of conventional traffic control techniques.

4.0 PREDICTION ALGORITHM REVIEW AND ASSESSMENT

The prediction algorithms reviewed and described herein work exclusively with digital-computer-based traffic responsive control systems. This kind of control system is rather new. A major research project using digital computer in urban traffic control in the United States was initiated in May 1968 by the Bureau of Public Roads (the predecessor of the Federal Highway Administration) under the name of Urban Traffic Control Systems (UTCS) (Reference 5). The UTCS project provided the impetus for the evolution of advanced traffic control strategies (which includes the prediction methodologies). There are a number of prediction algorithms which have been proposed and studied for urban traffic control. However, the two most prominent algorithms are those which stemmed from the UTCS control software, namely, the Second Generation and the Third Generation Predictors. Both predictors are designed for single location measurements. Both algorithms use the linear combination of residues for prediction. The Second Generation Predictor requires historical data (or pattern of the data) as the reference. The Third Generation Predictor does not require historical data and makes predictions based on current traffic measurements only.

4.1 SECOND GENERATION UTCS PREDICTOR

The predictor for the Second Generation UTCS software predicts the next control interval (on the order of 5 to 15 minutes) traffic volume at each detector location in real-time based on the measurements from the same location only. The algorithm makes use of both smoothed historical traffic data and current traffic volume measurements from the vehicle detector. The rationale is that if the traffic volume of the day in question follows the average historical pattern, the historical pattern would give good predictions on volume changes in the near future. In addition, as a supplement, the current measurements are used by the algorithm to correct for the traffic deviations from the average historical pattern.

4.1.1 Predictor Formulation

Mathematically, the Second Generation Predictor is defined by the following sequence of computations. First, the residue (i.e., the difference) r(t) between the measured volume f(t) and the corresponding historical volume m(t) is computed as follows:

$$r(t) = f(t) - m(t) \tag{1}$$

This residue is then smoothed using an exponential filter. The predicted value of r(t) is defined as c(t) by the following smoothing process,

$$c(t) = \alpha c(t-1) + \beta r(t-1) \tag{2}$$

where

$$c(0) = 0, \text{ and}$$

$$\alpha + \beta = 1, \ \alpha \geq 0 \text{ and } \beta \geq 0$$

The difference, h(t), between the residue r(t) and the smoothed residue c(t) is computed as

$$h(t) = r(t) - c(t) \tag{3}$$

An empirical adjustment term d(t) is then computed as,

$$d(t) = \gamma h(t-1) \tag{4}$$

Finally, the predicted volume at time t is the sum of three terms,

$$v(t) = m(t) + c(t) - d(t) \tag{5}$$

(where, all the terms in the equations are defined in the text of this subsection).

In summary, the predicted volume is the average historical pattern modified by the predicted residue between current and historical traffic, and the result is further adjusted by a fraction of the difference between the same residue and the predicted one.

For a given location, there are two parameters α and γ to be determined. The parameter β is constrained by $\beta = 1 - \alpha$. The parameter α is the smoothing coefficient for the exponential smoothing process of the residues. For the UTCS system which was installed in Washington, D.C., the value of α was tuned to 0.9. The parameter γ is a constant obtained off-line. It is computed from representative volume data of the location in question by

$$\gamma = \frac{(n-1)\sum_{t=1}^{n-1} h(t)h(t-1)}{(n-2)\sum_{t=1}^{n} h^2(t)} \qquad (6)$$

where n is the number of data points of h(t) of the representative data set. For the UTCS system in Washington, D.C., 0.2 was used for γ.

Instead of storing the historical traffic pattern itself, for each measurement location, the Fourier series approximation of the historical traffic patterns are stored in the computer. This is done by finding the coefficients (a_0, a_i's and b_i's) of the following Fourier series.

$$m(t) = a_0 + \sum_{i=1}^{k} (a_i \cos 2\pi it/N + b_i \sin 2\pi it/N) \qquad (7)$$

which best fits the historical data for the location in question. This curve fitting process is done off-line using representative data for each location. In equation (7), parameter N is the total number of time intervals in the representative data set (e.g., if 15 minute intervals are used, the data for a 24-hour day will consist of 96 intervals), and parameter k is a user input parameter which determines the fidelity of the Fourier series approximation. It is usually the result of a tradeoff between the accuracy of the Fourier series for representing a time varying function and the storage space and computational effort one is willing to pay. In general, for more rapidly varying functions, higher values of k should be used. The upper bound k is N/2. Numbers from six to twenty have been used for k in past applications.

4.1.2 Discussions

The following subsections discuss the Second Generation UTCS Predictor. The discussions consist of some qualitative evaluations of the algorithm based on analytic reasoning, and some potential improvements to the algorithm. The discussions do not represent final recommendations for algorithm modification. The validity of the evaluation and the amount of improvements can only be verified numerically using real data.

4.1.2.1 Historical Data Storage Considerations

The historical data used in the Second Generation UTCS Predictor is approximated by a Fourier series. One of the reasons that the Fourier series was originally selected in the algorithm for historical data representation was probably for achieving memory space savings in the computer. Instead of storing the historical pattern itself, the Fourier series representation of the pattern in terms of a set of Fourier coefficients is stored in the memory. These coefficients generally occupy less memory space than that of the volume pattern and thus offer some memory space savings. However, in order to accurately represent the time fluctuations of the traffic volume pattern, high harmonics were needed in the series which result in a large number of coefficients. Thus, the memory space savings by using Fourier series approximation was not as much as was originally anticipated.

The Fourier series representation may have other undesirable characteristics in terms of computational requirement. Once the historical pattern is stored in Fourier series form, the real-time computations required for retrieving the historical pattern are an order of magnitude more than the remaining computations in the predictor algorithm. Even though sine and cosine computations may be unified for all detectors, for each location of interest and for each control interval, there are many multiplications and sine and cosine computations involved to compute the historical volume m(t). Each of the many measurement locations in the system requires a prediction and the retrieval of the historical pattern. The involved computation that is required in the retrieval process using Fourier series represents a considerable added burden to computational requirement.

Furthermore, it is well established that traffic control and traffic pattern changes are mutually interactive. Once a timing pattern (i.e., of the traffic control signals) change is made in the traffic control system, the traffic pattern (e.g., traffic volume profile) will change accordingly as a result of the new signal timing pattern. Therefore, the historical traffic pattern should be updated for signal timing pattern optimization. If the Fourier series is used for traffic pattern representation, real-time updating of historical traffic pattern means real-time updating of the Fourier coefficients. However, as previously mentioned the Fourier coefficients estimation process is also time (CPU time) consuming, and perhaps for real-time operation. Therefore, for real-time traffic control system design, careful tradeoff between memory space savings and computational time savings should be done before reaching the final decision on the type of prediction algorithm to be implemented.

Another consideration in historical pattern representation is the filtering of historical data. As any curve fitting process, Fourier series representation has the effect of filtering out the interval-to-interval fluctuations of the historical traffic pattern. However, it is uncertain whether this kind of filtering is necessary or advantageous. Since the historical pattern in the Second Generation Predictor is designed for preserving the traffic pattern for prediction, the fluctuations may be important in short term predictions. Furthermore, the historical pattern itself is the result of a filtering process. Each point on the historical pattern is an average of the data over many days, and thus, the day-to-day variations have been filtered out.

4.1.2.2 Prediction Structure:

By some mathematical manipulations, the basic structure of the Second Generation UTCS Predictor can be deduced. From equation (2), by successive substitutions, the variable c(t) can be written as follows:

$$c(t) = \alpha^K c(t-K) + (1-\alpha)\sum_{S=0}^{K-1} \alpha^{K-S-1} r(t-K+S) \qquad (8)$$

The constant K is the number of intervals that the predictor has been in operation since the latest start of the system (e.g., the starting time of a working day). Again from equation (2):

$$c(t-1) = \frac{1}{\alpha}[c(t)-(1-\alpha)r(t-1)]$$

With simple manipulations, it can be proved that

$$c(t-1) = \alpha^{K-1} c(t-K) + (1-\alpha)\sum_{S=0}^{K-2} \alpha^{K-S-2} r(t-K+S) \qquad (9)$$

Define $\delta(t)$ as the deviation (difference) between the predicted and the historical volumes at time t, then,

$$\begin{aligned}\delta(t) &= v(t) - m(t)\\ &= c(t) - d(t)\\ &= c(t) - \gamma h(t-1)\\ &= c(t) - \gamma[r(t-1) - c(t-1)] \qquad (10)\end{aligned}$$

Substituting equations (8) and (9) into equation (10), one has,

$$\begin{aligned}\delta(t) &= \alpha^{K-1}(\alpha+\gamma)c(t-k)\\ &+ (1-\alpha)(\alpha+\gamma)\sum_{S=0}^{K-2} \alpha^{K-S-2} r(t-K+S) \qquad (11)\\ &+ (1-\alpha-\gamma)r(t-1)\end{aligned}$$

If the predictor has been initiated for a considerable time (i.e., for a large K), because α is less than one, the first term of equation (11), $\alpha^{K-1}(\alpha-\gamma)c(t-K)$, is negligible. Thus, $\delta(t)$ can be approximated as,

$$\delta(t) = (1-\alpha)(\alpha+\gamma)\sum_{S=0}^{K-2} \alpha^{K-S-2} r(t-K+S) + (1-\alpha-\gamma)r(t-1) \tag{12}$$

Note that, since α and γ are constants, the right hand side of equation (12) is a linear combination of all the previous residues r(t-1), r(t-2),..., r(t-K) between measured and the historical volumes. In other words, the prediction on volume deviations, by the Second Generation Predictor, are based on only the linear combination of previous residues between measured and historical volumes. Whether the linear combination of residues is the best methods for volume predictions can only be determined numerically using actual field data.

4.1.2.3 Implementing an Adaptive Feature:

From equation (5) - the basic equation for the Second Generation Predictor - it is seen that, since m(t) is the known historical volume and d(t) is an empirical correction term, predicting v(t) is essentially the same as the determination of c(t). From equation (2), c(t) is determined by an exponential smoothing process. If the smoothing parameter α is made adaptive, better smoothing can be obtained such that the output of the adaptive smoothing filter tracks the underlying process more closely.

In other words, an improvement may be obtained from the Second Generation Predictor by simply making the parameter α in equation (2) adaptive. That is,

$$c(t) = \alpha(t-1)c(t-1) + [1-\alpha(t-1)]r(t-1) \tag{13}$$

It is emphasized that this improvement is desirable, because the Second Generation UTCS Predictor has been showing the best performance so far.

The Trigg and Leach method (Reference 6) could be used to determine the parameter $\alpha(t)$. With this method

$$\alpha(t) = f(1 - \frac{Q(t)}{\Delta(t)}) \tag{14}$$

where f is a constant and $0 < f < 1$, $Q(t)$ is the smoothed forecast error and $\Delta(t)$ is the smoothed mean absolute deviation, both computed at the end of period t. The smoothed error is computed according to

$$Q(t) = \phi Q(t-1) + (1-\phi)h(t) \tag{15}$$

$$\Delta(t) = \phi\Delta(t-1) + (1-\phi)|h(t)| \tag{16}$$

where $h(t) = r(t) - c(t)$ is an error term as defined in the previous Section and ϕ is a smoothing constant such that $0 < \phi < 1$.

The term $\frac{Q(t)}{\Delta(t)}$ is called the smoothed error tracking signal. It is noted that, from equations (15) and (16), the following condition,

$$0 \leq |Q(t)| \leq \Delta(t)$$

is always true, and thus the smoothed error tracking signal always lies in the interval (-1, +1).

If the forecasting system as defined in equation (13) is performing adequately, the value of the smoothed forecast error Q(t) will fluctuate between positive and negative values around zero (while $\Delta(t)$ is always greater than zero and greater than $|Q(t)|$). As a consequence, the value of smoothed tracking signal will be small, near zero, and the forecasting system in "in control." If the underlying form of the time series r(t) changes, the forecasting system will eventually begin to generate large errors and the tracking signal will move towards either plus or minus unity. That is, the forecasting system is "out of control." The Trigg and Leach method decreases the smoothing constant $\alpha(t)$ when the tracking signal indicates an out-of-control condition thus giving more weight to the recent data and allowing the system to more rapidly track the new signal. When the system has stabilized, however, the value of the smoothing constant is returned to its normal value (i.e., f) automatically.

It is noted that the Trigg and Leach method is not proposed as a stand alone algorithm for traffic prediction. Rather, it is a potential improvement on both the Second and the Third Generation UTCS Prediction algorithms.

4.2 THIRD GENERATION UTCS PREDICTOR

The predictor for the Third Generation UTCS software predicts the traffic volume two control intervals into the future. (Two control intervals of lead time are required by the Third Generation UTCS software.) Like the Second Generation predictor, the

Generation Predictor also predicts the volume at each location in real time based on measurements from the same location. However, it is different from the Second Generation Predictor in that the prediction process relies solely on current-day measurements (no historical traffic pattern is required for prediction). The Third Generation Predictor operates as follows.

For each location of interest, the vehicle detector measurements are exponentially smoothed. The predicted value of the traffic volume at that location is the sum of the most recent smoothed value and the residue (i.e., the difference between smoothed and un-smoothed values) which is extrapolated to the prediction time. (Refer to equations (19) and (20) as to how the extrapolation was done.) The extrapolating coefficient is determined from representative data for the location in question and also is a function of how far in the future the prediction is to be made.

The rationale for the Third Generation Predictor was to develop a methodology that did not depend on historical data. The development of such a predictor was considered desirable for the following reasons[3]:

- A large data base is required for the historical data. This data base consumes computer storage space and must be updated periodically off-line.
- Traffic volume can vary substantially depending on various external (with respect to the algorithm) factors (e.g., weather conditions, special events, developments

in other modes of transportation, and even the traffic control change itself).

- o An analysis conducted early in the UTCS project using "simulated" traffic data indicated that utilizing historical data was not always necessary to achieve good prediction.
- o The predictor algorithm would be more practical due to its transferability to other systems.
- o There is a general concensus amongst traffic engineers that a highly responsive control software (the Third Generation control software) should have a predictor that did not rely on historical data.

4.2.1 Predictor Formulation

Mathematically, the Third Generation Predictor is defined by the following sequence of computations. First, the current day traffic measurements are smoothed to obtain the up-to-the-moment trend of the traffic volume.

$$\hat{u}_k(i) = \beta\hat{u}_k(i-1) + (1-\beta)v_k(i) \qquad (17)$$

where $v_k(i)$ = traffic measurement at station k (i.e., the traffic detector) at current time interval i,

$\hat{u}_k(i)$ = the exponentially smoothed measurement at station k up to time interval i,

β = smoothing constant.

The residue $y_k(i)$ is next computed as the difference between the measured and the smoothed values.

$$y_k(i) = v_k(i) - \hat{\mu}_k(i) \tag{18}$$

An extrapolation is used, in the original Third Generation UTCS Predictor, to predict the estimated residue j intervals in the future. (Recall that the residue is defined as the difference between smoothed and un-smoothed traffic measurements),

$$\hat{y}_k(i+j) = \alpha_j y_k(i) \tag{19}$$

where α_j is the extrapolation coefficient. This coefficient is obtained off-line, using a set of "representative data", by the following equation:

$$\alpha_j = \frac{(N-1)\sum_{S=1}^{N-j} y_k(S)y_k(S+j)}{(N-1-j)\sum_{S=1}^{N} y_k(S)y_k(S)} \tag{20}$$

where N is the number of sample points of the representative data.

An extrapolation is also performed on $\hat{\mu}_k(i)$ to estimate the smoothed volume j intervals in the future as follows:

$$\hat{\mu}_k(i+j) = a_1 + a_2 j \tag{21}$$

The variables a_1 and a_2 are computed in either one of the following two ways:

(1) The variables a_1 and a_2 are determined such that the forecast $\hat{\mu}_k(i+j)$ is the linear extrapolation of the last two values. That is

$$a_1 = \hat{\mu}_k(i-1), \text{ and}$$

$$a_2 = [\hat{\mu}_k(i)-\hat{\mu}_k(i-1)], \text{ or}$$

(2) The $\hat{\mu}_k(i+j)$ is selected as the last value. That is,

$$a_1 = \hat{\mu}_k(i), \text{ and}$$

$$a_2 = o$$

In the final software which was installed in Washington, D.C., the second method was used to determine $\hat{\mu}_k(i+j)$, that is

$$\hat{\mu}_k(i+j) = \hat{\mu}_k(i) \tag{22}$$

Finally, the predicted volume j intervals in the future (i.e., the quantity of interest) is determined as,

$$\hat{v}_k(i+j) = \hat{y}_k(i+j)+\hat{\mu}_k(i+j) \tag{23}$$

The component $\hat{\mu}_k(i+j)$ can be interpreted as the smoothed coarse prediction of the traffic j intervals in the future, and the $\hat{y}_k(i+j)$ term as a fine adjustment of the prediction to account for the residues.

This form of the third generation predictor is discussed further in Part II of this report where the connection with optimal "ARIMA" predictors is noted.

4.2.2 Discussion

In the Third Generation Predictor, the prediction is done by extrapolation. The extrapolation coefficient α_j is obtained off-line using "representative" data. The implicit assumption is that there is a norm of the traffic pattern which can be reproduced according to a "representative" data set. It is interesting to note that this assumption is in conflict with the idea of "highly responsive control software" (i.e., the Third Generation control software) for which the predictor was designed. Aside from this conflict, it is still reasonable to ask--to what degree is the assumption valid? In other words, how much benefit (in terms of algorithm performance) could be obtained if the process for computing α_j were made in real-time (i.e., computing α_j using current data)?

The on-line computation of α_j could be done by a simple modification to equation (20) as follows:

$$\alpha_j = \frac{(N-1)\sum_{S=1-N}^{-j} y_k(s)y_k(S+j)}{(N-j-1)\sum_{S=1-N}^{0} y_k(S)\ y_k(S)} \qquad (24)$$

where N is the number of the latest residue data to be used for the computation, j is the number of intervals in the future which the predictions will be made, and the current time interval is denoted as interval 0. Note that the coefficient α_j is now a time varying function which is adaptive to the latest trend of the traffic deviations.

Compared to the original Third Generation Predictor, additional real-time computation is needed for parameter α_j. In addition, more storage space is needed to save the N latest residues for each location. However, the number N can be constrained by the user of the algorithm and it is usually a small number, which puts a bound on both the storage and CPU time expenditures. Furthermore, a recursion formula can be derived to replace equation (24) which will make the α_j computation very efficient.

In the Third Generation Predictor, exponential smoothing is used to filter the volume measurements (equation 17). The smoothing constant β can be made adaptive by the Trigg and Leach method. (See Section 4.1.2.3) With this modification on parameter β and the on-line computation of α_j (equation 24), the new Third Generation Predictor would be fully adaptive. The application of the Trigg and Leach method to the Third Generation Predictor is evaluated with actual data in Part II of this report.

4.3 GENERAL ASSESSMENT ON THE UTCS PREDICTOR

Based on the literature, the review and assessment identified two basic types of predictors for traffic control, namely, the Second and Third Generation UTCS predictors. These predictors are at the forefront of prediction methodology for urban traffic control. During the development of advanced traffic control strategies for the UTCS Project, various prediction algorithms were proposed and studied, but later dropped from consideration during the evaluation phase. These algorithms are not included in the Report.

The major difference between the Second and the Third Generation Predictors is that in the Third Generation Predictor, the historical data is not required for prediction (although some off-line determination of α_j is required). This difference represents a considerable computer storage savings for the Third Generation Predictor. This difference also represents a considerable savings in the effort involved in the collection and preparation of historical traffic data in the initial system set-up stage. Since the Fourier series representation of historical data is used in the Second Generation Predictor, eliminating the historical data requirement also represents a considerable savings in CPU time during system operation.

It is noted that for the system which is warranted for advanced traffic control, hundreds of system sensors (i.e. traffic detectors) will be required (e.g., even the experimental UTCS system in Washington, D.C. requires that many sensors). If the Second Generation Predictor is used, for each sensor and each time interval the Fourier series computations will be required. The real-time computation involved may not be a small amount. Also, because of the real-time considerations, for each sensor, the whole day's historical traffic pattern has to be stored in core, so that the roll-in and roll-out times can be eliminated for speedy real-time processing. (The historical traffic data for other days of the week is stored on disk). This represents a principal portion of the core requirement of the traffic control strategy.

In terms of the effectiveness of the predictors, based on the limited test results available, (Refs. 1, 2), the Second Generation Predictor worked consistently better than the Third Generation Predictor. This implies that the urban street traffic pattern has considerable repeatability, so that the historical volume data is very desirable even for short-term traffic predictions. Therefore, a tradeoff between prediction accuracy and the historical data requirement, or some simplification in data storage and retrieval is in order.

The test results also showed that the predicted values of both the Second and Third Generation Predictors track the trend of the actual values of the volume measurements, and both algorithms made some improvements in prediction compared to using the current measurement as the predicted value. However, both predictors had the problem of time-lagging between predicted values and the actual measurements. This time-lag was especially obvious with the Third Generation Predictor. Theoretically, historical patterns used in the Second Generation Predictor should help to resolve the time-lagging problem. However, because of the stochastic nature of the traffic the time-lag was not fully compensated by the Second Generation Predictor.

There are similarities between the Second and Third Generation Predictors. Both predictors use the residues between the measured and smoothed volumes (See Section 4.1.1 and 4.2.1) for predictions.

Another common factor for the Second and Third Generation algorithms is that there is currently no on-line updating capability of the historical patterns and the parameters. However, as explained in previous sections, on-line updating and real-time adaptive parameter changes are worth considering. It is noted that an off-line procedure for historical pattern updating had been tried for the Second Generation Predictor in the UTCS Project. This was done at the end of the day when the traffic control system was inoperative. The traffic data which was recorded during the day was processed and the historical traffic pattern updated. This procedure represents a possible added storage requirement for the current-day traffic pattern, and extra effort of the analyst.

Exponential smoothing was used in both the Second and Third Generation Predictors. In the Third Generation Predictor the exponential smoothing was used to determine the up-to-the-moment estimate of traffic volume. In the Second Generation Predictor the exponential smoothing was used as part of the prediction. The exponential smoothing process is very efficient in real-time processing and consumes minimal core storage. It thus seems to be an effective method.

In summary, while a limited amount of data is available showing that the UTCS predictors are capable of predicting the short term traffic fluctuations, these predictors still may not represent the ultimate in prediction capability. In an attempt to identify improved or promising techniques, Part II of this report continues the investigation of prediction algorithms for urban traffic control. More general time series analysis methods are developed and their performance studied with actual data. The Second and Third Generation predictors are shown to be special cases of the general methods examined.

References

1. John B. Kreer, "A Comparison of Predictor Algorithms for Computerized Traffic Control Systems," Traffic Engineering, April 1975.
2. E. B. Lieberman, et al, "Variable Cycle Signal Timing Program, Volume 4. Prediction Algorithm, Software and Hardware Requirements, and Logical Flow Diagrams," Prepared for Federal Highway Administration, May 1974, PB-241 720.
3. William R. McShane, "Developing a Predictor for Highly Responsive System-Based Traffic Signal Control," Presented at the 55th Annual Meeting of the Transportation Research Board, January 1976, Washington, D.C.
4. "A Traffic Volume Predictor for Networks," Prepared for the Federal Highway Administration by Integrated Systems, Inc., 11141 Georgia Avenue, Wheaton, Maryland, April 1972.
5. "Prospectus, Advanced Control Technology in Urban Traffic Control System-Installation," Prepared by Traffic Systems Division, Office of Research and Development, Bureau of Public Roads, December 1969.
6. Trigg, D. W., and A. G. Leach, "Exponential Smoothing With an Adaptive Response Rate," Operational Research Quarterly, Vol. 18, no. 1, p.p. 53-59, 1967.

PART II

EXAMINATION OF GENERAL TIME SERIES ANALYSIS METHODS

1.0 INTRODUCTION

Traffic prediction algorithms have to date achieved a success which seems neither entirely satisfactory nor so poor that we expect to easily find improvements. Improvement is, however, needed and it is desirable to know how accurately a traffic volume predictor, such as the Second and Third Generation UTCS predictors can perform.

Past efforts to find better predictors have explored various suggestions; it is appropriate at this time to attempt a systematic approach for finding an optimum traffic predictor.

Although it may not, in general, be possible to sharply delimit the ultimate obtainable in a prediction problem, the techniques which we have developed and used in this study, including optimization of the class of ARIMA predictors given by Box and Jenkins and of an extension of this class to probe the usefulness of non-linear or adaptive variations, provide a step in that direction.

The results, although obtained on a sample representing a limited data base, suggest what may be expected in a more general case.

2.0 BOX-JENKINS (Ref 1)

2.1 GENERALITY:

The Box-Jenkins approach to time series prediction appears to be the most comprehensive approach to general time series prediction. It encompasses in effect all possible linear single-channel predictors. The term "single-channel" refers to the fact that the previous values of the time series being predicted are the only real-time quantities used in making the prediction. Although other techniques are available which are non-linear and, thus, not covered by the Box-Jenkins approach, they tend to be ad hoc procedures not tied together in a coherent program for time series analysis. As a result, the Box Jenkins approach may be called "state-of-the-art" for times series prediction and any attempt to define the limits of predictive power achievable in almost any context should contain a Box-Jenkins analysis or an evaluation of the applicable Box Jenkins predictors.

As to the generality and acceptance of the Box-Jenkins methodology, we quote the first sentence of each paragraph in Section 9-5 of Forecasting and Time Series Analysis by D.C. Montgomery and L.A. Johnson (Ref 2) which section is entitled, "A Critique of the Box-Jenkins Models": "The Box-Jenkins methodology is a powerful approach to the solution of many forecasting problems...not without several important limitations...[1] In general we require at least 50 and preferably 100 observations to develop an acceptable Box-Jenkins model....[2] Another disadvantage of the Box-Jenkins models is that there is not, at present, a convenient way to modify or update the estimates of the model parameters as each new observation becomes available, such as there is in direct smoothing....[3] A final drawback of the Box-Jenkins models is the investment in time and other resources to build a satisfactory model....Despite these shortcomings, the Box-Jenkins models are probably the most accurate class of forecasting models available today..." (The bracketed numerals have been inserted.)

From the point of view of this study, the above quotes amount almost to an unqualified recommendation of Box-Jenkins. The possible shortcoming [2] identified by Montgomery and Johnson and alluded to in Part I as applying to the second and third generation predictors (which, as we shall later show, are special cases of Box-Jenkins predictors) is lack of adaptation. Adaptation amounts to a non-linearity. Certain aspects of adaptive predictors were discussed in Part I and the subject will be addressed below. We shall address the question of how well Box-Jenkins predictors work when augmented with adaptive terms.

There can be no general class of time series predictors, and no analysis of predictors for specific time series (in this case traffic volume counts) can determine the best predictors or the ultimate performance obtainable, since there are always other predictors that can be added to any class of predictors and the optimum may be outside of the class examined. The "augmented Box-Jenkins probe" will, however, represent a limited approach to this ideal.

Further, we shall show the form which the Box-Jenkins predictors take. As we have noted, they form a very general class whose essential restriction is that it consists only of linear predictors. A linear predictor is of the form:

$$\hat{Z}_t(\ell) = \sum_{k=o}^{\infty} \alpha_k Z_{t-k}$$

where $\hat{Z}_t(\ell)$ is the predicted value of $Z_{t+\ell}$ at time t. Thus $\hat{Z}_t(\ell)$ is called a predictor of "lagℓ". If $\ell = 1$, we have a "1-step predictor"; if $\ell = 2$, a "2-step predictor". A predictor of lag ℓ, i.e. an ℓ-step predictor, predicts the value, $Z_{t+\ell}$, of the time series ℓ steps <u>after</u> the latest available value, Z_t. When $\ell = 1$ we use the simplified notation, $\hat{Z}_{t+1}$, for the 1-step predictor $\hat{Z}_t(1)$. As the equation shows, $\hat{Z}_t(\ell)$ (the predicted value) is formed as a linear combination of Z_t, Z_{t-1}, Z_{t-2}...It is assumed that the coefficients α_k drop off rapidly as the value of Z_{t-k} which they multiply are further removed from Z_t (i.e. as k gets larger). The generality and comprehensiveness of the class of linear predictors is much greater than one would initially expect. It might be pointed out that an <u>analytic</u> function is determined by all its derivatives at a point and, therefore, <u>all future values</u> would be <u>determined</u> by a (convergent) <u>linear combination</u> of past values.

2.2 SECOND GENERATION UTCS AND THIRD GENERATION UTCS PREDICTORS AS BOX-JENKINS PREDICTORS

The second and third generation predictors of UTCS are linear predictors, as may be readily verified. They are also predictors of the Box-Jenkins (B-J) type. By this we mean that they are of the class of B-J predictors for the so-called ARIMA processes or models.

ARIMA(p,d,q) stands for the general class of models for time series considered by B-J. The values of p, d, and q (all small integers) characterize a classification of these models. The ARIMA(p,d,q) model is produced by passing (discrete) white noise through a (discrete) linear filter with p poles and q zeroes and subjecting the result to d summations, leading to the d'th order non-stationarity. (See Appendix A for more details.) A particularly simple and useful non-stationary model is obtained when p=q=d=1 giving the ARIMA(1,1,1) model. In Section 5.4.6 (Ref 1) Box and Jenkins show the explicit form of the ARIMA(1,1,1) predictor and it is easily seen to be equivalent to the UTCS third generation predictor (Part I Sec. 4.2). In Appendix F, we provide the algebraic details showing the equivalence. Reading Appendix F should serve to fix the notions of certain aspects of notation and the formalism of ARIMA filters especially as they relate to UTCS predictors.

The UTCS second generation predictor (Part I, Sec. 4.1) operates on the difference between the present traffic count at a sensor and the average of past counts for the same time of day at the same sensor. It operates on this difference in precisely the same manner as the third generation predictor acts on the traffic count itself. In this important respect the second generation predictor also falls in the realm of ARIMA(1,1,1) predictors. The details of this correspondence are shown in Appendix F.

If the historical average is determined as an exponentially smoothed moving average of past counts then, it turns out as shown in Appendix F that the second generation predictor is a predictor for a seasonal multiplicative model [ARIMA(1,1,1) X $(0,1,1)_{288}$] in the classification of Box and Jenkins. (The subscript 288 results from the fact that there are 288 (5 min.) time periods in one day).

The historical average that is used for second generation predictors and more general predictors based on historical data can be determined in a number of ways. In Appendix E the merits of various orders of Fourier smoothing of historical averages are investigated. It is concluded that the most accurate predictions are based on very high orders of Fourier smoothing or on a straight (unsmoothed) historical averages. As a consequence all the historical predictors that are reported on outside of Appendix E are based on straight (unsmoothed) historical averages. See Appendix E for more details.

As the best current predictors already fall into the class of ARIMA predictors this enables us to work outwards from this starting point in an attempt to find still better predictors for the traffic problem. Our general probe, whether it be of the general B-J ARIMA predictors or of the boundaries of this class in order to establish its sufficiency, will have the second and third generation predictors as a starting point.

2.3 THE APPROACH TO OPTIMAL BOX-JENKINS PREDICTORS AND VARIATIONS THEREON

Our approach to finding optimum or best predictors will be based, as we have noted, on the Box-Jenkins class of predictors as derived for the ARIMA processes. However, we use a somewhat different procedure for finding the best predictor in this class from the one proposed by Box and Jenkins in their book. Their approach is based on identifying and calibrating an ARIMA model and then using the theoretically derived optimum predictor for this model. Our approach is based on optimizing the coefficients (or parameters) of the predictor of the form of the optimum predictor for any given classification of ARIMA process. The classification in terms of p, d, and q, the three small integers which determine the general ARIMA predictor form, is determined by optimizing ARIMA predictors of more than one form.

How to set up a predictor of the proper form and how to optimize its coefficients are discussed in Appendix A. Our procedure optimizes the parameters of an arbitrary ARIMA predictor. The optimization is of the performance of the predictor, as measured by mean square error, on a sample of actual traffic count data* (see Appendix D for discussion of data).

The technique employs an efficient non-linear iterative technique for finding a least mean square error predictor of a specified ARIMA type (using a given sample data set of the time series to be predicted) to determine the optimum coefficients. The development in Appendix A is an important part of this study and is recommended for the reader with mathematical interest in the technique. The mathematical development is put in Appendix A in order to make the rest of the report more accessible to readers with less mathematical interests.

---

*The data was kindly furnished by Prof. John B. Kreer of Michigan State University. Prof. Kreer used this data in a previous study of traffic predictor performance. (Ref. 3)

Incidentally the B-J "optimum" ARIMA predictor is optimum only in the sense that it would give a least mean square error for an actual ARIMA process of the given type. Our "optimized" predictor is of the same form but with coefficients calculated to give a least mean square error on a specific data set. Once one takes into account the method by which B-J estimates the parameters of the model from which the optimum predictor is derived, the two methods would generally seem to be rather close in results. If they were not close, it would seen that "optimized" predictors would be preferable to "optimum" predictors if calculated on large data sets since any discrepancy would presumably be due to model failure.

If one has a very long segment of sample data as we do here, the process of model identification or, equivalently, predictor form identification is achieved by optimizing or fitting a few ARIMA predictors, i.e. the predictors corresponding to a few selected values of p, d, and q.

In general we have two criteria for choosing a predictor—simplicity and performance. By the simplest we mean the predictor with the fewest parameters (which means in turn, p, d, q, each as small as possible) - the principle of parsimony. This conflicts with the principle of best performance in that adding more parameters can always improve the performance of the optimized predictor. The resolution is simply to choose the predictor with the fewest coefficients which does not perform significantly worse than the highest order (or least simple) predictor optimized.

The question remains to determine when we have tested the highest order predictor that need be tested. A traditional regression analysis criterion is to stop increasing the complexity when the improvement achieved becomes insignificant. As an ARIMA(p,d,q) predictor can be made more complex by increasing both p and q, our criterion will simply be that an ARIMA(p,d,q) predictor has enough complexity when an optimal ARIMA (p+1,d,q+1) predictor does not perform significantly better. This also tests for d being increased by one since the ARIMA (p+1,d,q+1) form includes ARIMA(p,d+1,q+1) and ARIMA(p,d+1,q) as special cases.

Note that by adding two parameters at once we are giving a very liberal opportunity for increased complexity to manifest an improvement as the usual procedure is to add only one additional parameter at a time and to investigate whether that parameter produces a significant improvement all by itself. We test two added parameters at once since we have discovered that (for traffic prediction on this data) they do not produce a significant improvement over ARIMA(1,1,1). Thus, this fact may as well be established with one test which is at the same time quicker to perform and more comprehensive.

The relation between the present analysis and a traditional Box-Jenkins analysis bears further elucidation.

Our analysis is based on the techniques outlined by B-J in their book (Ref. 1). Our intent has been to simplify the procedure (for our purposes) with no appreciable sacrifice in validity of results.

Our first observation is that the B-J predictors are of a form suitable for direct optimization. A technique for optimizing such predictors is presented in Appendix A. The technique is not wholly unlike that given by B-J for finding an optimal estimate of the parameters for a fixed ARIMA process.

The chief difference between our analysis and that of B-J seems to be not in the optimization (or fitting) technique but in the method of finding the best model form--i.e. in identifying the best values for p, d and q. We have short circuited the "model identification" stage of B-J which would be lengthy and somewhat inconclusive in order to jump immediately to a "base" model (in this case ARIMA(1,1,1)) and then test its adequacy by "overfitting." This whole procedure might well be considered to fit within the program of model development as presented by B-J in Ref. 1--certainly they do not expect every technique and procedure they develop or present to be exercised in each application. Indeed B-J seem to emphasize model estimation and diagnostic checking as the key elements ensuring model adequacy. For us "model estimation" equals "optimization" while our approach to diagnostic checking is overfitting.

Overfitting is a primary method of diagnostic checking in the program of B-J. Sample quotes from chapter 8 (Ref 1) on "Model Diagnostic Checks": "No system of diagnostic checks can ever be comprehensive, since it is always possible that characteristics in the data of an unexpected kind could be overlooked." Also see the last paragraph on p. 285 of Ref 1 (not reproduced here because of its length.) It should be added parenthetically that B-J are always concerned only with various ARIMA models (those of higher order, in general), when they express possible concerns about model inadequacy. They recommend overfitting as a means of seeing if a model is inadequate. Overfitting, of course, means fitting an ARIMA model of higher order than has been expected to be sufficient.

In short the mean square error criterion of the optimized model is the ultimate basis for choosing the model. As optimizing models is relatively simple, our technique of examining what would amount to a gross overfit of an ARIMA(1,1,1) vs. an ARIMA(2,1,2) and then showing that that overfitted model is negligibly better in measured performance might be expected to be an adequate way to establish the simpler ARIMA(1,1,1) for the data at hand.

This is analogous to the assumption that 5th and higher degree terms of polynomial fit may be ignored if a second degree polynomial is fit to some points and it is found that third and fourth degree terms have negligible effect on the sum of the square error. We use an example of two extra terms proven negligible in order to correspond to the two extra parameters in ARIMA(2,1,2). But the principle is the same as if only one extra term were tested, (i.e., to infer the absence of higher order complexity) because of the proven absence of one or more intervening orders of complexity. One certainly cannot prove, for example, that the absence of third and fourth degree terms in a least squares fit to some point precludes with absolute certainty a significant fifth degree fit. Nor can we even demonstrate that the fifth degree term is unlikely to be appreciable (if the third and fourth degree terms in the fit are negligible) except with a-priori assumptions, the exact nature of which would be interesting to develop but would carry us outside our practical scope.

A good discussion of the mathematical implications of the procedure seems to be lacking in the literature in spite of its place in common practice and its intuitive reasonableness. Natrella's book Experimental Statistics (Ref. 7) (G.P.O.), noteworthy for its emphasis on specifying sound accepted practice (but containing very little explanation of the underlying theory), may be quoted on this matter (the general problem of choosing the degree for a least squares polynomial fit): (p. 6-19, Ref. 7) "In using a polynomial as an approximation to some unknown function, or as an interpolation formula, the correct degree for the polynomial is usually not known. The following procedure usually is applied:

(a) Carry through the steps in fitting polynomials of 2nd, 3rd, 4th, 5th...degrees.

(b) If the reduction in the error sum of squares due to fitting the kth degree is statistically significant on the basis of the F - test whereas the similar test for the (k+1) degree term is not, then the kth degree polynomial is accepted as the best fitting polynomial."

This is evidently the principle we are stating. As always (whatever the approach to model form selection) one must stop somewhere and wherever one stops an assumption is inevitably involved.

The acceptance of ARIMA(2,1,2) as the highest order of complexity to be entertained is inevitably arbitrary to an extent.

However we give an important further justification for this choice by predisclosing the result that in all tests with the data at hand the improvement (in going from ARIMA(1,1,1) to ARIMA(2,1,2)) was consistently very small absolutely and relatively as will be seen subsequently. Finally we note that higher order ARIMA predictors (such as ARIMA(3,1,2) and ARIMA(2,1,3)) could have been entertained (the quantity of data available would support such complexity) but the tests were considered unnecessarily extravagant.

3.0 A GENERAL PROBE FOR OPTIMAL TRAFFIC PREDICTORS; EXPERIMENTS ON OPTIMIZING ARIMA AND ADAPTIVE PREDICTORS

3.1 OPTIMIZING AND COMPARING PERFORMANCE OF ARIMA(1,1,1) AND ARIMA(2,1,2) GENERAL CONSIDERATIONS:

As we have noted, an ARIMA(p,d,q) predictor is to be considered sufficiently complex or more simply "sufficient" if the best ARIMA(p+1,d,q+1) predictor is not significantly better. The use of the term "significantly better" will be made more precise when we present the results. In some cases there may be statistical significance to a certain difference between predictors but no practical significance (i.e. no substantial difference) in the results.

As we have indicated our modified Box-Jenkins analysis is mostly boiled down to three simple tasks:

1. Construct the best performing ARIMA(1,1,1) (one-step) predictor on a given sample of data.

2. Construct the best ARIMA(2,1,2) predictor on the same sample. If we then show that the ARIMA(2,1,2) predictor does not

perform significantly better, then we know that the Box-Jenkins predictor of choice is ARIMA(1,1,1) or simpler. All simpler predictors than ARIMA(1,1,1) are special cases of ARIMA(1,1,1) (i.e. special ranges of the coefficients) therefore, the B-J analysis is completed by:

3. A parametric study of the ARIMA(1,1,1) predictor.

It should be noted that this is not a standard Box-Jenkins analysis but is useful for establishing the limits of predictive power achievable in the traffic prediction problem. We could extend our analysis if necessary to investigate more complex ARIMA predictors.

The Box-Jenkins analysis is closest to our optimization procedure when one is dealing with one-step prediction--or stated differently--working out the optimum one step predictor results in essentially an estimate of the Box-Jenkins ARIMA model. Box and Jenkins then deduce theoretically the 2-step predictor from the model parameters. We can optimize 2-step predictors directly by the methods given in Appendix A and we do so in certain cases, but the majority of our optimization experiments are on one-step predictors as this simplifies the job and in theory should lead the way to optimum predictors of both the one-step and two-step type. There can be little doubt that any conclusions we establish about one-step predictors carry over to two-step predictors. As we have noted, the standard Box-Jenkins analysis is not sensitive to such a distinction.

We now recall the form of the ARIMA(1,1,1) one-step predictor:

1) $$\hat{Z}_{t+1} = \lambda\bar{Z}_{t-1} + (1-\lambda)Z_t$$

2) $$\bar{Z}_t = \Theta\bar{Z}_{t-1} + (1-\Theta)\, Z_t$$

The intermediate quantity (time series) $\bar{Z}_t$ used in this prediction scheme is called the "exponentially weighted moving average" or more simply the "exponential moving average" of the time series Z_t. $\hat{Z}_{t+1}$ is the predicted value of Z_{t+1} based on the observed values Z_t, Z_{t-1}, Z_{t-2},...etc. If we let $W_t = Z_t - Z_{t-1}$ and $\hat{W}_t = \hat{Z}_t - Z_{t-1}$ then equations (1) and (2) yield:

3) $$\hat{W}_{t+1} = \Theta\hat{W}_t - \lambda W_t$$

as is readily shown.*

If Z_t represents the traffic count at time t then, as shown in Appendix E, (3) or (1) and (2) are of the form of the UTCS third generation predictor (Θ replaces the β of Part I, Sec. 4.2, while λ replaces $\beta(1-\alpha_j)$ of that section). If Z_t represents the

*See discussion following equation F. 17 (Appendix F).

difference between the current traffic count and the historical average, then (1) and (2), or (3) are of the form of the UTCS second generation predictor with $\lambda = 1-\alpha-\gamma$ and $\Theta = \gamma$ in terms of the parameters α, γ of Part I Sec. 4.1.

Henceforth (except where noted in Appendix A) W_t will denote the first difference of Z_t, that is $W_t = Z_t - Z_{t-1}$. $\hat{W}_t$ is the predicted value of W_t at time t-1, thus $\hat{W}_t = \hat{Z}_t - Z_{t-1}$.

Note incidentally that the ARIMA(1,1,1) two-step predictor as derived by Box and Jenkins has the same form as the one-step predictor. Since both predictors are derived theoretically from the same model, there is a relation between the two-step and one-step coefficients. If we denote the one-step coefficients by Θ_1, λ_1 and the two-step coefficients by Θ_2, λ_2 then B-J derive:

4) $$\Theta_2 = \Theta_1 \qquad \text{and} \qquad \lambda_2 = \lambda_1 \, (1+\Theta_1-\lambda_1)$$

We, of course, may optimize either or both sets independently depending on the application. The theoretical relation has apparently not been noted in the traffic prediction literature.

Equation 3 forms the backbone of our analysis. The experiments to follow will consist of adding various terms to the right-hand side and then optimizing the coefficient of all terms simultaneously (using the methods of Appendix A).

The most important experiments are to delimit the predictive power of the class of Box-Jenkins ARIMA predictors. The procedure will be the same for both the historical data case and the non-historical data case; the only difference will be that in the historical data case (as with the Second Generation predictor) we perform the tests and experiments on the difference (or the residual) obtained by subtracting the historical average from the current count while in the non-historical case we work directly with the traffic count data. The fundamental experiment consists of optimizing equation (3) with respect to θ and λ . Then the ARIMA(2.1.2) predictor as expressed thusly:

5) $$\hat{W}_{t+1} = a_o\hat{W}_t + a_1\hat{W}_{t-1} + b_oW_t + b_1W_{t-1}$$

is optimized with respect to a_o, a_1, b_o, b_1.

We then compare the mean square error of the optimum predictor of the form (equation 3)(i.e., ARIMA(1,1,1)), with the mean square error of the optimum predictor of the form (equation 5) (i.e., ARIMA(2,1,2)).

3.2 EXPERIMENTS WITH ARIMA(2,1,2) VS. ARIMA(1,1,1)

Several experiments were run to determine if higher order ARIMA predictors (i.e. with higher values of p, d, or q) offered advantages over ARIMA(1,1,1). As we have noted, the preliminary results had all indicated that no more complex predictor offered an advantage over the ARIMA(1,1,1) in either the historical or the non-historical case. In the historical case the ARIMA predictor acts on the residual after the historical average is subtracted out.

The experiments each consisted of optimizing an ARIMA(2,1,2) predictor on half of the available field data (the other half was used to construct the historical average) or about 5240

5-minute intervals over 29 days for specific sensors (i.e. specific street locations). The program and a sample run are given in Appendix B.

It was expected that the ARIMA(2,1,2) would likely show a a greater advantage in the case of non-historical data since the historical data predictors already perform much better and are expected to react to unusual situations which might actually call for a simpler predictor. We note that the ARIMA(2,1,2) predictor, unlike the ARIMA(1,1,1) actually projects increasing or decreasing trends (in general).

The results of the comparison of the optimum ARIMA(2,1,2) predictor with the optimum ARIMA(1,1,1) predictor are shown in Table 1. Column 1 gives the location (sensor, see Appendix D), and Column 2 indicates whether the predictor is based on

TABLE 1. EXPERIMENTS COMPARING ARIMA(2,1,2) VS. ARIMA(1,1,1)

1 Sensor	2 Historical Average Used?	3 a_o	4 a_1	5 b_o	6 b_1	7 RMS Error ARIMA (2,1,2)	8 RMS Error ARIMA (1,1,1)	9 % Difference (2,1,2) vs (1,1,1)	10 RMS Error (0,0,0) (no change)
1	no	-.408	.379	1.27	-.521	10.63	10.77	1.3 %	11.6
1	yes	-.743	-.531	.112	.592	8.92	8.94	0.18%	---
2	no	-.467	.292	1.168	-.501	13.09	13.24	1.1 %	14.6
2	yes	-.817	-.494	.266	.54	11.15	11.15	0.0 %	---
3	no	-.589	-.065	.596	-.121	11.31	11.39	0.7 %	13.2
3	yes	-.823	-.423	.419	.417	10.29	10.37	0.8 %	---
4	no	-.496	.09	.75	-.29	12.7	12.78	0.6%	14.3
4	yes	-.83	-.566	.243	.588	10.9	10.9	0.1 %	---

historical averages or not. Columns 3, 4, 5, and 6 gives the values of a_o, a_1, b_o, and b_1 respectively in equation 5 for the optimized ARIMA(2,1,2) predictor. Column 7 gives the RMS (root over square) error for the corresponding ARIMA (2,1,2) predictor. Column 8 gives the RMS error for the optimized ARIMA(1,1,1) predictor (i.e., 2nd generation form if historical data is used, 3rd generation form if historical data is not used). Column 9 gives the percentage difference between column 7 and column 8. Column 10 gives the RMS error for the simplest predictor, that is the "non-predictor," which simply bases its prediction on the assumption that the traffic count will stay the same in the next time interval (this can be called the ARIMA(0,0,0) predictor).

Referring to Table 1 we note the following observations. In each location (i.e. each sensor) the performance was in this order (best to worst): 1. ARIMA(2,1,2) (historical), 2. ARIMA(1,1,1) (historical), 3. ARIMA(2,1,2) (non-historical), 4. ARIMA(1,1,1) (non-historical), 5. ARIMA(0,0,0). The biggest improvement is in going from ARIMA(0,0,0) to ARIMA(1,1,1)(non-historical) i.e. in going from a zero change ("non-predictor") predictor to a UTCS third generation type. The second largest change is for passing from no use of historical averages to use of historical averages (i.e. between ARIMA(2,1,2) (non-historical) to ARIMA(1,1,1) (historical)). In contrast it is seen that the passage from ARIMA(1,1,1) to ARIMA(2,1,2) (which gives rise to the percent difference in column 9) in no case yields more than a 1.3% improvement in RMS error. This occurred at sensor 1 for the non-historical case. The improvement in passing from ARIMA(1,1,1) to

ARIMA(2,1,2) was substantially less in the historical case than in the non-historical case except at sensor 3 which showed the greatest improvement, 0.8%, for passage from ARIMA(1,1,1) to ARIMA(2,1,2) in the historical case and an essentially equal improvement in the non-historical case. Although the statistical significance for small differences using this method have not been determined, it is doubtful whether any of these improvements in going from ARIMA(1,1,1) to ARIMA(2,1,2) is statistically significant. There is surely no practical significance for an improvement of less than 2% in the RMS error (which represents less than 0.3% of the traffic count itself or less than 1 vehicle in 15 minutes).

We note in passing that if $a_o+a_1+b_o+b_1 \approx 1$ there would be evidence that the ARIMA(2,1,2) predictor could be replaced by an ARIMA(1,2,2) predictor (to which it would be equivalent under those circumstances) which would in turn suggest checking out ARIMA(2,2,2). As the equality, $a_o+a_1+b_o+b_1 = 1$, is seen not to hold even approximately (in any of the cases in Table 1), we have here no reason to entertain the possibility of a higher value of d (i.e. ARIMA(1,2,2) or ARIMA(2,2,2)).

We further note that all evaluations in this section are based on exponential moving average evaluations of the RMS error (all with time constants greater than 500 time units). The RMS errors in each horizontal line in Table 1 are based on the same moving average weights and are thus quite comparable. The reduction in comparability in passing from row to row in Table 1 is slight (less than a few tenths of a percent). Tables 2-8 (to be

ARIMA(2,1,2) was substantially less in the historical case than in the non-historical case except at sensor 3 which showed the greatest improvement, 0.8%, for passage from ARIMA(1,1,1) to ARIMA(2,1,2) in the historical case and an essentially equal improvement in the non-historical case. Although the statistical significance for small differences using this method have not been determined, it is doubtful whether any of these improvements in going from ARIMA(1,1,1) to ARIMA(2,1,2) is statistically significant. There is surely no practical significance for an improvement of less than 2% in the RMS error (which represents less than 0.3% of the traffic count itself or less than 1 vehicle in 15 minutes).

We note in passing that if $a_o+a_1+b_o+b_1 \approx 1$ there would be evidence that the ARIMA(2,1,2) predictor could be replaced by an ARIMA(1,2,2) predictor (to which it would be equivalent under those circumstances) which would in turn suggest checking out ARIMA(2,2,2). As the equality, $a_o+a_1+b_o+b_1 = 1$, is seen not to hold even approximately (in any of the cases in Table 1), we have here no reason to entertain the possibility of a higher value of d (i.e. ARIMA(1,2,2) or ARIMA(2,2,2)).

We further note that all evaluations in this section are based on exponential moving average evaluations of the RMS error (all with time constants greater than 500 time units). The RMS errors in each horizontal line in Table 1 are based on the same moving average weights and are thus quite comparable. The reduction in comparability in passing from row to row in Table 1 is slight (less than a few tenths of a percent). Tables 2-8 (to be introduced later) give RMS errors for ARIMA(1,1,1) predictors based on straight averages over the data base rather than long time constant exponential moving averages. There is consequently a slight discrepancy between RMS errors in Table 1 compared to Tables 2 - 6, but it is no more than 1.4% which appears to be quite negligible. This discrepancy in any case does <u>not</u> affect the comparison (here) of optimum ARIMA(2,1,2) predictors vs optimum ARIMA(1,1,1) predictors which comparisons are based on identical exponentially weighted moving averages. All the comparisons we have noted from Table 1 are unaffected but we simply call attention to the negligible discrepancy in the absolute minimum RMS error in the ARIMA(1,1,1) case (historical and non-historical) between Table 1 and Tables 2 - 6 and the reason for this negligible discrepancy.

To recapitulate: the results indicate that the improvement (in going from ARIMA(1,1,1) to ARIMA(2,1,1)) is slight and probably not statistically significant. The improvement <u>was never more than 1.3%</u> in the root mean square and preliminary estimates of the bias introduced by optimizing over the additional parameters indicate that the improvement in most cases did not exceed the bias. There was less improvement in the historical case than in the non-historical case, but this is not conclusive since the improvement of ARIMA(2,1,2) over ARIMA(1,1,1) was marginal and inconclusive in <u>all</u> cases, i.e. for all locations tested, both historical and non-historical.

Therefore (concerning possible improvement in performance using ARIMA(2,1,2) vs. ARIMA(1,1,1) historical or non-historical) we conclude:

1. The improvement is of no practical significance--i.e. a reduction of the error by at most .1 to .2 vehicles out of about 10 to 15 root mean square error vehicles.

2. There appeared to be no statistically significant improvement.

3. Since the filter was more complex, any gains could be quickly lost due to "detuning" if traffic conditions changed slightly.

All this tends to show that there is no advantage to using more complex (higher order) ARIMA predictors over the simpler ARIMA(1,1,1) predictors. This conclusion is, of course, based on experience with a specific data set (see Appendix D). The generalizability of the conclusion is difficult to assess. Further studies of the type done here must be done on other data from other locations, before this matter is considered closed. However, there is nothing in the results we obtained to suggest that the results would be substantially different on other data.

3.3 IMPROVEMENTS THROUGH OPTIMALLY WEIGHTED NON-LINEAR ADAPTIVE TERMS, METHOD AND EXPERIMENTS:

A similar set of experiments were run to see if non-linear adaptive terms could be added to the ARIMA(1,1,1) predictor to improve its performance. Since the non-historical case requires that the predictor adapt to rather large and quickly developing changes, the experiments on adaptation were all carried out for the non-historical case.

To show how the non-linearities were introduced, recall the form of the ARIMA(1,1,1) one-step predictor:

$$6) \quad \hat{W}_t = \Theta \hat{W}_{t-1} - \lambda W_{t-1}$$

where $\quad W_t = Z_t - Z_{t-1}$ and $\hat{W}_t = \hat{Z}_t - Z_{t-1}.$

In practice, λ is usually not extremely different from θ.
If $\lambda = \Theta$ then the whole predictor reduces to the simple exponential moving average, i.e.

$$7) \quad \hat{Z}_t = \overline{Z}_{t-1} = \Theta\hat{Z}_{t-1} + (1-\Theta)Z_{t-1}$$

If we denote $\hat{W}_t - W_t$ by δ_t

then δ_t can be called the "overshoot" (or residual). It is the amount by which the predicted value differs from the actual value at time t.

For the case of the exponential moving average

$$8) \quad \hat{W}_t = \theta\delta_{t-1}$$

The non-linear terms by which we alter equation $\overline{3}$ can be of various forms. Several types of non-linear terms were tried in preliminary experiments. The only one which showed even a hint of promise involved trying to adapt the coefficient of δ_t. We followed up on this with two experiments to adapt the coefficient of δ_t: in the manner of Trigg-Leach and with exponentially weighted least squares.

Accordingly setting:

9) $\hat{W}_t = \theta\hat{W}_{t-1} - \lambda W_{t-1} + \gamma A\delta_{t-1}$

where A is some function of the previous values of W_t and $\hat{W}_t$ (it is not a constant, otherwise the last term would be redundant). The Trigg-Leach predictor (Part I Sec. 4.1.2.3) is equivalent to equation 9 with $\gamma = \Theta = 1$ and

$$A = -\left| \frac{Q(t)}{\Delta(t)} \right|$$

Therefore, we optimized (equation 9) (over θ,λ,δ) with $A = -\left|\frac{Q(t)}{\Delta(t)}\right|$ so that the Trigg Leach filter would be in the family of filters optimized over.

Recall that:

$$Q(t) = \rho Q(t-1) + (1-\rho)\delta_t$$

$$\Lambda(t) = \rho\Lambda(t) + (1-\rho)\ |\delta_t|\quad 0<\rho<1$$

An experiment to determine the effects of optimizing a Trigg-Leach type term appended to an ARIMA(1,1,1) predictor (i.e. equation 9 with A as just defined), using non-historical data, was carried out on the data for one location only (sensor 4). The result was an improvement in the root mean square error (over that for a straight ARIMA(1,1,1) of slightly over 2.0%, i.e. just under

0.3 cars per 5 minutes out of a root mean square error of 12.2 cars per 5 minutes. This was better than the improvement obtained going from ARIMA(1,1,1) to ARIMA(2,1,2) which was a reduction in RMS error of less than 1%.

The most effective value of ρ (when using the predictor on the data) was about .85 to .9. For $\rho = .9$ the coefficients in the filter (equation 9) were $\theta = .85$, $\lambda = .78$, $\delta = 1.0$. This is to be compared to the optimized θ and λ for the ARIMA(1,1,1) predictor for sensor 4 (non-historical): $\theta = .43$, $\lambda = .495$. Clearly, the optimum predictor was closer to a straight Trigg-Leach smoother and as we noted the Trigg-Leach did result in the lowest mean square error (the reduction was larger than for any other technique involving non-historical data only). This was a surprising result, but the RMS reduction of 2% (.3 vehicles per 5 minute period) would not appear to be of practical significance, although it was perhaps statistically significant.

Another similar type of adaptive term was tried. This term was derived for an optimum adaptive term using exponentially weighted least squares as follows:

$$A = \frac{r(t)}{D(t)} \qquad \text{where}$$

$$r(t) = \rho r(t-1) + (1-\rho)\delta_t \, \delta_{t-1}$$

$$D(t) = \rho D(t-1) + (1-\rho)\delta_t \, \delta_t \qquad (0>\rho>1)$$

The basic idea is the same as with the Trigg-Leach approach: $r(t)$ is small compared to $D(t)$ when δ_t is uncorrelated with δ_{t-1} but approaches 1 (for ρ near 1) when δ_t is highly correlated to δ_{t-1}. Thus, if $\rho = .6$, $A(t)$ measures the <u>local</u> correlation of the "overshoots" (or residuals), δ_t, and weights the current δ_t high if the correlation is currently high. This seemed to be a very attractive type of adaptation term. However, when tried for sensor 3, non-historical, with $\rho = .6$, it gave essentially no improvement in the mean square error. When we went to sensor 2, however, (sensors 1, 2 and 4 had more rapidly changing patterns than did sensor 3) there was an improvement of the same order as achieved with ARIMA(2,1,2) i.e. about 1.5% in the root mean square error. Again, the improvement is probably not statistically significant and certainly not worth the added complexity in practical terms. Interestingly, the added term had a large effect on the optimum coefficients even though the decrease in mean square error was slight: Without the adaptation term (sensor 2, one-step, non-historical) we obtained the following optimum values:

$$\Theta = .39, \; \lambda = .46$$

Adding the adaptive term, we obtained:

$$\Theta = .54, \; \lambda = .46, \; \gamma = .54$$

The adaptive term was working but essentially no improvement was achieved. This is similar to the result obtained with the Trigg-Leach adaptive term. Our tentative conclusion based on these limited experiments with adaptation is that adaptive terms have no significant predictive power to add to ARIMA(1,1,1) predictors (non-historical). More experiments of this type could be performed on data from locations expected to be considerably different from the Toronto locations studied here. Based on the above results it would, of course, be quite surprising if an adaptive predictor outperformed even higher order ARIMA predictors.

The testing of optimized predictors with non-linear terms included into the basic ARIMA(1,1,1) framework should be considered secondary to the testing of optimized ARIMA(2,1,2) predictors. From a theoretical point of view the ARIMA(2,1,2) predictors being part of the B-J hierarchy are to be given more consideration. As has just been seen the testing of ARIMA(2,1,2) predictors was much more systematic and extensive on this project than was the testing of non-linear adaptive predictors--consistent with the theoretical position.

The results of the general probe of higher order ARIMA predictors and of adaptive predictors yielded one solid conclusion: No more complex predictor outperformed the ARIMA(1,1,1) predictor by more than 2% in these experiments. Other data might alter this conclusion, however, the tentative conclusion is that traffic count data does not lend itself to complicated prediction processes, but is best predicted by simpler predictors. This suggests that as a general rule traffic predictors should be ARIMA(1,1,1) or simpler for both the historical data case (working on the residuals after subtracting out the historical mean pattern) and the non-historical case. Complex filters should be avoided because even if they apparently perform slightly better in a certain situation, they could be more sensitive to changing conditions.

The property of retaining optimal or near optimal performance under altered conditions is sometimes referred to as "robustness" in statistical parlance.* Traffic predictors are in general surprisingly robust. We shall present some quantitative results on the robustness of the ARIMA(1,1,1) predictors below.

4.0 PARAMETRIC STUDY OF ARIMA(1,1,1) SECOND AND THIRD GENERATION UTCS PREDICTORS

Having presented some evidence that more complex B-J predictors than ARIMA(1,1,1) (i.e. than the Second Generation UTCS predictor for the historical data case or than the Third Generation UTCS predictor for the non-historical case) are not warranted for traffic prediction, we now examine more closely the ARIMA(1,1,1) predictors. In so doing, it is desired to find the optimum value for the parameters θ and λ for various situations, to examine the sensitivity of performance to variations in these parameters, and to determine what practical simplifications and/or improvements are possible in terms of:

1. Uniform parameter values for use in all cases.
2. Reduction of order of ARIMA predictor.
3. Increased responsiveness

--all with negligible degradation of performance below the best obtainable with ARIMA(1,1,1) with optimized coefficients.

---

*This term has varied usage in statistical literature; the usual context refers to the behavior of certain statistics derived for normal distributions as they are used on non-normal data.

Tables 2, 3, 4 and 5 show the values of Θ and λ (see equations 1, 2, & 3) which optimize the one-step and two-step predictors for both the historical and non-historical case. Thus Table 2 gives the optimum values of Θ and λ for the (UTCS) Second Generation 1-step predictor (for this data), Table 3 the optimum values for a Second Generation two-step predictor, Table 4 the optimum values for a Third Generation one-step predictor and Table 5 the optimum values for a Third Generation two-step predictor. The three error measures used in all four tables are: root mean square, root mean fourth power, and mean absolute value. (See Table 2.) The root mean fourth error is more sensitive to infrequent large errors while the mean absolute error is more sensitive to frequent smaller errors, both as compared to the basic root mean square measure.

Tables C-1 to C-4 Appendix C list the performance (as evidenced by the same three error measures as used in Table 2-5) of various ARIMA(1,1,1) predictors under various stated conditions. The fact that a uniform set of coefficients (θ,λ) can be used to obtain nearly optimum performance is highlighted in Tables 6-9 which are to be comparod with Tables 2-5 to see how small the increase in root mean square error is in each case--i.e. for each sensor, one-step or two-steps, historical or non-historical--when one uses a single set of coefficients for each sensor. We have shown in Tables 6-9 the effect of using the smallest optimum θ and λ for all four sensors. Thus, in comparing Table 2 and 6, we find that if the optimum values for sensor 1 are used for sensor 4 (historical, one-step) the root mean square error increases only from 10.89 to 11.02 and the increase is smaller for the othor sensors.

Table 2

Historical Average

1-step

Optimum θ and λ

Sensor#	θ	λ	$(\overline{\delta^2})^{1/2}$	$(\overline{\delta^4})^{1/4}$	$\overline{\|\delta\|}$
1	.79	.74	8.90	13.47	6.385
2	.872	.81	11.07	15.35	8.535
3	.85	.80	10.40	15.31	7.935
4	.88	.84	10.89	16.50	8.268

and measures of prediction error,

$(\overline{\delta^2})^{1/2}$ = root mean square (RMS) error

$(\overline{\delta^4})^{1/4}$ = fourth root of the mean of the fourth power of the error

$\overline{|\delta|}$ = mean absolute value of error

These measures are decreasingly sensitive to persistent small errors and increasingly sensitive to short lasting large errors as one passes from

$$\overline{|\delta|} \text{ to } (\overline{\delta^2})^{1/2} \text{ to } (\overline{\delta^4})^{1/4}$$

Table 3

Historical Average

2-step

Optimum θ and λ

Sensor#	θ	λ	$(\overline{\delta^2})^{1/2}$	$(\overline{\delta^4})^{1/4}$	$\overline{\lvert\delta\rvert}$
1	.81	.779	9.18	14.33	8.044
2	.88	.86	11.27	15.74	8.676
3	.84	.88	10.55	15.88	7.990
4	.88	.905	11.04	16.78	8.338

Table 4

Non-historical

1-step

Optimum θ and λ'

Sensor#	θ	λ	$(\overline{\delta^2})^{1/2}$	$(\overline{\delta^4})^{1/4}$	$\overline{\lvert\delta\rvert}$
1	.26	.39	10.73	16.59	7.565
2	.39	.46	13.14	18.20	10.120
3	.57	.59	11.37	16.44	8.763
4	.43	.495	12.96	18.60	9.674

Table 5

Non-historical

2-step

Optimum Values for θ and λ

Sensor#	θ	λ	$(\overline{\delta^2})^{1/2}$	$(\overline{\delta^4})^{1/4}$	$\overline{\lvert\delta\rvert}$
1	.09	.36	12.55	19.43	8.706
2	.30	.464	14.96	21.00	11.404
4	.325	.52	14.29	20.70	10.801

Table 6

Historical Average

1-step

Other Smaller Values of θ and λ

Sensor#	θ	λ	$(\overline{\delta^2})^{1/2}$	$(\overline{\delta^4})^{1/4}$	$\overline{\lvert\delta\rvert}$
1	.77	.72	8.91	13.49	6.395
2	.79	.74	11.15	15.39	8.622
3	.79	.74	10.45	15.30	7.984
4	.79	.74	11.02	16.60	8.378

$\theta = .75 \quad \lambda = .75$

Sensor#	$(\overline{\delta^2})^{1/2}$	$(\overline{\delta^4})^{1/4}$	$\overline{\lvert\delta\rvert}$
1	8.91	13.42	6.396
2	11.19	15.43	8.658
3	10.48	15.37	8.009
4	11.06	16.66	8.412

Table 7

Historical Average

2-step

Other Smaller Values of θ and λ

Sensor#	θ	λ	$(\overline{\delta^2})^{1/2}$	$(\overline{\delta^4})^{1/4}$	$\overline{\lvert\delta\rvert}$
1	.8	.7	9.21	14.18	6.543
2	.81	.779	11.35	15.75	8.749
3	.81	.779	10.61	15.92	8.044
4	.81	.779	11.16	16.90	8.446

$\theta = .8 \quad \lambda = .75$

Sensor#	$(\overline{\delta^2})^{1/2}$	$(\overline{\delta^4})^{1/4}$	$\overline{\lvert\delta\rvert}$
1	9.19	14.25	6.508
2	11.39	15.79	8.781
3	10.65	15.96	8.077
4	11.22	16.97	8.485

$\theta = .75 \quad \lambda = .75$

Sensor#	$(\overline{\delta^2})^{1/2}$	$(\overline{\delta^4})^{1/4}$	$\overline{\lvert\delta\rvert}$
1	9.20	14.18	6.526
2	11.45	15.85	8.832
3	10.68	16.02	8.099
4	11.26	17.01	8.524

Table 8

Non-historical

1-step

Smaller Values of θ and λ

Sensor#	θ	λ	$(\overline{\delta^2})^{1/2}$	$(\overline{\delta^4})^{1/4}$	$\overline{\lvert\delta\rvert}$
1	.2	.37	10.74	16.57	7.578
2	.26	.39	13.20	18.22	10.176
3	.26	.39	11.73	17.17	9.016
4	.26	.39	12.88	18.75	9.789

Table 9

Non-historical

2-step

$\theta = .2 \quad \lambda = .4$

Sensor#	$(\overline{\delta^2})^{1/2}$	$(\overline{\delta^4})^{1/4}$	$\overline{\lvert\delta\rvert}$
1	12.57	19.50	8.698
2	14.99	20.92	11.469
4	14.39	20.81	10.895

Tables 6-9 show the apparent extreme insensitivity of the error to increasing or decreasing λ and especially Θ, substantially. One may observe not only the root mean square error but also the mean absolute error and the root mean 4th error (the fourth root of the mean of the fourth powers of the errors or residuals).

The use of smaller than optimum values of Θ and λ therefore, does not apparently decrease the mean performance substantially. The decrease actually increases the robustness. Robustness is a statistical term indicating the lack of sensitivity (i.e. criticalness) of performance to non-standard or unexpected conditions. We use the term here to denote the property of a predictor to perform well in a traffic situation which is somewhat different, for any reason, from traffic situations considered in constructing or tuning* the predictor. It can be seen that if tuning is critical the predictor is not robust. This property is closely related to but not identical with responsiveness. The fact that lower values of Θ and λ lead to increased robustness can be seen from the fact that the sensitivity** of performance to decreasing Θ and λ is small. As we noted when a set of values of Θ and λ were selected which were essentially optimum

*By "tuning" we mean calibration, that is the selection of specific values of the parameters to fit the specific traffic situation.

**In this sense "sensitivity" refers to the criticalness of the tuning. The tuning is sensitive if a slight change in conditions requires different parameters for nearly optimal performance. Thus in our present terminology "sensitivity of tuning" = "non-robustness." Obviously "sensitivity" is not equivalent to "responsiveness", in effect more like the opposite will be the case.

at the location where the smallest optimum Θ and λ were obtained, these values of Θ and λ resulted in predictors which were nearly optimum at the other locations (the loss in performance below optimal was always less that 4% and usually less than 1%). Thus, by decreasing Θ and λ we reach the optimum values for a wider range of conditions with little sacrifice in performance from conditions which are optimized by higher values of Θ and λ.

Furthermore, it can be shown that the smaller values of Θ and λ lead to more responsive predictors--i.e., predictors which respond more quickly to changing conditions. Thus, for example, if the Θ and λ for a predictor using historical data could be reduced substantially without degrading mean performance, the response of the predictor under completely atypical traffic conditions could be nearly as good as a non-historical predictor.

For the non-historical case the optimum Θ and λ vary widely and in some cases (when Θ is very small) there is less sensitivity to <u>increasing</u> rather than decreasing Θ. In these situations moderate values are to be recommended.

For the historical case a value of .9 has been recommended (e.g. see Ref. 3,6) in the past for Θ. This value might have been chosen as optimum for certain locations, but as we have seen it is best (in the historical case at least) to take as <u>small</u> a value as possible for Θ as is consistent with <u>near</u> optimum performance at all locations. This will often be near the optimum value of Θ at the location where the optimum value is least. This means that an average or median of Θ and λ over various locations is not recommended, rather the least near optimum values should be chosen. We see that for these four locations in Toronto an average value for Θ of about .85 and for λ of about .8 would have been chosen but $\Theta = .79$, $\lambda = .74$ or $\Theta = .8$, $\lambda = .75$ would in all probability be better for universal use at these locations or any with similar conditions. By "universal use" we refer to the use of a single set of parameters at a large number of locations. By choosing a single set which perform well at many locations we achieve several ends:

1. Implementation is easier as the predictors do not have to be tailored to each location
2. Individual determination of parameters at each location is subject to error and the optimum values are such as to determine predictors which are more sensitive in performance to actual conditions than suboptimum universal values
3. As conditions change over a long period of time the predictors would not degrade in performance so quickly
4. Responsiveness to atypical traffic patterns (special events etc.) would be greater

Note that in the non-historical two-step case, $\Theta = .2$, and $\lambda = .4$ gives very near optimum results in each instance. In the past, values of Θ as high as .95 have been proposed for this case (see Ref. 3, also Ref. 2 of Part I). Such high values are not warranted for this Toronto data.

We note incidentally the degree to which the theoretical equation derived by Box and Jenkins to relate the optimum 1-step (ARIMA(1,1,1)) predictor to the optimum 2-step predictor is either confirmed or contradicted by this data.

We recall the theoretical relation between the Θ and λ for one- and two-step predictors, namely equation (4): $\Theta_2 = \Theta_1$, $\lambda_2 = \lambda_1(1+\Theta_1-\lambda_1)$. The agreement between the actual optimum two-step coefficients and that derived by the theoretical formula from the one-step coefficients appears to be satisfactory in the historical case and unsatisfactory in the non-historical case. For example, for the historical case one-step coefficients for sensor 1 we have $\Theta = .79$, and $\lambda = .774$. The theoretical formula then gives for the two-step case $\Theta = .79$, $\lambda = .777$ compared to the actual values $\Theta = .81$, $\lambda = .779$. For the same sensor non-historical $\Theta_1 = .26$, $\lambda_1 = .39$/yield $\Theta_2 = .26$, $\lambda_2 = .34$ (theorectical) while the actual values are $\Theta_2 = .09$, $\lambda_2 = .36$. However, if we consider the extreme insensitivity of the root mean square error to the co-efficients, the theoretically derived coefficients (for the two-step case) are nearly as good as the actual optimum two-step coefficients for both the historical and non-historical cases.

The better agreement with the theoretical B-J relations in the historical data case is consistent with the hypothesis that the residuals after subtracting off historical average data is more like a true ARIMA(1,1,1) process than is the raw traffic count data itself.

Finally we recall as noted in Part I that predictors based on historical average data (Second Generation) have been shown to perform consistently better than those not based on historical data (Third Generation). The data used in this study has already been used by Kreer (Ref. 3) in his study which establishes this point. As the values of Θ and λ found here to be optimum are different from those reported by Kreer we note that Tables 2, 3, 4, 5 using the optimum point support these values of Θ & λ very strongly. (We have already pointed out the same effect in Table 1). The difference in RMS error between Second and Third Generation predictors with optimized coefficients vary from 10% to 20% for 1-step predictors and from 30% to 45% for 2-step predictors as may be seen from Tables 2 through 5. For example, Table 2 shows that at sensor 1, using a historical based one-step predictor, the RMS error was 8.9, while using a non-historical one-step predictor the RMS error was 10.73. The differences in RMS error between Second and Third Generation (i.e., between historical data and non-historical data) is thus far larger than any differences in RMS error produced by going to higher order ARIMA or adaptive predictors which as we noted were not of practical significance.

5.0 Conclusions

A new technique has been developed to optimize ARIMA (Box Jenkins (Ref. 1) predictors on data. The technique easily allows a general form to be optimized so that the basic ARIMA form can be augmented by non-linear adaptive terms and even by more general terms such as terms involving concurrent time series (multichannel data).

The technique has been applied to a sample of traffic count data from Toronto, Canada large in extent (thousands of points) but limited in scope (four locations).

The conclusions drawn from the analysis are that traffic volume is best predicted with very simple predictors when only non-historical data, i.e., near-past values of the traffic counts themselves, are being used, and with similar techniques applied to the residuals after the historical average is subtracted off when historical data is used. In short, the Second and Third Generation predictor forms were not found to be more effective than more complicated predictors.

This conclusion was arrived at by comparing the performance of ARIMA(2,1,2) predictors optimized on the same data as ARIMA(1,1,1) predictors. The ARIMA(2,1,2) predictor did not perform significantly better than ARIMA(1,1,1) predictor: improvement in performance observed at any sensor (historical or non-historical) was 1.5% in the RMS error. Thus, it was concluded that higher order ARIMA predictors are not warranted on this data.

A more limited experiment indicated no substantial improvement using non-linear adaptive terms (optimally combined with the ARIMA(1,1,1) terms).

Due to the limited nature of the data and the limited experiments using adapation, the conclusion must be considered tentative. However, the conclusion that the Second and Third Generation UTCS predictors (i.e., ARIMA(1,1,1)) can be made optimal or near optimal through the right choice of coefficients was persistent in all experiments and there is no evidence to lead one to expect it to fail.

Consistent results can however strongly suggest practical limitations as we have noted.

Further observations and recommendations about the details of implementation of traffic predictors have been given. The need for responsiveness and "robustness" as well as good mean performance has been advised; this leads to an altered choice of paramenters selected. In general, smaller values of θ and λ are recommended than are cited in previous literature on UTCS predictors. It appears that smaller values are optimal in some cases while in others the sacrifice in mean performance is negligible.

One of the most important results of this study was the development of a technique for optimizing ARIMA predictors and ARIMA-type predictors containing non-ARIMA or generalized terms. The mathematical details are given in Appendix A of this report.

In the traffic prediction context the largest potential for this predictor optimization capability would appear to be in the ability it affords to evaluate the enhancement capabilities of the basic predictors with external information i.e., with information derived from other real-time quantities, such as traffic at other locations or with mean occupancy, which is measured at the same sensors that measure the volume being predicted.

A continuation of this study into the multichannel aspects of traffic prediction (just alluded to) may be possible with presently available data. Some data for the city of Indianapolis, obtained from Prof. Kreer at the same time as the Tornoto data, contains occupancy information. The Indianapolis data was used in Kreer's study of Second and Third Generation traffic predictors, but occupancy data has apparently not been included in traffic volume predictors. Also, the Toronto data provides the possibility of including data from sensors at multiple locations. The Toronto and Indianapolis data would provide a start at such a wider investigation into traffic volume prediction using multiple sensor data and occupancy data.

Eventually the methods we have developed should be applied to a wider variety of data in an attempt to further explore and define optimum traffic predictors. Data from other locations should be obtained to further explore the possibilities of higher order ARIMA predictors as well as non-linear adaptive predictors which were both tentatively ruled out as they did not substantially improve predictive capability on this data.

The methods are now available for relatively easy analyses. Further application should be quick and inexpensive once data is obtained.

References

1. Time Series Analysis: Forecasting and Control, 2nd edition (1976), by George E. P. Box and Gwilym M. Jenkins; Holden-Day publisher, San Francisco.
2. Forecasting and Time Series Analysis by Douglas C. Montgomery and Lynwood A. Johnson; McGraw Hill, New York (1976).
3. A Comparison of Predictor Algorithms for Computerized Traffic Control Systems by John B. Kreer; Traffic Engineering, April 1975.

4. The Analysis of Variance by Henry Scheffé; John Wiley and Sons, Inc., New York (1959).

5. Applied Linear Statistical Models by John Neter and William Wasserman; Richard D. Irwin, Inc., Illinois (1974).

6. Surface Street Traffic Control: 1980 and Beyond by R. W. Kessman and H. Lum in Compendium of Technical Papers, 47th Annual Meeting of Institute of Transportation Engineers, Oct. 1977.

7. Experimental Statistics - National Bureau of Standards Handbook 91 (1963) by Mary G. Natrella (GPO).

APPENDIX A

PREDICTION OPTIMIZATION ALGORITHMS

The Box-Jenkins ARIMA models are considered to yield the most accurate, sophisticated and comprehensive systematic analysis of time series prediction currently available for general use. The term "ARIMA" stands for "AutoRegressive Integrated Moving Average." To assume that a time series (such as a sequence of five minute traffic counts) is modelled by an ARIMA process is to assume that the development (in time) of the real time series proceeds as might some true (hypothetical) ARIMA process. All the ARIMA processes are generated by finite iterative linear transformations on a sequence of uniform variance, normally distributed, uncorrelated "random shocks." It is convenient to refer to the latter as time sampled random noise. The reader is referred to Reference 1 for a complete description of the ARIMA models and their application but here we describe very briefly a few salient features.

The simplest finite explicit expression of the general ARIMA(p,d,q) model is given by Equation 4.2.1 of Reference 1:

A1) $Z_t = \omega_1 Z_{t-1} + \cdots + \omega_{p+d} Z_{t-p-d} - \theta_1 a_{t-1} \cdots \theta_q a_{t-q} + a_t$

where Z_t represents the modelled (ARIMA(p,d,q)) time series, and a_t represents the series of random shocks (discrete sampled white noise), and ω_k, θ_ℓ ($k=1,p+d;\ell=1,q$) represent constant multipliers The ω_k's and θ_ℓ's are arbitrary subject to certain constraints -- both equality constraints and inequality constraints. The equality constraints limit the number of independent parameters (degrees of freedom) to p+q (p independent parameters determined by $\omega_1,\ldots,\omega_{p+d}$ and q independent parameters determined by $\theta_1,\ldots,\theta_q$).

The equality restrictions on the ω's are quite important and represent a modelling of the "order of non-stationarity." This can be explicated at the expense of making the model implicit for Z_t. Thus let $W_t = \nabla Z_t = Z_t - Z_{t-1}$ let $\nabla^2 Z_t = \nabla W_t = W_t - W_{t-1} = Z_t - 2Z_{t-1} + Z_{t-2}$ and in general let $\nabla^d Z_t$ represent the dth difference of Z_t; in particular $\nabla^{d+1} Z_t = \nabla^d Z_t - \nabla^d Z_{t-1}$. In the Box-Jenkins modelling it is assumed that even if Z_t is non-stationary, then $\nabla^d Z_t$ is stationary for some small integer value of d. Thus, if one considers A2) $W_t = \nabla^d Z_t$ rather than Z_t itself, W_t will be stationary and so it is modelled as an autoregressive moving average process.

A3) $W_t = \phi_1 W_{t-1} + \ldots + \phi_p W_{t-p} - \theta_1 a_{t-1} \ldots - \theta_q a_{t-q} + a_t$

Equation A3 together with Equation A2 is equivalent to Equation A1. The quantities a_t and θ_ℓ are the same between Equations A1 and A3. The ω_k determine the ϕ_k and vice versa when the order of differencing (d) is taken into account. Now θ_ℓ and ϕ_k are subject only to inequality constraints ensuring stationarity and invertibility (see Reference 1). Equation A3 is said to determine an ARMA(p,q) process, W_t; while Z_t having W_t as its d'th difference is called an ARIMA(p,d,q) process.

Equation A3 for the ARMA process is evidently very close in form to Equation A1 for the ARIMA process. In fact, the form of Equation A3 substituting p+d for p and Z for W would be exactly that of Equation A1. The difference is in the restrictions on ω_k and ϕ_k. Consider for the moment an ARMA(p,q) process Z_t. Then:

A4) $Z_t = \phi_1 Z_{t-1} + \ldots + \phi_p Z_{t-p} - \theta_1 a_{t-1} - \ldots - \theta_q a_{t-q} + a_t$

According to Box-Jenkins, the one step predictor Z_t for such a process is given by

A5) $\hat{Z}_t = \phi_1 Z_{t-1} + \ldots + \phi_p Z_{t-p} - \theta_1 (Z_{t-1} - \hat{Z}_{t-1}) - \ldots - \theta_q (Z_{t-q} - \hat{Z}_{t-q})$

where $\hat{Z}_t$ is the predicted value of Z_t based on $Z_{t-1}, Z_{t-2}, \ldots$ (i.e. on all values of Z_τ up to $\tau = t-1$: hence it is called a one step predictor since it only predicts one step ahead). Equation A5 follows from the rules given by Box-Jenkin's equations 5.1.18 and 5.1.22 (Reference 1) and the sequels. In particular, a_τ is estimated by 0 when $\tau = t$ (or any other future time) and a_τ is estimated by $Z_\tau - \hat{Z}_\tau$ when $\tau = t-1, t-2, t-3 \ldots$ (any present or past time). The net effect of these rules (as applied here) is that the recursive Equation A5 for the (on-step) predictor is obtained from A4 by replacing a_τ (for all τ) by $Z_\tau - \hat{Z}_\tau$. A similar equation to A5 could be written down for an ℓ-step predictor. $\hat{Z}_t(\ell)$ for an arbitrary ARIMA(p,d,q) process. However, the coefficients would be subject to constraints or side conditions. This is a minor annoyance, but we shall avoid dealing explicitly with it by limiting ourselves at first to the form (A5), which also serves as the predictor for $W_t = Z_t - Z_{t-1}$ for an ARIMA(p,1,q) process. (Incidentally from this point on in this Appendix and elsewhere throughout this report W_t always refers to the first difference of Z_t. Earlier in this Appendix and in Reference 1, W_t is used to refer to arbitrary differences of Z_t.) Thus our development will be explicitly in terms of Z_t, where Z_t will be predicted according to an optimal one-step predictor for an ARMA(p,q) process and so Z_t will either be the time series to be predicted (in our case the traffic counts if we seek an ARMA(p,q) = ARIMA(p,0,q) predictor) or else Z_t will be the first difference of the time series to be predicted (if we seek an ARIMA (p,1,q) predictor).

We now adopt a general notation to cover these two special cases and in so doing cover a wider class with no additional encumbrances. To this end let V_τ denote what is to be predicted at time τ. Thus if we seek a one-step predictor for an ARIMA(p,0,q) process, V_τ would denote $Z_{\tau+1}$. For a one-step predictor for an ARIMA (p,1,q) process V_τ would stand for $W_{\tau+1}$ (where $W_t = Z_t - Z_{t-1}$). For an ℓ-step predictor for an ARIMA (p,0,q) process, V_τ could denote $Z_{\tau+\ell}$, and so on. We denote the value of V_τ as predicted at time τ (when V_τ is not yet known) by $\hat{V}_\tau$ so that for a one-step ARIMA (p,1,q) predictor, $\hat{V}_\tau$ represents $\hat{W}_{\tau+1}$, etc. Now $\hat{V}_\tau$ is to be constructed as a linear combination of quantities already known at time τ:

A6) $$\hat{V}_\tau = \sum_{k=1}^{L} C_k X_{k,\tau}$$

where the C_k (k=1, ..., L) are constants and the $X_{k,\tau}$ are constructed from $\hat{V}_{\tau-j}$ $(j > 0)$ i.e., past values of $\hat{V}_\tau$ and from quantities $Y_{k,\tau}$ (k=1, ..., L) available at time τ which do not depend (explicitly at least) on the $\hat{V}_{\tau-j}$, specifically:

A7) $$X_{k,\tau} = \begin{cases} Y_{k,\tau} & \text{for } k=1, \ldots, j \\ \hat{V}_{\tau-k+j} + Y_{k,\tau} & \text{for } k = j+1, \ldots, L \end{cases}$$

To fix the ideas, note that for the one-step ARIMA(p,0,q) predictor of equation A5 we have (with $\tau+1 = t$):

$$\hat{V}_\tau = \hat{Z}_{\tau+1}, \quad J=p, \quad L=p+q$$

$$\left.\begin{aligned} C_k &= \phi_k \\ Y_{k,\tau} &= Z_{\tau-k+1} \end{aligned}\right\} \quad \text{for } k = 1, \ldots, J$$

$$\left.\begin{aligned} C_{k+J} &= \theta_k \\ Y_{k,\tau} &= \hat{Z} - Z_{\tau-k+J} \end{aligned}\right\} \quad \text{for } k = J+1, \ldots, L$$

and so

$$X_{k,\tau} = \begin{cases} Z_{\tau-k+1} & \text{for } k = 1, \ldots, J \\ \hat{Z}_{\tau-k+J} - Z_{\tau-k+J} & \text{for } k = J+1, \ldots, L \end{cases}$$

With this substitution equation A6 represents equation A5. It is evident that equation A6 can represent much more general predictions including even the case where some of the $Y_{k,\tau}$ are not linear in the values of the time series being predicted (i.e., not previous values of V_τ or linear combinations thereof) as in the general ARIMA case, but are instead perhaps values of a completely separate time series, or perhaps non-linear in the predicted series. Equations A6 and A7 will serve to represent as general a class of predictors as we shall have occasion to use. They are general enough to represent the general optimal ARIMA(p,d,q) one-step predictor. As noted, they can represent more general multichannel situations. A simple generalization would make them representative of the general ARIMA (p,d,q) ℓ-step predictor, but the extra encumbrance in notation will not be added as this feature is not to be used. We recall that B-J fit only one model for predictors of all lags (i.e., 1-step, 2-step, etc.) and derive the predictors from the model. As we have demonstrated, the one-step predictor corresponds directly to the model. As a result, in the matter of the forms available for predictor fit, the present formulation is currently more general than that of B-J.

Parenthetically it may be noted that the ARIMA(p,d,q) model (which contains ARIMA(p'd'q), where p'+d' = p, as a degenerate special case) is produced by passing white noise through a filter with p poles and q zeros. If the white noise is represented by Σ_t then

$$Z_t = F_p Z_{t-1} + F_q \Sigma_{t-1} + \Sigma_t$$

where $F_p Z_{t-1}$ represents a linear combintion of Z_{t-1} and its past values:

$$F_p Z_{t-1} = \sum_{k=1}^{p} f_k Z_{t-k}.$$

Then the one-step predictor is

$$\hat{Z}_t = F_p Z_{t-1} + F_q(Z_{t-1} - \hat{Z}_{t-1})$$

$$= (F_p + F_q) Z_{t-1} - F_q \hat{Z}_{t-1}$$

which is a linear filter with q poles, i.e., the predictor has as many poles as the model has zeros.

Proceeding with the general program of optimization of A6 with respect to C_k we note:

A9) $$\frac{\partial \hat{V}_\tau}{\partial C_k} = X_{k,\tau} + \sum_{j=J+1}^{L} C_j \frac{\partial \hat{V}_{\tau-j+J}}{\partial C_k}$$

In vector notation:

A10) $$\hat{V}_\tau = \underline{C}^T \underline{X}_\tau$$

A11) $$\frac{\partial \hat{V}_\tau}{\partial \underline{C}} = \underline{X}_\tau + \sum_{j=1}^{L-J} C_{j+J} \frac{\partial \hat{V}_{\tau-j}}{\partial \underline{C}}$$

Also note:

A12) $$\hat{V}_\tau = \sum_{j=1}^{L} C_j Y_{\tau,j} + \sum_{j=1}^{L-J} C_{j+J} \hat{V}_{\tau-j}$$

which has all terms depending on $\hat{V}$ collected in the second sum. This is of the form of a recursive discrete time filter with inputs $y_{k,\tau}$ and poles determined by C_{j+J}, j=1, L-J.

We optimize the filter by constructing an error function and minimizing it with respect to $\underline{C}$.

Let

A13) $$E_t = (1-R) \sum_{\tau=0}^{t} R^{t-\tau}(\hat{V}_\tau - V_\tau)^2$$

where $0 < R < 1$

Now $$\frac{\partial E_t}{\partial C_k} = -2(1-R) \sum_{\tau=0}^{t} R^{t-\tau}(\hat{V}_\tau - V_\tau)\frac{\partial \hat{V}_\tau}{\partial C_k}$$

Setting $\frac{\partial E_t}{\partial C_k} = 0$, we obtain

A14) $$\sum_j - A_{k,j} C_j = B_k$$

where

A15) $$A_{k,j} = (1-R) \sum_{\tau=0}^{t} R^{t-\tau} X_{k,\tau} X_{j,\tau}$$

A16) $$B_k = (1-R) \sum_{\tau=0}^{t} R^{t-\tau}[(V_\tau - \hat{V}_\tau)\frac{\partial \hat{V}_t}{\partial C_k} + \hat{V}_\tau X_{k,\tau}]$$

In vector notation $\underline{A} = \{A_{k,j}\}$ $\underline{B} = \{B_k\}$

and A14 becomes

A17) $$\underline{A}_t \underline{C} = B_t$$

Letting

A18) $$\underline{\eta}_\tau = \hat{V}_\tau \underline{X}_\tau + (V_\tau - \hat{V}_\tau)\frac{\partial \hat{V}_\tau}{\partial \underline{C}}$$

and letting

A19) $\underline{P}^{-1} = \underline{A}$ i.e., $\underline{AP} = I$

we have:

A20) $\underline{P}_\tau^{-1} = R\underline{P}_{\tau-1}^{-1} + (1-R)\underline{X}_\tau \underline{X}_\tau^T$

A21) $\underline{B}_\tau = R\underline{B}_{\tau-1} + (1-R)\underline{\eta}_\tau$

Equation A17 shows clearly that $\underline{C}$ depends on τ. Therefore, we denote it by $\underline{C}_\tau$. However, $\hat{V}_\tau$ was to be computed using constant $\underline{C}$ for all τ. This requirement must be abandoned in order to use the current value of $\underline{C}_\tau$ to update $\underline{P}^{-1}$ and $\underline{B}$. Therefore, we set:

A22) $\hat{V}_\tau = \underline{C}_{\tau-1}^T \underline{X}_\tau$

since $\underline{C}_{\tau-1}$ is known when $\hat{V}_\tau$ must be computed to update $\underline{C}_\tau$. Now following P.C. Young (Ref. 3, the goal here is rather different) (A20) leads to:

$$(A23)\quad \underline{P}_\tau = \frac{1}{R}\underline{P}_{\tau-1} - \frac{1-R}{R}\frac{\underline{P}_{\tau-1}\underline{X}_\tau \underline{X}_\tau^T \underline{P}_{\tau-1}}{R+(1-R)\underline{X}_\tau^T \underline{P}_{\tau-1}\underline{X}_\tau}$$

or letting

$\underline{\xi}_\tau = \underline{P}_{\tau-1}\underline{X}_\tau$, we get,

$$A24)\quad \underline{P}_\tau = \frac{1}{R}\underline{P}_{\tau-1} - \frac{1-R}{R}\frac{\underline{\xi}_\tau \underline{\xi}_\tau^T}{R+(1-R)\underline{X}_\tau^T \underline{\xi}_\tau}$$

and (A14, A20-A23) yield

$$A25)\quad \underline{C}_\tau = \underline{C}_{\tau-1} + (1-R)\underline{P}_\tau \underline{\eta}_\tau - (1-R)\hat{V}_\tau \underline{P}_\tau \underline{X}_\tau$$

or

A26) $\underline{C}_\tau = \underline{C}_{\tau-1} + (1-R)\underline{P}_\tau(V_\tau - \hat{V}_\tau)\frac{\partial \hat{V}_\tau}{\partial \underline{C}}$

Thus, to update $\underline{C}_{\tau-1}$ to $\underline{C}_\tau$, calculate $\frac{\partial \hat{V}_\tau}{\partial \underline{C}}$ by equation (A11) using $\underline{C}_{\tau-1}$ for explicit appearances of $\underline{C}$. Thus:

A26a) $\frac{\partial \hat{V}_\tau}{\partial \underline{C}} = \underline{X}_\tau + \sum_{j=1}^{L-J} C_{j+J,\tau-1}\frac{\partial \hat{V}_{\tau-j}}{\partial \underline{C}}$

Then calculate $\hat{V}_\tau$ from equation A22, $\underline{P}_\tau$ from equation A23, and finally $\underline{C}_\tau$ from equation A26. (Note that V_τ may not be available until _after_ time τ, which merely means that C_τ cannot _actually_ be calculated until that time. This presents no problem.)

Now reconsider equation A26 (the reader may skip to the continuation of the more practical discussion at the bottom of page 92 if desired)

$$\underline{C}_\tau = \underline{C}_{\tau-1} + (1-R)\underline{P}_\tau(V_\tau - \hat{V}_\tau)\frac{\partial \hat{V}_\tau}{\partial \underline{C}}$$

Let $\underline{C}_\tau - \underline{C}_{\tau-1}$ be denoted by $\delta\underline{C}_\tau$

Note that

$$\frac{\partial}{\partial \underline{C}}(V_\tau - \hat{V}_\tau)^2 = -2(V_\tau - \hat{V}_\tau)\frac{\partial \hat{V}_\tau}{\partial \underline{C}}$$

and replace

$\frac{1-R}{2}$ by λ and $\underline{P}_\tau$ by $\underline{Q}_\tau^{-1}$,

(temporary replacements for the sake of freedom in the ensuing discussion). Now we have:

A27) $\delta\underline{C}_\tau = -\lambda\underline{Q}_\tau^{-1}\frac{\partial}{\partial \underline{C}}(V_\tau - \hat{V}_\tau)^2$

$\underline{Q}_\tau$ will be assumed to be a positive definite matrix (as $\underline{P}_\tau^{-1}$ is) and λ will be assumed to be a sufficiently small positive quantity (as $\frac{1-R}{2}$ is).

We shall now show that if $\underline{Q}_\tau$ is slowly varying enough (which surely can be provided if (1-R)/2 is actually smaller than λ which is the case if (1-R)/2 is very small and λ=(1-R/2) and λ is small enough, then equation A27 ensures that $\underline{C}_\tau$ converges as $\tau \to \infty$.
Multiplying equation A27 by

$$\frac{-1}{\lambda}\delta\underline{C}_{\tau'}^{T}\underline{Q}_\tau^{-1}$$

We get:

A28) $$\frac{-1}{\lambda}\delta\underline{C}_{\tau'}^{T}\underline{Q}_\tau\delta\underline{C}_\tau = \delta\underline{C}_{\tau'}^{T}\frac{\partial}{\partial\underline{C}}(V_\tau-\hat{V}_\tau)^2$$

Let us assume that $\underline{Q}_\tau$ is essentially unchanged over a time period from τ_1 to τ_2. Now sum τ' between τ_3 and τ_4 and sum τ between τ_1 and τ_2.

We get:

A29) $$\frac{-1}{\lambda}(\underline{C}_{\tau_4} - \underline{C}_{\tau_3})^T\underline{Q}_{\tau_1}(\underline{C}_{\tau_2} - \underline{C}_{\tau_1}) \approx (\underline{C}_{\tau_4} - \underline{C}_{\tau_3})^T\frac{\partial}{\partial\underline{C}}\sum_{\tau=\tau_1}^{\tau_2}(V_\tau - \hat{V}_\tau)^2$$

Now suppose that the sequence

$$V_\tau;\ \tau = 1,2,3,\ldots,\infty$$

is the infinite repetition of a finite sequence of length T_M.

Thus $Z_{\tau+T_M} = Z_\tau$.

(The discontinuity that this introduces at Z_{T_M+1}, Z_{2T_M+1}, etc., can be eliminated as shown below or ignored if T_M is large enough as it is in the case of our data.) Under these circumstances of recycling data

$$\hat{V}_{\tau+nT_M} = \hat{V}_\tau (1+O(n\lambda T_M)$$

where n is an integer. (The notation "O(U)" denotes an indefinite quantity of order of magnitude U. If $\xi = O(U)$ and $\delta > 0$, then $\lim_{U\to 0} \frac{\Sigma^{1+\delta}}{U} = 0$.)

In the same vein we have:

$$\text{A30)} \quad \delta\underline{C}_{\tau+nT_M} = \delta\underline{C}_\tau (1+0(n\lambda T_M))$$

Thus if $\tau_3 = \tau_1 + nT_M$, $\tau_4 = \tau_2 + nT_M$

we have:

$$\text{A31)} \quad \frac{-1}{\lambda}(\underline{C}_{\tau_2} - \underline{C}_{\tau_1})^T \underline{Q}_{\tau_1} (\underline{C}_{\tau_2} - \underline{C}_{\tau_1})(1+0(n\lambda T_M))$$

$$= (\underline{C}_{\tau_4} - \underline{C}_{\tau_3})^T \frac{\partial}{\partial \underline{C}} \sum_{\tau=\tau_1}^{\tau_2} (V_\tau - \hat{V}_\tau)^2$$

If $n\lambda T_M$ is small enough the left hand side is obviously negative ($\underline{Q}$ is positive definite). The right hand side is approximately the difference in

$$\sum_{\tau=\tau_1}^{\tau=\tau_2} (V_\tau - \hat{V}_\tau)^2$$

induced by changing $\underline{C}_\tau$ (everywhere) by the change in $\underline{C}_\tau$ going between τ_3 and τ_4.

Thus

$$\Delta n \sum_{\tau=\tau_1}^{\tau=\tau_2} (V_\tau - \hat{V}_\tau)^2 < 0$$

where Δn indicates the change induced by changing $\underline{C}_0$ by $\underline{C}_{\tau_4} - C_{\tau_3}$ and where $\tau_4 - \tau_3 = \tau_2 - \tau_1$ and $\tau_3 - \tau_1 = nT_M$.

As a result: for λ small enough,

$$e_\tau = \sum_{\tau'=\tau}^{\tau'=\tau+T_M} (V_\tau - \hat{V}_\tau)^2$$

decreases as τ goes from τ_1 to $\tau_1 + T_M$ at least up to some point in time $\tau_1 = t(\lambda)$. If a point is reached were $e_\tau + mT_M$ does not decrease as m goes from m to m+1, then one of the approximations such as equation A30 has broken down. This can only happen if $\lambda T_M \hat{V}_{\tau'}$ or $\lambda T_M \frac{\partial \hat{V}_{\tau'}}{\partial \underline{C}}$ has become too large for some $\tau' < t(\lambda)$. Thus, either $\lambda T_M \hat{V}_{\tau'} > \delta$ or $\lambda T_M \frac{\partial \hat{V}_{\tau'}}{\partial \underline{C}} > \delta$ $(\delta > 0)$. Suppose $\hat{V}_\tau > \frac{\delta}{\lambda T_M}$ then

$$\sum_{\tau''=\tau-T_M/2}^{\tau''=\tau+T_M/2} (V_{\tau''} - \hat{V}_{\tau''})^2 > \frac{1}{2} \frac{\delta^2}{\lambda^2 T_M^2}$$

This is unbounded as $\lambda \to 0$ contradicting the requirement that c_τ always decrease up to $\tau = t(\lambda)$. Therefore, assume

$$\frac{\partial \hat{V}_\tau}{\partial \underline{C}} > \frac{\delta}{\lambda T_M}$$

Now equations A11 and A12 show that $\hat{V}_\tau$ and $\partial\hat{V}_\tau/\partial C$ are outputs of recursive filters which become unstable together:

$$\hat{V}_\tau = F_1\underline{Y}_\tau + F_2\hat{V}_\tau$$

$$\frac{\partial\hat{V}_\tau}{\partial\underline{C}} = F_3\underline{Y}_\tau + F_4\hat{V}_\tau + F_2\frac{\partial\hat{V}_\tau}{\partial\underline{C}}$$

where F_1, F_2, F_3, F_4 represent linear combinations of the given quantitites with backwards shift i.e., linear filters wihout poles (bounded).

$$\hat{V}_\tau = \frac{I}{I-F_2} F_1\underline{Y}_\tau$$

$$\frac{\partial\hat{V}_\tau}{\partial\underline{C}} = \frac{I}{I-F_2}(F_3\underline{Y}_\tau + F_4\hat{V}_\tau)$$

As a result $\partial\hat{V}_\tau/\partial\underline{C}$ can become unbounded only as $\hat{V}_\tau$ becomes unbounded or $I/I-F_2$ becomes unstable. But in that case $\hat{V}_\tau$ also becomes unbounded. In order for $\partial\hat{V}_\tau/\partial\underline{C}$ to become larger than $r = \delta/\lambda T_M$ we must, therefore, have that V_τ becomes larger than r' and r' goes to infinity as r does -- leading again to the same contradiction.

Thus if A26 is altered to $\underline{C}_\tau = \underline{C}_{\tau-1} + 2\lambda\underline{P}_\tau(V_\tau - \hat{V}_\tau)\partial V_\tau/\partial\underline{C}$ where 2λ replaces 1-R then $\underline{C}_\tau$ converges providing λ is small and 1-R is considerably smaller.

Practically the algorithm runs thusly:

Initialize $\underline{C}_o$ to reasonable values.

Initialize $\underline{P}_o$ to $K\underline{I}$ where K is a large number. Update $\underline{P}_\tau$ until reasonably converged by equation A23 (a quick process). Then update $\underline{C}_\tau$ by A26 and $\underline{P}_\tau$ by A23 using the same R for both. The value of R is allowed to change with τ:

$$\theta_\tau = 1 - R = \frac{a_1}{a_2+\tau}$$

In the final values of $\underline{P}_\tau^{-1}$ and $\underline{B}_\tau$ the contributions at time τ (which are $\underline{X}_\tau \underline{X}_\tau^T$ and $\underline{n}_\tau$ respectively) enter with the factor $F_{t,\tau}$ where

$$F_{t,\tau} = (1-R_\tau) \prod_{\tau'=\tau}^{t} R_{\tau'}$$

$$\log F_{t,\tau} \approx \log \theta_\tau - \sum_{\tau'=\tau}^{t} \theta_{\tau'}$$

$$\log F_{t,\tau} \approx \log \left(\frac{a_1}{a_2+\tau}\right) - a_1 \log \left(\frac{a_2+t}{a_2+\tau}\right)$$

or

$$F_{t,\tau} \approx \frac{a_1}{a_2+\tau} \left(\frac{a_2+\tau}{a_2+t}\right)^{a_1}$$

The factors with which the initial values of $\underline{P}_\tau^{-1}$ and of $\underline{B}_\tau$ enter into the final value is similarly

$$f_{t,\tau} = \left(\frac{a_2+\tau}{a_2+t}\right)^{a_1}.$$

This is, therefore, approximately the weight of $\underline{C}_0$ in $\underline{C}_t$. If there were no recycling and all new data were being presented then for large τ, $\delta\underline{C}_\tau$ would have a relative variance going roughly as $1/\tau$ from its true value. Therefore, it is reasonable to want $F_{t,\tau}$ to go as $\tau+a_2$ and therefore to take $a_1=2$. If $a_1=1$ then all contributions to $\underline{C}_t$ are weighted equally in the final result.

This might be better for recycled data when convergence is very nearly complete, but to get quick convergence a_1 should be considerably greater than 2, say $a_1=5$, is used initially ($a_2 \tilde{} T_M$ is a good choice) and then eventually $a_1 \geq 2$ is used.

The criterion for convergence is that the small changes in $\delta \underline{C}_\tau$ are not leading to a persistent change.

Thus form:

$$\Gamma_\tau = R\Gamma_{\tau-1} + \underline{P}_\tau^{-1}\delta\underline{C}_\tau = R\underline{\Gamma}_{\tau-1} + (1-R)(V_\tau - \hat{V}_\tau)\frac{\partial \hat{V}_\tau}{\partial \underline{C}}$$

$$D_\tau = \Gamma_\tau^T \underline{P}_\tau \Gamma_\tau$$

$$d_\tau = Rd_{\tau-1} + \delta\underline{C}^T \underline{P}_\tau \delta\underline{C} = Rd_{\tau-1} + (1-R)(V-\hat{V}_\tau)\delta\underline{C}_\tau^T \frac{\partial \hat{V}_\tau}{\partial \underline{C}} .$$

We may recognize 3 cases for the statistical properties of $\delta\underline{C}_\tau$ as reflected in D_τ/d_τ.

1. If $\underline{C}_\tau$ has converged essentially to its true value but new independent data is being fed in (no recycled data) then the $\delta\underline{C}_\tau$'s will be independent and the limiting expected value for D_τ/d_τ (when 1-R is very small) is 1/2. It will fluctuate a great deal and one should wait for an especially small value to stop the iteration.

2. If we are using recycled data then as $\underline{C}_\tau$ converges, D_τ/d_τ will approach zero (for T_M large). It will fluctuate considerably and a quite small value is best used as the stopping place.

3. If $\delta\underline{C}_\tau$ is largely biased, i.e., has a strong constant component (even though $\delta\underline{C}_\tau$ is absolutely small) as would occur if $\underline{C}_\tau$ were slowly but steadily drifting towards a substantially different

value, then (when 1-R is very small) D_τ/d_τ will become actually much larger than 1.

From these considerations we see that, as we are dealing with recycled data, we should not stop the iteration unless D_τ/d_τ is small. $D_\tau/d_\tau < .02$ should be a minimum requirement when $T_M = 5000$ as it does in this study. In general it would seem that $D_\tau/d_\tau < 1/\sqrt{T_M}$ should be required.

We mentioned earlier that the discontinuity introduced by having Z_1 follow Z_{T_M} was not of much significance for us since $T_M \approx 5000$.

The sequence of values is $Z_1, Z_2, Z_3, \ldots, Z_{T_M-1}, Z_{T_M}, Z_1, Z_2, \ldots, Z_{2T_M}, Z_1$. The discontinuity which occurs between Z_{T_M} and Z_1 is not only in Z_τ but in all its derivatives. All these discontinuities can be eliminated by using the infinite sequence:

$Z_1, Z_2, Z_3, \ldots, Z_{T_{M-1}}, Z_{T_M}, 2Z_{T_M} - Z_{T_{M-1}}, 2Z_{T_M} - Z_{T_{M-2}}, \ldots,$

$2Z_{T_M} - Z_3, 2Z_{T_M} - Z_2, 2Z_{T_M} - Z_1, 2Z_{T_M} - 2Z_1 + Z_2, 2Z_{T_M} - 2Z_1 +$

$Z_3, \ldots, 3Z_{T_M} - 2Z_1, 4Z_{T_M} - 2Z_1 - Z_{T_{M-1}}, \ldots$

Let the value of this sequence be denoted by S_j, i.e.:

$S_1 = Z_1, S_2 = Z_2, \ldots, S_{T_{M+1}} = 2Z_{T_M} - Z_{T_{M-1}}, \ldots,$ etc.

Let $j = (2L+K)T_M+J$

where $L \geq 0,\ 0 \leq J \leq T_M$,

and K is either 0 or 1. It can be seen that for $j > 0$ there is just one way of representing j by this formula with J, K, L obeying these restrictions. The values of the sequence are determined thus:

$$S_j = 2(L+K)Z_{T_M} - 2LZ_1 + (-1)^K Z_I$$

where

$$J = j\text{Mod}(T_M)$$

$$K = [(j-J)/T_M]\text{Mod}(2)$$

$$2L = (j-J)/T_M - K$$

$$I = (-1)^K J + KT_M$$

Expressed in informal Fortran:

```
              j = 0
     DO 100   L = 1,∞
     DO 100   K = 1,2
     DO 100   J = 1,T_M
              j = j+1
              I = (-1)**K*J+K*T_M
    100    S(j) = 2*(L+K)*Z(T_M)-2*L*Z(1)
                    + (-1)**K*Z(I)
```

If the trend from Z_1 to Z_{T_M} is first removed by letting

$$U(J) = Z(J) - Z(1) - \frac{J-1}{T_{M-1}} [Z(T_M) - Z(1)] \qquad J = 1,T_M$$

Then setting

$$S_j = (-1)^K U(I)$$

yields an S_j without a trend.

This means of eliminating the discontinuity is in keeping with the ARIMA formulation which is indifferent to the direction of change. For a highly non-linear predictor which fits an increasing trend as well as accounting for local dips (or peaks), the lack of distinction between "up" and "down" could be a problem. However, with the right formulation this would not be a problem and the "double mirror" means of recycling can be used in the traffic prediction context when the sample length is so short that the discontinuities become a problem.

APPENDIX B

SAMPLE RUN AND PROGRAM FOR OPTIMIZING ARIMA PREDICTORS

Annotation of sample run: Historical ARIMA(2,1,2) 1-step predictor for sensor 1.

1. 0,5240 - "0" means use historical data ("1" would mean non-historical), "5240" means read 5240 records (i.e. 5 minute intervals) from the date file
2. 100,100 - "100" means initialize K to 100 I (Appendix A)
3. MANDY - name of data file = first half of Toronto data
4. 1 - use sensor 1
5. Y2AVE - Read historical data from file containing averages over second half of Toronto data
6. 5239 - Use first 5239 records (from) MANDY as the traffic data
7. 2 - Print out first 2 records
8. .79,0,.79,0,0,0 - Initialize $C_{1,0}$ to - .79, $C_{2,0}$ = 0, $C_{3,0}$ = .79 etc. (Appendix A)
9. 9,1 - "9" is a control parameter - means standard run, "1" means 1-step predictor ("2" for 2-step or "3" for 3-step etc. may be used)
10. 3,4 - means three points after data gap before predictor is working normally, 4 points after data gap before error statistics are updated.

11-12. Two sets of θ,λ for running two fixed second generation predictors side by side with ARIMA(2,1,2) being optimized by this program.

These ARIMA (1,1,1) are for comparison

13. 2000,9 - This part of the run will iterate through 2000 points. (If 5239 were less than 2000 then when 5239 was reached the program would go back to the beginning of the traffic data.) The program will not work with the first 9 points on the data file.

14. 2,1 - $a_1 = 2$ in $\theta\tau = \frac{a_1}{a_2+\tau}$ (Appendix A.) "1" has no important meaning here (value 1 for all normal use)

15. 40,2000 - $a_2 = 2000$; 40 determines a wild point selector. If square of the difference in two successive values of Z_τ exceeds the mean square value by a factor of 40, the point is considered a wild point and treated the same as a data gap.

16. 2,2 - p = 2, q = 2
This program has d=1 built in (a minor modification of the program is used when d=0 is desired). Thus, an ARIMA(2,1,2) predictor is to be optimized.

17. 0,0 - $\lambda = 0 \times (1-R)$ therefore $\underline{C}_\tau$ is not being updated on this part of the run. When "1,1" is entered, $\underline{C}_\tau$ and $\underline{P}_\tau$ are being updated together.

18-19-20. Show mean square prediction errors (exponential moving averages) for the two ARIMA(1,1,1) predictors and the ARIMA(2,1,2) predictor being optimized. 79.66555 is the value of the latter. 124.6547 is the mean square error of the "predictor" which predicts that the next value will be the same as the current value.

21. This line gives C_1, C_2, C_3, C_4 at time t=8,000 (at the end of this part of the run). (the equivalent of a_o, a_1, b_o, b_1 of Equation 5.

$$Z_{\tau+1} = C_1 Z_\tau + C_2 Z_{\tau-1} + C_3 \hat{Z}_\tau + C_4 \hat{Z}_{\tau-1}$$

22. This is the value of d_τ at τ = 8000 (Appendix A)

($\frac{2d_\tau}{\theta_\tau T_M}$ also estimates the optimization bias for statistical significance).

23. This is the value of D_τ at τ = 8000 (Appendix A).

24. These are the values of $\underset{\sim}{P}_\tau$ at τ = 8000.

Note that in this case the mean square error for the ARIMA(1, 1,1) was 79.95 (it was previously optimized) and for the ARIMA(2,1.2) it was 79.66. The improvement in this case was much less than 1% going from ARIMA(1,1,1) to ARIMA(2,1,2) As noted in Section 3.0 this improvement could be as high as 3% at some sensors, yielding a 1.5% improvement in the RMS error.

ANNOTATED COMPUTER RUN OF PROGRAM FOR OPTIMIZING PREDICTOR

NOTE: Underline indicates input

```
.EX
LINK:    LOADING
[LNKXCT NLB EXECUTION]
0,53\3\240     (1)
               0                5240
100,100   (2)
MANDY   (3)
1    (4)
               1
Y2AVE (5)
5239 (6)
           5248
           5239
2  (7)
               2
               3                  10                 40
               4                   7                 40
  0.2880000E+03   0.4014286E+02   0.4142857E+02
  0.7557143E+02   0.5990000E+03   0.5654167E+02
-.79                (8)
       -0.7900000
.\.\
        0.0000000
.79
        0.7900000
        0.0000000
        0.0000000
        0.0000000
        0.0000000
```

ANNOTATED COMPUTER RUN OF PROGRAM FOR OPTIMIZING PREDICTOR

```
9,1              (9)
3,4                 (10)
                   3                 4
.79,.79         (11)
       0.7900000         0.7900000
.79,.74         (12)
       0.7900000         0.7400000
2000,9          (13)
            2000                 9
2,1           (14)
40,2000          (15)
2,2           (16)
       2.0000000         1.0000000
      40.0000000      2000.0000000
0,0          (17)
       0.0000000         0.0000000
               2                 2
      -1.0000000         0.0000000
       0.0000000
      0.1131619E+03    0.4995720E+02
      0.5024261E+02   -0.8733333E+01
      0.4995720E+02    0.4916134E+02
       -0.7900000          0.0000000           0.7900000          0.0000000
    -0.4951133E+00   -0.1285714E+01
      0.0000000E+00    0.0000000E+00
                  50                47                  53                 0
       0.0190730         0.0105988         -0.0058060         -0.0113552
       0.0105988         0.4549629          0.5651731         -0.4553450
      -0.0058060         0.5651731          0.7517604         -0.5647248
      -0.0113552        -0.4553450         -0.5647248          0.4868848
```

ANNOTATED COMPUTER RUN OF PROGRAM FOR OPTIMIZING PREDICTOR

```
9,1
3,4
              3                    4
.79,.74
    0.7900000          0.7400000
.79,.75
    0.7900000          0.7500000
5000,9
           5000                    9
5,1
40,5000
2,2
    5.0000000          1.0000000
   40.0000000       5000.0000000
1,1
    1.0000000          1.0000000
              2                    2
   -4.0000000       2000.0000000
    1.0000000
   0.1222902E+03   0.7719514E+02
   0.7719889E+02   0.1988000E+02
   0.7685371E+02   0.7702872E+02
    -0.7636296        -0.5376476        0.1407914        0.5932268
   0.3580626E+01  -0.1324000E+02
   0.7276974E+00   0.1618651E+00
            11               104              166                0
     0.0130978       0.0032521       -0.0091740       -0.0030558
     0.0032521       0.1781594        0.2331201       -0.1931154
    -0.0091740       0.2331201        0.3431113       -0.2541893
    -0.0030558      -0.1931154       -0.2541893        0.2340218
```

ANNOTATED COMPUTER RUN OF PROGRAM FOR OPTIMIZING PREDICTOR

```
9,1
4,5
                    4                    5
.79,.74
            0.7900000            0.7400000
.79,.76
            0.7900000            0.7600000
7000,9
                 7000                    9
3,1
40,7000
2,2
            3.0000000            1.0000000
           40.0000000         7000.0000000
1,1
            1.0000000            1.0000000
                    2                    2
           -2.0000000         6000.0000000
            1.0000000
        0.1222555E+03        0.7764185E+02
        0.7762232E+02        0.8125000E+01
        0.7728594E+02        0.7728835E+02
           -0.7502273           -0.5597880            0.1047144            0.6092877
        0.1653543E+01       -0.5000000E+01
        0.3[illegible]14145E+00        0.1580786E-01
                  155                  139                  207                    0
            0.0129188            0.0029251           -0.0091783           -0.0028387
            0.0029251            0.2425253            0.3161746           -0.2675341
           -0.0091783            0.3161746            0.4485773           -0.3508445
           -0.0028387           -0.2675341           -0.3508445            0.3188768
```

ANNOTATED COMPUTER RUN OF PROGRAM FOR OPTIMIZING PREDICTOR

```
9,1
4,5
                    4                    5
.79,.76
      0.7900000          0.7600000
.79,.77
      0.7900000          0.7700000
8000,9
                 8000                    9
2,1
40,8000
2,2
      2.0000000          1.0000000
     40.0000000       8000.0000000
1,1
      1.0000000          1.0000000
                    2                    2
     -1.0000000       7000.0000000
      1.0000000
     0.1246547E+03    0.7995104E+02 (18)
(19) 0.7997197E+02    0.1200000E+02
(20) 0.7966555E+02    0.7966625E+02
      -0.7434935         -0.5305944          0.1124870          0.5924631 (21)
    -0.1673080E+01   -0.2785714E+01
(22) 0.2337735E+00    0.4352359E-02 (23)
                      196                 171                 293                   0
        0.0125531         0.0027167          -0.0089580          -0.0023742
        0.0027167         0.2540377           0.3335998          -0.2801077
       -0.0089580         0.3335998           0.4735075          -0.3702793 (24)
       -0.0023742        -0.2801077          -0.3702793           0.3322131
K/^C
```

```
.TYPE NLB
        COMMON/CNN/ETA(12,12),ZSS(5),ICS,DC(10)
        COMMON /CMD/ ITME(5255),ISEN(5255),KN,HIZ(300),IUU,IUUU,ITA,JX
        COMMON/CMJ/RAS,IJF,IC,ID
        COMMON/CMA/A(10,10),C(10),CA(10),R,R3,R2
        COMMON/CMB/X(10),ZHA,ZHB,Z,NP,NQ,NT
        COMMON/CMC/SA,SB,SC,Z1,Z2,Z3,IX,A1,A2,A3,A4,A5,A6
        EAY=0
        ZHA=0.
        SC=0.
        ACCEPT 4,IT,IX
        TYPE 4,IT,IX
        JX=IT
        KN=IX
        Z=0
        Z1=0.
        ZAE=0.
        Z2=0.
        ACCEPT 5,A1,A2
        A5=A2
4       FORMAT(2I)
5       FORMAT(2F)
        NT=10
        AQA=100.
        AQP=0.
        EAB=0.
        EAA=0.
        RAS=1
76      CONTINUE
        SB=0.
        DO 347 IPP=1,300
347     HIZ(IPP)=0.
        CALL TORIN
        NT=7
        DO 100 I=1,NT
        ACCEPT 313,CA(I)
        TYPE 313,CA(I)
        X(I)=0.
        C(I)=CA(I)
        DO 99 J=1,NT
        ETA(I,J)=0.
        A(I,J)=0.
99      CONTINUE
        A(I,I)=A1
100     CONTINUE
        DO 66 ITS=1,20
        ACCEPT 4,IST,ISU
        IF(IST.EQ.0) GO TO 478
```

```
313     FORMAT(1F)
        ACCEPT 4,IDQ,IE
        TYPE 4,IDQ,IE
        ACCEPT 5,A1,A2
        TYPE 5,A1,A2
        ACCEPT 5,A3,A4
        TYPE 5,A3,A4
        ICZ=0
        ACCEPT4,IT,IJF
        TYPE 4,IT,IJF
        ACCEPT 5,R,R2
        ACCEPT 5,R3,R6
        ACCEPT 4,NF,NQ
        TYPE 5,R,R2
        A6=R6
        TYPE 5,R3,R6
        ACCEPT 5,HDA,HDB
        A5=HDB
        TYPE 5,HDA,HDB
        TYPE 4,NF,NQ
        NT=NF+NQ
        IF(IST.EQ.6) NT=NT+1
        NPQ=4
        NPQ=IDQ
        R2S=R2
        RW=R3
        R3=R2
        RQQ=R
        RQ6=R6
        RC6=1.-R6
        RCE=1.-R
        ICE=0
        ICR=0
        ICS=0
        ICT=0
        ICU=0
        ZAF=0.
        A4M=1.-A4
        JTT=0
478     TYPE 5,RCE,RTT
        IF(IST.NE.0) GO TO 468
        ACCEPT 4,IT,ISV
        TYPE 4,IT,JTT
        TYPE 6,RCE,RCX
468     CONTINUE
        DO 55 ITT=1,IT
        JTT=JTT+1
        IF(ITT.EQ.800) TYPE 5,R2
        RTT=JTT
        R2=HDA
        RCE=RQQ/(RTT+RQ6)
```

```
C        RCE=RCE*(1.+R2S*RCE)
         R=1.-RCE
         R6=R
         RC6=RCE
         ZAF=ZAF*A3+(1.-A3)*Z2
         IF(ICU.LE.2+ISU) ZAF=Z2
         ZAL=A4*ZAF+A4M*Z1
         ZAE=ZAE*A1+(1.-A1)*Z2
         IF(ICU.LE.2+ISU) ZAE=Z2
         ZAK=ZAE*A2+(1.-A2)*Z1
         IF((Z-Z1)**2.LT.AQA*RW) GOTO758
         ICZ=ICZ+1
         ICU=0
758      CONTINUE
         IF(ICU.GT.IE) AQS=AQS*R6+RC6*(Z-ZAL)**2
         IF(ICU.GT.IE) AQA=AQA*R6+RC6*(Z-Z1)**2
         IF(ICU.GT.IE) AQP=R6*AQP+RC6*(Z-ZAK)**2
         IF(ICU.GT.5)     EAB=EAB*R6+RC6*(Z-Z1-ZHA)**2
         DD2=DD1
         DD1=(Z-Z1-ZHA)
         IF(ICU.GT.IE)EAW=EAW*A3+(1.-A3)*ABS(DD1)
         IF(ICU.GT.IE)EAY=EAY*A3+(1.-A3)*DD1
         IF(ICU.GT.IE)EAA=EAA*R6+RC6*(Z-Z1-ZHA)**2
         Z3=Z2
         Z2=Z1
         Z1=Z
         DO 656 JBJ=1,ISU
         ZN=ZSS(JBJ)
         ZSS(JBJ)=Z1
         Z1=ZN
656      CONTINUE
         Z1=ZSS(ISU)
         IC2=IC1
         IC1=IC
         CALL ZGEN
         ICE=ICE+IC
         ICR=ICR+IC*(1-IC1)
         ICS=ICS+IC
         ICS=ICS*IC
         ICT=ICT+ICS
         ICC=1-IC
         ICU=ICU+ICC
         ICU=ICU*ICC
         IF(ICU.LE.1) GO TO 55
         XQ=Z1-Z2
         X(NQ)=ZHA
         IF(ICU.LE.NPQ) X(NQ)=0.
         DO 47 I=1,NT
         XP=X(I)
         X(I)=XQ
         XQ=XP
```

```
47      CONTINUE
        IF(IST.EQ.6) X(NT)=(Z1-Z2-ZHA)*ABS(EAY)/EAW
        IF(ICU.LT.NPQ)GO TO 55
        Z=Z-Z1
        CALL PRED
        Z=Z+Z1
7       FORMAT(1X,4F)
6       FORMAT(3X,2E)
55      CONTINUE
        TYPE 6,AQA,ACP
        TYPE 6,AQS,X(2)
        TYPE 6,EAA,EAB
        TYPE 7,C(1),C(2),C(3),C(4)
        DCC=0
        DO 544 JBK=1,NT
        DO 544 JBJ=1,NT
544     DCC=DCC+DC(JBJ)*A(JBJ,JBK)*DC(JBK)
        TYPE 6,ZHA,Z
        TYPE6, DCS,DCC
        TYPE 87,ICE,ICR,ICT,ICZ
        TYPE 7,(((A(I,J),I=1,NT),J=1,NT))
87      FORMAT(2X,4I)
66      CONTINUE
        STOP
        END
        SUBROUTINE TORIN
        COMMON/CMJ/RAS,IJF,IC,ID
        COMMON /CMD/ ITME(5255),ISEN(5255),KN,HIZ(300),IUU,IUUU,ITA,JK
        INTEGER A,B,C,D
        ACCEPT 1,MANDY
1       FORMAT(A5)
        CALL IFILE (1,MANDY)
        ACCEPT 8,ID
        TYPE 8,ID
        IUU=1
        ACCEPT1,MINDY
        IUUU=0
        GO TO (11,12,13,14) ID
11      READ(1)K,(ITME(J),ISEN(J),A,B,C,D,J=1,KN)
        GO TO 15
12      READ(1)K,(ITME(J),A,ISEN(J),B,C,D,J=1,KN)
        GO TO 15
13      READ(1)K,(ITME(J),A,B,ISEN(J),C,D,J=1,KN)
        GO TO 15
14      READ(1)K,(ITME(J),A,B,C,ISEN(J),D,J=1,KN)
15      CONTINUE
        REWIND 1
        TYPE 8,K
        ACCEPT 8,ITA
        TYPE 8,ITA
```

```
      ACCEPT 8,IU
      TYPE 8,IU
      DO 45 IKK=1,IU
      IK=IKK+IJF
      TYPE 9,ITME(IK),ISEN(IK),D
45    CONTINUE
      CALL IFILE(1,MINDY)
      GO TO (21,22,23,24,25) ID+JX
21    READ(1)K,(E,HIZ(J),F,GH,G,H,J=1,288)
      GO TO 25
22    READ(1)K,(E,F,HIZ(J),GH,G,H,J=1,288)
      GO TO 25
23    READ(1)K,(E,F,GH,HIZ(J),G,H,J=1,288)
      GO TO 25
24    READ(1)K,(E,F,GH,G,HIZ(J),H,J=1,288)
25    CONTINUE
8     FORMAT(1I)
      REWIND 1
      TYPE 10 ,E,F,G,GH,H,HIZ(100)
10    FORMAT(3E)
9     FORMAT(3I)
      RETURN
      END
      SUBROUTINE ZGEN
      COMMON/CMJ/RAS,IJF,IC,ID
      COMMON /CMB/X(10),ZHA,ZHB,Z,NF,NT
      COMMON /CMD/ ITME(5255),ISEN(5255),KN,HIZ(300),IUU,IUUU,ITA,JX
      IC=1
      IUUU=IUUU+1
      IUU=IUU+1
      IF(IUU.GT.ITA)IUU=1
      IVA=ITME(IUU+IJF)
      IF(IVA.NE.IUUU)GO TO 98
      Z=ISEN(IJF+IUU)
      Z=Z-HIZ(IVA)
      IC=0
98    IUUU=IVA
      RETURN
      END
      SUBROUTINE PRED
      COMMON/CMC/SA,SB,SC,Z1,Z2,Z3,IX,A1,A2,A3,A4,A5,A6
      COMMON/CNN/ETA(12,12),ZSS(5),DCS,DC(10)
      COMMON/CMB/X(10),ZHA,ZHB,Z,NF,NQ,NT
      COMMON/CMA/A(10,10),C(10),CA(10),R,R3,R2
      DIMENSION V(10)
      ESQ=0
      DO 61 II=1,NT
      ETT=X(II)
      DO 62 JJ=1,NF
      ETT=ETT+R2*C(NQ+JJ)*ETA(II,JJ)
      ESP=ETA(II,JJ)
```

PREDICTOR OPTIMIZING PROGRAM

```
      ETA(II,JJ)=ESQ
      ESQ=ESP
62    CONTINUE
      ETA(II,1)=ETT
61    CONTINUE
      RCE=1.-R
      RCEE=RCE*A5
            RA=1.-RCEE*R2
      ANT=NT
      D=0.
      ZHN=0.
      DO 30 I=1,NT
      ZHN=ZHN+C(I)*X(I)
      V(I)=0.
      DO 20 J=1,NT
20    V(I)=V(I)+A(I,J)*X(J)
30    D=D+V(I)*X(I)
      RM=1./R
      RC=RM*(1.-R)/(R+D*(1.-R))
      ZHA=ZHN
      DES=0
      ZHB=0.
      DO 50 I=1,NT
      DET=0
      ZHB=ZHB+CA(I)*X(I)
      DO 40 J=1,NT
      A(I,J)=RM*A(I,J)-RC*V(I)*V(J)
      DET=DET+ETA(J,1)*(Z-ZHN)*A(I,J)*RCEE
      IF(J.GE.I) GO TO 40
      ASA=A(I,J)+A(J,I)
      ASA=.5*ASA
      A(I,J)=ASA
      A(J,I)=ASA
40    CONTINUE
      DEU=ETA(I,1)*(Z-ZHN)*RCEE
      DC(I)=DC(I)*RA+DEU
      DES=DES+DEU*DET
      C(I)=C(I)+DET
50    CONTINUE
      DCS=RA*DCS+DES
78    FORMAT(1X,5E)
      RETURN
      END
```

APPENDIX C

TABLES

TABLE C-1
Historical 1-Step

S	ALPHA	BETA	SQRT(AVE SQ)	AVE ABS	4TH RT(AVE 4TH)	OBS.
1	.7500	.700	8.918170	6.407750	13.373933	4718
1	.7500	.740	8.911419	6.397045	13.407196	4718
1	.7500	.800	8.928010	6.395170	13.502519	4718
1	.7500	.810	8.933824	6.396842	13.523646	4718
1	.7500	.840	8.956607	6.404543	13.595864	4718
1	.7500	.900	9.026267	6.431271	13.779209	4718
1	.7900	.700	8.911248	6.388400	13.422617	4718
1	.7900	.740	8.904199	6.385222	13.466391	4718
1	.7900	.800	8.920389	6.380790	13.579097	4718
1	.7900	.810	8.926202	6.381739	13.603289	4718
1	.7900	.840	8.948949	6.387129	13.684958	4718
1	.7900	.900	9.018102	6.411671	13.888164	4718
1	.8720	.700	8.940902	6.403469	13.611045	4718
1	.8720	.740	8.939222	6.387669	13.689379	4718
1	.8720	.800	8.964083	6.380241	13.858248	4718
1	.8720	.810	8.971408	6.380378	13.892241	4718
1	.8720	.840	8.998754	6.384056	14.003961	4718
1	.8720	.900	9.077651	6.408187	14.269690	4718
1	.8500	.700	8.924864	6.396599	13.547380	4718
1	.8500	.740	8.920649	6.380623	13.614622	4718
1	.8500	.800	8.941447	6.372389	13.765595	4718
1	.8500	.810	8.948067	6.372299	13.796479	4718
1	.8500	.840	8.973296	6.376347	13.898687	4718
1	.8500	.900	9.047664	6.400507	14.144714	4718
1	.8800	.700	8.948933	6.406968	13.636523	4718
1	.8800	.740	8.948465	6.392439	13.719640	4718
1	.8800	.800	8.975265	6.385886	13.895548	4718
1	.8800	.810	8.982925	6.385951	13.930761	4718
1	.8800	.840	9.011338	6.389661	14.046205	4718
1	.8800	.900	9.092325	6.414635	14.319628	4718
1	.9200	.700	9.017651	6.454628	13.794340	4718
1	.9200	.740	9.026914	6.444199	13.902251	4718
1	.9200	.800	9.069181	6.442757	14.118213	4718
1	.9200	.810	9.079511	6.443762	14.169321	4718
1	.9200	.840	9.116071	6.450809	14.296761	4718
1	.9200	.900	9.213902	6.484511	14.613273	4718
2	.7500	.700	11.229110	8.693364	15.478839	4718
2	.7500	.740	11.195423	8.663447	15.437317	4718
2	.7500	.800	11.178226	8.639856	15.423378	4718
2	.7500	.810	11.179264	8.639005	15.426740	4718
2	.7500	.840	11.189065	8.642143	15.446595	4718
2	.7500	.900	11.238649	8.674162	15.530175	4718
2	.7900	.700	11.166063	8.656202	15.435165	4718
2	.7900	.740	11.147995	8.621971	15.389349	4718
2	.7900	.800	11.123974	8.592759	15.368943	4718
2	.7900	.810	11.123649	8.590978	15.371232	4718
2	.7900	.840	11.130126	8.591085	15.387894	4718
2	.7900	.900	11.172531	8.618517	15.465226	4718

TABLE C-1
Historical 1-Step (Cont.)

2	.8720	.700	11.138772	8.612175	15.411513	4718
2	.8720	.740	11.096995	8.572086	15.365494	4718
2	.8720	.800	11.067361	8.537329	15.345656	4718
2	.8720	.810	11.066298	8.535046	15.348149	4718
2	.8720	.840	11.069765	8.533782	15.365623	4718
2	.8720	.900	11.106581	8.557453	15.445506	4718
2	.8500	.700	11.143862	8.618690	15.405843	4718
2	.8500	.740	11.102093	8.578821	15.359263	4718
2	.8500	.800	11.072394	8.544761	15.335673	4718
2	.8500	.810	11.071312	8.542753	15.337659	4718
2	.8500	.840	11.074704	8.542213	15.353528	4718
2	.8500	.900	11.111302	8.562956	15.429930	4718
2	.8800	.700	11.138944	8.611704	15.419571	4718
2	.8800	.740	11.097428	8.571616	15.371534	4718
2	.8800	.800	11.068237	8.536532	15.353365	4718
2	.8800	.810	11.067254	8.534154	15.356155	4718
2	.8800	.840	11.070970	8.532717	15.374575	4718
2	.8800	.900	11.108323	8.557141	15.456508	4718
2	.9200	.700	11.164670	8.626839	15.474817	4718
2	.9200	.740	11.127672	8.589367	15.439351	4718
2	.9200	.800	11.105807	8.560634	15.437045	4718
2	.9200	.810	11.106104	8.559425	15.442655	4718
2	.9200	.840	11.113756	8.561288	15.469794	4718
2	.9200	.900	11.159370	8.589860	15.570358	4718
3	.7500	.700	10.512254	8.041036	15.358666	4718
	.7500	.740	10.482845	8.013672	15.365916	4718
	.7500	.800	10.470038	7.993424	15.425652	4718
3	.7500	.810	10.471557	7.991997	15.441313	4718
3	.7500	.840	10.482431	7.992415	15.493034	4718
3	.7500	.900	10.532269	8.013404	15.654899	4718
3	.7900	.700	10.476994	8.013602	15.295761	4718
3	.7900	.740	10.444149	7.984377	15.301495	4718
3	.7900	.800	10.426032	7.959987	15.357947	4718
3	.7900	.810	10.426660	7.958060	15.373117	4718
3	.7900	.840	10.434795	7.956035	15.428464	4718
3	.7900	.900	10.479097	7.970921	15.582998	4718
3	.8720	.700	10.457772	7.998923	15.232992	4718
3	.8720	.740	10.424456	7.967403	15.242585	4718
3	.8720	.800	10.405911	7.940334	15.308809	4718
3	.8720	.810	10.406374	7.938209	15.325990	4718
3	.8720	.840	10.414355	7.934778	15.387429	4718
3	.8720	.900	10.458528	7.948643	15.556293	4718
3	.8500	.700	10.452593	7.995986	15.238266	4718
3	.8500	.740	10.418127	7.963492	15.244646	4718
3	.8500	.800	10.397598	7.934780	15.305098	4718
3	.8500	.810	10.397629	7.932608	15.321110	4718
3	.8500	.840	10.404787	7.929748	15.379268	4718
3	.8500	.900	10.446798	7.942499	15.540644	4718
3	.8800	.700	10.462606	8.002412	15.234454	4718
3	.8800	.740	10.430072	7.971655	15.245749	4718
3	.8800	.800	10.412690	7.946381	15.314949	4718
	.8800	.810	10.413473	7.944382	15.332573	4718

TABLE C-1
Historical 1-Step (Cont.)

3	.8800	.840	10.422128	7.941063	15.395818	4718
3	.8800	.900	10.467711	7.954851	15.568408	4718
3	.9200	.700	10.526570	8.051258	15.288747	4718
3	.9200	.740	10.502819	8.024819	15.315574	4718
3	.9200	.800	10.499442	8.008507	15.411041	4718
3	.9200	.810	10.502647	8.007805	15.433367	4718
3	.9200	.840	10.518705	8.009906	15.511235	4718
3	.9200	.900	10.579639	8.032972	15.715159	4718
4	.7500	.700	11.110148	8.457149	16.708533	4718
4	.7500	.740	11.070166	8.419014	16.666700	4718
4	.7500	.800	11.043088	8.388795	16.644236	4718
4	.7500	.810	11.042433	8.386675	16.645330	4718
4	.7500	.840	11.047092	8.385980	16.657116	4718
4	.7500	.900	11.086132	8.406248	16.719884	4718
4	.7900	.700	11.062250	8.418643	16.650273	4718
4	.7900	.740	11.017156	8.372392	16.600146	4718
4	.7900	.800	10.982072	8.343078	16.572191	4718
4	.7900	.810	10.990048	8.339788	16.571791	4718
4	.7900	.840	10.980550	8.335501	16.579010	4718
4	.7900	.900	11.011086	8.350379	16.632141	4718
4	.8720	.700	11.000898	8.362092	16.590482	4718
4	.8720	.740	10.949981	8.320023	16.539342	4718
4	.8720	.800	10.905879	8.280978	16.501723	4718
4	.8720	.810	10.902325	8.277279	16.500141	4718
4	.8720	.840	10.898196	8.270365	16.503636	4718
4	.8720	.900	10.919343	8.278552	16.548598	4718
4	.8500	.700	11.010439	8.374055	16.596475	4718
4	.8500	.740	10.960199	8.331832	16.545246	4718
4	.8500	.800	10.917107	8.292626	16.507565	4718
4	.8500	.810	10.913721	8.288821	16.505984	4718
4	.8500	.840	10.910091	8.281322	16.509504	4718
4	.8500	.900	10.932218	8.290117	16.554635	4718
4	.8800	.700	10.999316	8.358758	16.590662	4718
4	.8800	.740	10.948398	8.316290	16.539834	4718
4	.8800	.800	10.904315	8.277720	16.502685	4718
4	.8800	.810	10.900766	8.274124	16.501180	4718
4	.8800	.840	10.896661	8.267836	16.504904	4718
4	.8800	.900	10.917874	8.276430	16.550302	4718
4	.9200	.700	11.015470	8.362870	16.615993	4718
4	.9200	.740	10.967607	8.321755	16.569753	4718
4	.9200	.800	10.928511	8.287478	16.539855	4718
4	.9200	.810	10.925838	8.284586	16.539595	4718
4	.9200	.840	10.924428	8.281335	16.547107	4718
4	.9200	.900	10.951321	8.295048	16.600271	4718

STOP

TABLE C-2
Historical 2-Step

S	ALPHA	BETA	SQRT(AVE SQ)	AVE ABS	4TH RT(AVE 4TH)	PTS.
1	.7800	.750	9.189920	6.514384	14.218957	4620
1	.7800	.779	9.185808	6.500509	14.274667	4620
1	.7800	.860	9.212966	6.495142	14.494112	4620
1	.7800	.880	9.228394	6.500273	14.562207	4620
1	.7800	.905	9.252493	6.511203	14.654736	4620
1	.7800	.930	9.281903	6.524623	14.755273	4620
1	.8100	.750	9.185787	6.505121	14.264775	4620
1	.8100	.779	9.181871	6.490766	14.326416	4620
1	.8100	.860	9.209736	6.482496	14.563632	4620
1	.8100	.880	9.225371	6.486893	14.636343	4620
1	.8100	.905	9.249749	6.497286	14.734744	4620
1	.8100	.930	9.279457	6.510540	14.841250	4620
1	.8400	.750	9.187945	6.500364	14.316880	4620
1	.8400	.779	9.184947	6.486641	14.385254	4620
1	.8400	.860	9.215683	6.477791	14.643116	4620
1	.8400	.880	9.232093	6.482109	14.721187	4620
1	.8400	.905	9.257472	6.491991	14.826421	4620
1	.8400	.930	9.288216	6.505027	14.939894	4620
1	.8800	.750	9.206314	6.507418	14.394355	4620
	.8800	.779	9.206216	6.495207	14.473195	4620
	.8800	.860	9.245788	6.492027	14.766270	4620
1	.8800	.880	9.264533	6.496433	14.848547	4620
1	.8800	.905	9.292907	6.506280	14.964271	4620
1	.8800	.930	9.326726	6.520158	15.088475	4620
1	.9100	.750	9.242057	6.532061	14.460303	4620
1	.9100	.779	9.246302	6.522204	14.547839	4620
1	.9100	.860	9.298933	6.522839	14.863220	4620
1	.9100	.880	9.321091	6.529521	14.956413	4620
1	.9100	.905	9.353824	6.541477	15.080994	4620
1	.9100	.930	9.392097	6.558118	15.214281	4620
2	.7800	.750	11.410594	8.800546	15.807685	4620
2	.7800	.779	11.383596	8.778814	15.784823	4620
2	.7800	.860	11.356555	8.752139	15.781407	4620
2	.7800	.880	11.360888	8.753817	15.794258	4620
2	.7800	.905	11.372432	8.761473	15.817929	4620
2	.7800	.930	11.390765	8.773226	15.850066	4620
2	.8100	.750	11.379859	8.772541	15.778646	4620
2	.8100	.779	11.350849	8.748824	15.754324	4620
2	.8100	.860	11.318005	8.718388	15.746980	4620
2	.8100	.880	11.320874	8.719250	15.758907	4620
2	.8100	.905	11.330573	8.724813	15.781449	4620
2	.8100	.930	11.347048	8.735605	15.812451	4620
2	.8400	.750	11.356616	8.750542	15.762017	4620
2	.8400	.779	11.326284	8.725263	15.737389	4620
2	.8400	.860	11.289686	8.692975	15.729593	4620
	.8400	.880	11.291620	8.693003	15.741501	4620
	.8400	.905	11.300150	8.697609	15.764076	4620
2	.8400	.930	11.315457	8.706832	15.795170	4620

TABLE C-2
Historical 2-Step (Cont.)

2	.8800	.750	11.342850	8.732697	15.767092	4620
2	.8800	.779	11.312351	8.707201	15.744577	4620
2	.8800	.860	11.275504	8.676540	15.743690	4620
2	.8800	.880	11.277426	8.677584	15.757533	4620
2	.8800	.905	11.285969	8.682673	15.782649	4620
2	.8800	.930	11.301321	8.691654	15.816422	4620
2	.9100	.750	11.354032	8.738687	15.801114	4620
2	.9100	.779	11.325393	8.715628	15.782952	4620
2	.9100	.860	11.294267	8.684791	15.795481	4620
2	.9100	.880	11.297712	8.685845	15.812912	4620
2	.9100	.905	11.308216	8.690463	15.842660	4620
2	.9100	.930	11.325588	8.700746	15.881223	4620
3	.7800	.750	10.657825	8.084293	15.980784	4620
3	.7800	.779	10.624729	8.055569	15.946887	4620
3	.7800	.860	10.577626	8.010105	15.916038	4620
3	.7800	.880	10.576350	8.004849	15.922640	4620
3	.7800	.905	10.580536	8.003225	15.938648	4620
3	.7800	.930	10.591138	8.007184	15.963162	4620
3	.8100	.750	10.642588	8.073968	15.953995	4620
3	.8100	.779	10.608425	8.044535	15.919467	4620
3	.8100	.860	10.558219	7.997298	15.886394	4620
3	.8100	.880	10.556154	7.993037	15.892317	4620
3	.8100	.905	10.559344	7.991811	15.907395	4620
3	.8100	.930	10.568940	7.996428	15.930881	4620
[illegible]	.8400	.750	10.636715	8.070605	15.938579	4620
[illegible]	.8400	.779	10.602329	8.040917	15.904727	4620
3	.8400	.860	10.551528	7.994052	15.873483	4620
3	.8400	.880	10.549324	7.990170	15.879820	4620
3	.8400	.905	10.552344	7.990782	15.895382	4620
3	.8400	.930	10.561777	7.995784	15.919310	4620
3	.8800	.750	10.652887	8.082302	15.946458	4620
3	.8800	.779	10.622317	8.053735	15.916689	4620
3	.8800	.860	10.575001	8.012169	15.897798	4620
3	.8800	.880	10.574238	8.009820	15.907367	4620
3	.8800	.905	10.579101	8.011390	15.927050	4620
3	.8800	.930	10.590422	8.017504	15.955181	4620
3	.9100	.750	10.699390	8.117349	15.992991	4620
3	.9100	.779	10.671135	8.092593	15.970575	4620
3	.9100	.860	10.638668	8.058214	15.974217	4620
3	.9100	.880	10.641239	8.057963	15.989770	4620
3	.9100	.905	10.650349	8.061735	16.017152	4620
3	.9100	.930	10.665996	8.070314	16.053212	4620
4	.7800	.750	11.233375	8.459802	16.979321	4620
4	.7800	.779	11.194354	8.469159	16.924561	4620
4	.7800	.860	11.133033	8.418026	16.829877	4620
4	.7800	.880	11.128797	8.412984	16.819622	4620
4	.7800	.905	11.129600	8.411707	16.814067	4620
4	.7800	.930	11.137178	8.415979	16.816559	4620
4	.8100	.750	11.205390	8.478930	16.959924	4620
4	.8100	.779	11.164197	8.446507	16.903384	4620
	.8100	.860	11.096489	8.389469	16.802845	4620
[illegible]	.8100	.880	11.09[illegible]614	8.382890	16.790927	4620

TABLE C-2
Historical 2-Step (Cont.)

4	.8100	.905	11.089339	8.379341	16.783167	4620
4	.8100	.930	11.094808	8.381762	16.783207	4620
4	.8400	.750	11.183999	8.462827	16.950864	4620
4	.8400	.779	11.141246	8.429636	16.893505	4620
4	.8400	.860	11.068977	8.366669	16.790009	4620
4	.8400	.880	11.061936	8.358680	16.777192	4620
4	.8400	.905	11.059186	8.354010	16.768204	4620
4	.8400	.930	11.063163	8.354456	16.766999	4620
4	.8800	.750	11.171310	8.455327	16.958806	4620
	.8800	.779	11.127938	8.420764	16.902328	4620
	.8800	.860	11.053947	8.353526	16.800944	4620
4	.8800	.880	11.046486	8.344391	16.788545	4620
4	.8800	.905	11.043214	8.338508	16.780012	4620
4	.8800	.930	11.046675	8.338942	16.779177	4620
4	.9100	.750	11.182673	8.469286	16.984463	4620
4	.9100	.779	11.140739	8.436220	16.930725	4620
4	.9100	.860	11.071128	8.371608	16.837426	4620
4	.9100	.880	11.064825	8.362635	16.827109	4620
4	.9100	.905	11.063039	8.356090	16.821219	4620
4	.9100	.930	11.068027	8.357721	16.823073	4620

STOP

TABLE C-3
Non-Historical 1-Step

S	ALPHA	BETA	SQRT(AVE SQ)	AVE ABS	4TH RT(AVE 4TH)	PTS
1	.2200	.350	10.743837	7.586454	16.558669	471
1	.2200	.390	10.735345	7.568439	16.585098	471
1	.2200	.460	10.763178	7.562359	16.662326	471
1	.2200	.495	10.797367	7.581083	16.725203	471
1	.2200	.590	10.956988	7.665263	17.034209	471
1	.2200	.630	11.052654	7.722007	17.160761	471
1	.2600	.350	10.742664	7.564341	16.561416	471
1	.2600	.390	10.723226	7.545363	16.586492	471
1	.2600	.460	10.758510	7.560368	16.681721	471
1	.2600	.495	10.791192	7.573167	16.753326	471
1	.2600	.590	10.946014	7.654284	17.028004	471
1	.2600	.630	11.039378	7.707872	17.176521	471
1	.3900	.350	10.769054	7.597523	16.615604	471
1	.3900	.390	10.761238	7.578035	16.645965	471
1	.3900	.460	10.789056	7.572866	16.749456	471
1	.3900	.495	10.822667	7.583593	16.825118	471
1	.3900	.590	10.978669	7.665107	17.110466	471
1	.3900	.630	11.072324	7.716675	17.265394	471
1	.4300	.350	10.783224	7.609456	16.649194	471
1	.4300	.390	10.763104	7.590993	16.684078	471
1	.4300	.460	10.815787	7.586614	16.795793	471
1	.4300	.495	10.851884	7.600922	16.875727	471
1	.4300	.590	11.014915	7.686620	17.173180	471
1	.4300	.630	11.111261	7.741656	17.333428	471
1	.5700	.350	10.919155	7.695372	16.858003	471
1	.5700	.390	10.935683	7.691475	16.924207	471
1	.5700	.460	11.008766	7.715450	17.093943	471
1	.5700	.495	11.066100	7.743567	17.204374	471
1	.5700	.590	11.269476	7.867593	17.589310	471
1	.5700	.630	11.412113	7.938793	17.788310	471
1	.6000	.350	10.965815	7.727082	16.928196	471
1	.6000	.340	10.990558	7.728683	17.005385	471
1	.6000	.460	11.079122	7.763844	17.195556	471
1	.6000	.495	11.144657	7.797249	17.316755	471
1	.6000	.590	11.391519	7.935177	17.732626	471
1	.6000	.630	11.524461	8.013492	17.945331	471
2	.2200	.350	13.255406	10.222333	18.260664	471
2	.2200	.390	13.217507	10.190212	18.229813	471
2	.2200	.460	13.203991	10.175510	16.214353	471
2	.2200	.495	13.222495	10.188051	18.242167	471
2	.2200	.590	13.356830	10.282143	18.436765	471
2	.2200	.630	13.449505	10.346903	18.570030	471
2	.2600	.350	13.241395	10.210958	18.271098	471
2	.2600	.390	13.200006	10.170377	18.216979	471
2	.2600	.460	13.179517	10.155306	18.195342	471
2	.2500	.495	13.194146	10.163910	18.219713	471
2	.2600	.590	13.316815	10.247140	18.403926	471

TABLE C-3
Non-Historical 1-Step (Cont.)

2	.2600	.630	13.404137	10.306724	18.532416	4718
2	.3900	.350	13.222912	10.194634	18.282943	4718
2	.3900	.390	13.175587	10.154509	18.227969	4718
2	.3900	.460	13.142690	10.120093	18.204150	4718
2	.3900	.495	13.150591	10.119505	18.227143	4718
2	.3900	.590	13.251943	10.178434	18.406746	4718
2	.3900	.630	13.329309	10.229205	18.532917	471
2	.4300	.350	13.227560	10.197944	18.303372	4718
2	.4300	.390	13.180441	10.156409	18.251[illegible]	471
2	.4300	.460	13.151914	10.122362	18.234075	4718
2	.4300	.495	13.155616	10.116531	18.260434	4718
2	.4300	.590	13.256061	10.173011	18.449910	4718
2	.4300	.630	13.333769	10.222183	18.560492	4718
2	.5700	.350	13.303740	10.249304	18.473356	4718
2	.5700	.390	13.269068	10.215062	18.453585	4718
2	.5700	.460	13.259806	10.189307	18.498356	4718
2	.5700	.495	13.279756	10.193748	18.558790	4718
2	.5700	.590	13.415700	10.256758	18.849122	4718
2	.5700	.630	13.508054	10.308561	19.025178	4718
2	.6000	.350	13.337764	10.273760	18.538948	4718
2	.6000	.390	13.309245	10.243275	18.532171	4718
2	.6000	.460	13.311698	10.224688	18.602384	4718
2	.6000	.495	13.357910	10.231942	18.676736	4718
2	.6000	.590	13.491987	10.304291	19.008255	4718
	.6000	.630	13.592378	10.358333	19.202870	4718
3	.2200	.350	11.837459	9.090517	17.294207	4718
3	.2200	.390	11.771459	9.040314	17.242131	4718
3	.2200	.460	11.703673	8.987885	17.211993	4718
3	.2200	.495	11.692632	8.977215	17.226235	4718
3	.2200	.590	11.741175	9.004700	17.363519	4718
3	.2200	.630	11.795278	9.038894	17.464288	4718
3	.2600	.350	11.807846	9.070013	17.228543	4718
3	.2600	.390	11.735957	9.015752	17.165425	4718
3	.2600	.460	11.656736	8.956257	17.114805	4718
3	.2600	.495	11.639078	8.941857	17.118274	4718
3	.2600	.590	11.669695	8.956707	17.224978	4718
3	.2600	.630	11.715544	8.985236	17.312461	4718
3	.3900	.350	11.731217	9.018633	17.038563	4718
3	.3900	.390	11.643858	8.953630	16.944211	4718
3	.3900	.460	11.534464	8.874584	16.835238	4718
3	.3900	.495	11.500935	8.853135	16.807997	4718
3	.3900	.590	11.482158	8.833782	16.827341	4718
3	.3900	.630	11.505920	8.846745	16.870755	4718
3	.4300	.350	11.714151	9.007526	16.987964	4718
3	.4300	.390	11.623269	8.941084	16.885496	4718
3	.4300	.460	11.506967	8.856727	16.761417	4718
3	.4300	.495	11.469649	8.832160	16.726236	4718
3	.4300	.590	11.439583	8.806776	16.723003	4718
3	.4300	.630	11.458207	8.815731	16.762566	4718
	.5700	.350	11.685159	8.990656	16.846640	4718
3	.5700	.390	11.587997	8.921114	16.722938	4718
3	.5700	.460	11.459280	8.831440	16.559880	4718

TABLE C-3
Non-Historical 1-Step (Cont.)

3	.5700	.495	11.415105	8.801546	16.504473	4718
3	.5700	.590	11.364452	8.763223	16.444480	4718
3	.5700	.630	11.373640	8.767726	16.459567	4718
3	.6000	.350	11.667082	8.993844	16.825396	4718
3	.6000	.390	11.590224	8.924793	16.699017	4718
3	.6000	.460	11.462072	8.834763	16.531294	4718
3	.6000	.495	11.416194	8.800031	16.473569	4718
3	.6000	.590	11.368390	8.769946	16.407558	4718
3	.6000	.630	11.377948	8.774801	16.420213	4718
4	.2200	.350	12.953542	9.849031	18.834472	4718
4	.2200	.390	12.904588	9.809065	18.785725	4718
4	.2200	.460	12.870895	9.772654	18.765520	4718
4	.2200	.495	12.878994	9.772483	18.786545	4718
4	.2200	.590	12.984398	9.844790	18.948066	4718
4	.2200	.630	13.064721	9.903886	19.061533	4718
4	.2600	.350	12.934355	9.831686	18.802456	4718
~~4~~	~~.2600~~	~~.390~~	~~12.881061~~	~~9.769311~~	~~18.746896~~	~~4718~~
4	.2600	.460	12.838356	9.745657	18.713415	4718
4	.2600	.495	12.842257	9.742181	18.727206	4718
4	.2600	.590	12.923638	9.802632	18.867409	4718
4	.2600	.630	13.007575	9.855548	18.971263	4718
4	.3900	.350	12.898564	9.795151	18.739279	4718
4	.3900	.390	12.836364	9.746060	18.669917	4718
4	.3900	.460	12.776254	9.691296	18.609587	4718
4	.3900	.495	12.769692	9.679736	18.608801	4718
4	.3900	.590	12.830962	9.714396	18.706226	4718
4	.3900	.630	12.891037	9.754984	18.790946	4718
4	.4300	.350	12.897290	9.792294	18.734578	4718
4	.4300	.390	12.834215	9.742828	18.664006	4718
4	.4300	.460	12.772224	9.685821	18.601407	4718
4	.4300	.495	12.764551	9.673782	18.599444	4718
4	.4300	.590	12.822283	9.705653	18.693633	4718
4	.4300	.630	12.880657	9.744600	18.777004	4718
4	.5700	.350	12.946608	9.824764	18.797724	4718
4	.5700	.390	12.890966	9.780459	18.739650	4718
4	.5700	.460	12.842531	9.733726	18.702394	4718
4	.5700	.495	12.842055	9.727036	18.714819	4718
4	.5700	.590	12.919695	9.770047	18.853822	4718
4	.5700	.630	12.986539	9.812638	18.958456	4718
4	.6000	.350	12.972781	9.843616	18.834185	4718
4	.6000	.390	12.921809	9.802628	18.783676	4718
4	.6000	.460	12.882363	9.761779	18.761595	4718
4	.6000	.495	12.880541	9.754563	18.782533	4718
4	.6000	.590	12.977956	9.810879	18.947616	4718
4	.6000	.630	13.051012	9.857217	19.064434	4718

TABLE C-4
Non-Historical 2-Step

B	ALPHA	BETA	SQRT(AVE SQ)	AVE ABS	4TH RT(AVE 4TH)	PTS
1	.0700	.320	12.558985	8.721205	19.417607	462
1	.0700	.360	12.551290	8.707791	19.422708	462
1	.0700	.464	12.608134	8.731202	19.545020	462
1	.0700	.520	12.684297	8.778905	19.676301	462
1	.0700	.550	12.737986	8.814735	19.765363	462
1	.0900	.320	12.559067	8.719990	19.422161	462
1	.0900	.360	12.550522	8.705757	19.426788	462
1	.0900	.464	12.604278	8.725712	19.546698	462
1	.0900	.520	12.678278	8.770722	19.676000	462
1	.0900	.550	12.730675	8.804925	19.763813	462
1	.3000	.320	12.604638	8.744510	19.530566	462
1	.3000	.360	12.596147	8.725916	19.542469	462
1	.3000	.464	12.643900	8.731005	19.673440	462
1	.3000	.520	12.711067	8.762112	19.803893	462
1	.3000	.550	12.758801	8.786838	19.890944	462
1	.3250	.320	12.616428	8.752657	19.551592	462
1	.3250	.360	12.609179	8.734506	19.565919	462
1	.3250	.464	12.659829	8.741063	19.702772	462
1	.3250	.520	12.728340	8.771490	19.836102	462
1	.3250	.550	12.776731	8.796199	19.924605	462
1	.3500	.320	12.629923	8.761871	19.574681	462
1	.3500	.360	12.624237	8.744661	19.591805	462
1	.3500	.464	12.678716	8.753204	19.735617	462
1	.3500	.520	12.749125	8.783911	19.872465	462
1	.3500	.550	12.798482	8.809052	19.962776	462
2	.0700	.320	15.118566	11.606165	20.969243	462
2	.0700	.360	15.076547	11.561919	20.940324	462
2	.0700	.464	15.065697	11.530771	20.996397	462
2	.0700	.520	15.118766	11.566750	21.104454	462
2	.0700	.550	15.164010	11.598477	21.184406	462
2	.0900	.320	15.110964	11.598200	20.965571	462
2	.0900	.360	15.066732	11.552039	20.934791	462
2	.0900	.464	15.048879	11.513993	20.984617	462
2	.0900	.520	15.097484	11.546396	21.088503	462
2	.0900	.550	15.140153	11.576253	21.165997	462
2	.3000	.320	15.079676	11.546831	21.006685	462
2	.3000	.360	15.022168	11.485104	20.972535	462
2	.3000	.464	14.960224	11.404336	21.004652	462
2	.3000	.520	14.979649	11.405303	21.093644	462
2	.3000	.550	15.005229	11.419586	21.161620	462
2	.3250	.320	15.082768	11.545404	21.023084	462
2	.3250	.360	15.025020	11.482658	20.990802	462
2	.3250	.464	14.961760	11.399854	21.027486	462
2	.3250	.520	14.980070	11.398711	21.118716	462
2	.3250	.550	15.004938	11.411835	21.187813	462
2	.3500	.320	15.087649	11.545157	21.042557	462
2	.3500	.360	15.030001	11.481709	21.012736	462
2	.3500	.464	14.966421	11.397719	21.055722	462

TABLE C-4
Non-Historical 2-Step (Cont.)

2	.3500	.520	14.984210	11.394491	21.150230	4620
2	.3500	.550	15.008701	11.406522	21.221032	4620
3	.0700	.320	12.894118	9.813073	19.115155	4620
3	.0700	.360	12.797410	9.747367	18.945864	4620
3	.0700	.464	12.639158	9.652593	18.636101	4620
3	.0700	.520	12.610999	9.640100	18.551452	4620
3	.0700	.550	12.612526	9.644796	18.530659	4620
3	.0900	.320	12.882813	9.804497	19.095016	4620
3	.0900	.360	12.783249	9.736287	18.921263	4620
3	.0900	.464	12.616102	9.635043	18.597823	4620
3	.0900	.520	12.582333	9.618439	18.504646	4620
3	.0900	.550	12.580640	9.620899	18.478983	4620
3	.3000	.320	12.803364	9.747369	18.927944	4620
3	.3000	.360	12.681733	9.662203	18.717006	4620
3	.3000	.464	12.444468	9.501510	18.278801	4620
3	.3000	.520	12.365671	9.451198	18.113355	4620
3	.3000	.550	12.337905	9.436061	18.046167	4620
3	.3250	.320	12.798593	9.744701	18.913073	4620
3	.3250	.360	12.675279	9.658317	18.698844	4620
3	.3250	.464	12.432450	9.493120	18.250416	4620
3	.3250	.520	12.349966	9.440197	18.078458	4620
3	.3250	.550	12.320036	9.423217	18.007498	4620
3	.3500	.320	12.794854	9.743241	18.899241	4620
	.3500	.360	12.670063	9.655657	18.681967	4620
3	.3500	.464	12.422291	9.486707	18.224068	4620
3	.3500	.520	12.336494	9.431229	18.046067	4620
3	.3500	.550	12.304610	9.412819	17.971597	4620
4	.0700	.320	14.559592	11.040774	21.050152	4620
4	.0700	.360	14.494355	10.985564	20.948437	4620
4	.0700	.464	14.423176	10.921237	20.828425	4620
4	.0700	.520	14.444260	10.938782	20.849148	4620
4	.0700	.550	14.472632	10.961538	20.884269	4620
4	.0900	.320	14.551961	11.034938	21.040416	4620
4	.0900	.360	14.484449	10.977825	20.935870	4620
4	.0900	.464	14.406030	10.907136	20.806781	4620
4	.0900	.520	14.422470	10.920167	20.821625	4620
4	.0900	.550	14.448159	10.940419	20.853327	4620
4	.3000	.320	14.518455	11.004825	21.011017	4620
4	.3000	.360	14.436550	10.937234	20.892178	4620
4	.3000	.464	14.310068	10.822753	20.714517	4620
4	.3000	.520	14.294543	10.805176	20.696486	4620
4	.3000	.550	14.301466	10.806623	20.708689	4620
4	.3250	.320	14.520840	11.006148	21.017497	4620
4	.3250	.360	14.438475	10.937955	20.899038	4620
4	.3250	.464	14.309953	10.820590	20.721865	4620
4	.3250	.520	14.292832	10.800639	20.703774	4620
4	.3250	.550	14.298761	10.800613	20.715843	4620
4	.3500	.320	14.524847	11.008696	21.026502	4620
4	.3500	.360	14.442332	10.939737	20.908944	4620
4	.3500	.464	14.312681	10.820659	20.733821	4620
4	.3500	.520	14.294512	10.798422	20.716628	4620
4	.3500	.550	14.299785	10.797386	20.729107	4620

APPENDIX D

DATA BASE

The data used in the analyses were obtained from Dr. John Kreer of Michigan State University. Dr. Kreer used the data himself in previous traffic studies.[3] The data consist of volume counts taken from four separate locations on the streets of Toronto between 9/24/73 and 12/10/73. At each location volume counts were obtained from each of two adjacent lanes. In this study the volume counts from each lane were added together to form a single volume count for each location. The four locations are shown in Figures D1 and D2 (indicated by the detector pairs). Note that the traffic is eastbound at location 3 while it is westbound at locations 2 and 4. ("Location" numbers are equivalent to "sensor" numbers elsewhere in this report.)

The volume counts were broken down into 5-minute time intervals, numbered from 1 to 288 for each day. There were 76 days of data. Except for 2 days that were missing (11/25/73 and 12/8/73), the days were consecutive and included weekends. The first 29 week days, with the exception of the October 8th holiday, formed Part 1 of the data for this study. The remaining 23 weekdays formed Part 2 of the data.

Part 2 of the data is used as historical data in this study, while Part 1 is used to test the predictors. Some of the data used in Part 1 is shown in Figures D3 through D22. Data for each day of the week are shown for each location. The day shown at the top right hand corner of each figure represents the particular day of testing (from the 76 total days). The data show the daily variations at each location as well as different characteristics peculiar to each location. The gaps shown are typical of the data available.

There are 98 gaps (a gap is a group of consecutive time intervals having no data) in Part 1 of the data. The average size of a gap is 38.55 time intervals or 3.2 hours. The actual sizes vary from 1 time interval to 148. In this prediction analysis the gaps were processed as follows. Each time a gap occurred, the calculation of the prediction errors at the time of the gap and at the next four or five consecutive time intervals of existing data (depending on whether the 1- or 2-step predictor was being used -- see text) was not included in the calculation of the total prediction error measures (mean square error, mean absolute error and mean of the fourth power of the error) for the evaluation runs. This was done in order to allow the predictor to re-initialize after passing through a gap.

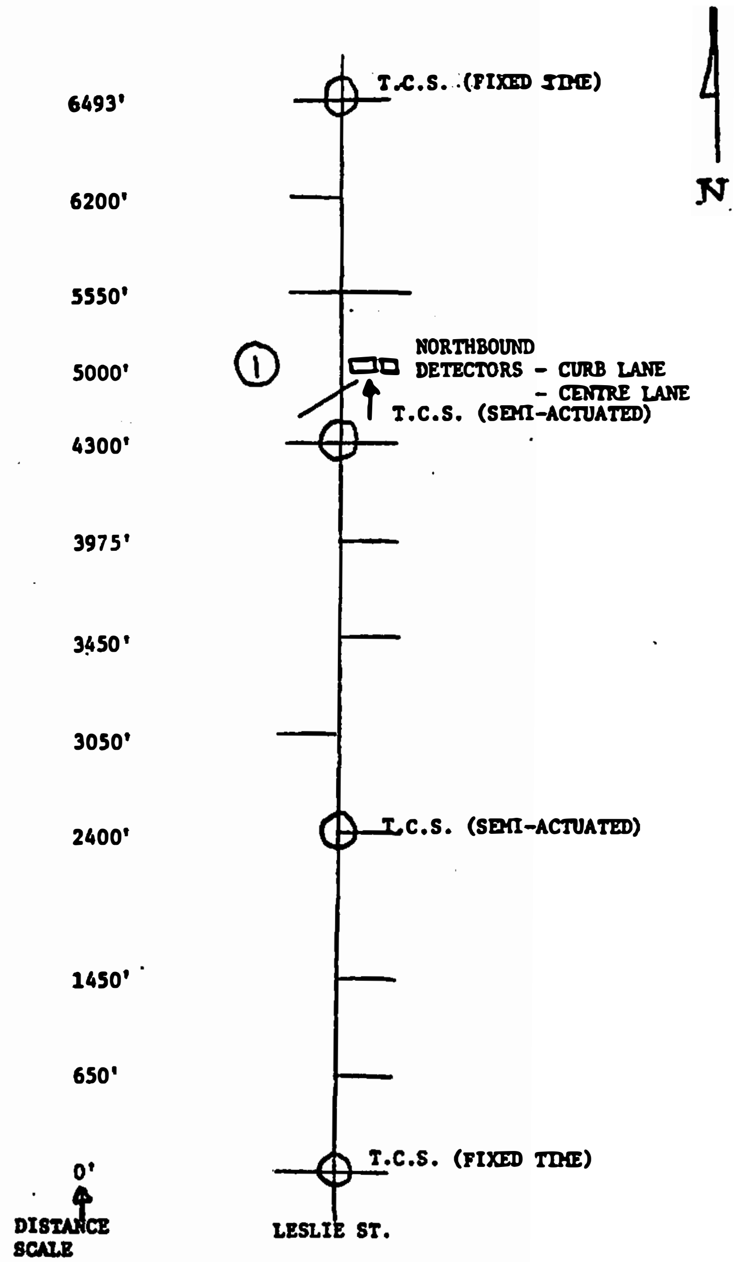

FIGURE D1. LOCATION OF SENSOR

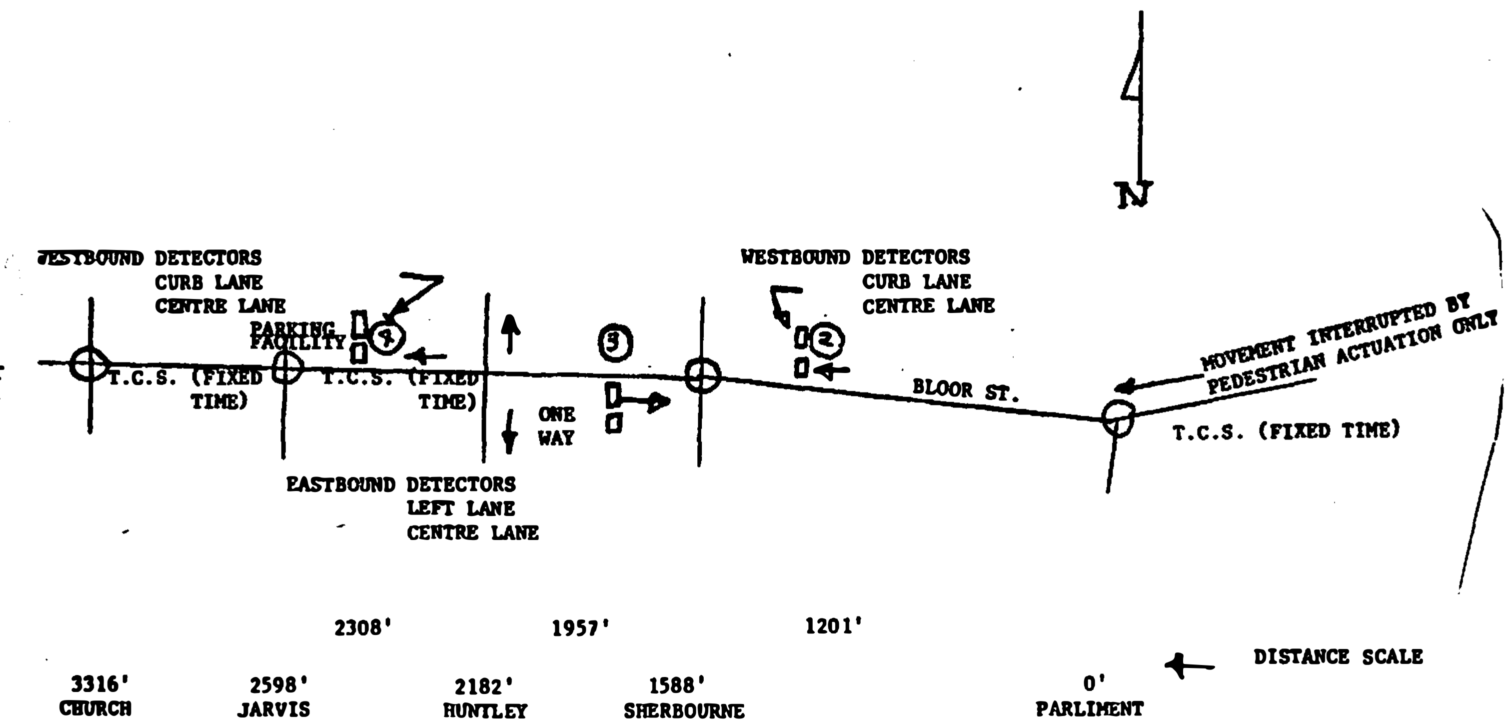

FIGURE D2. LOCATIONS OF SENSORS 2, 3 AND 4

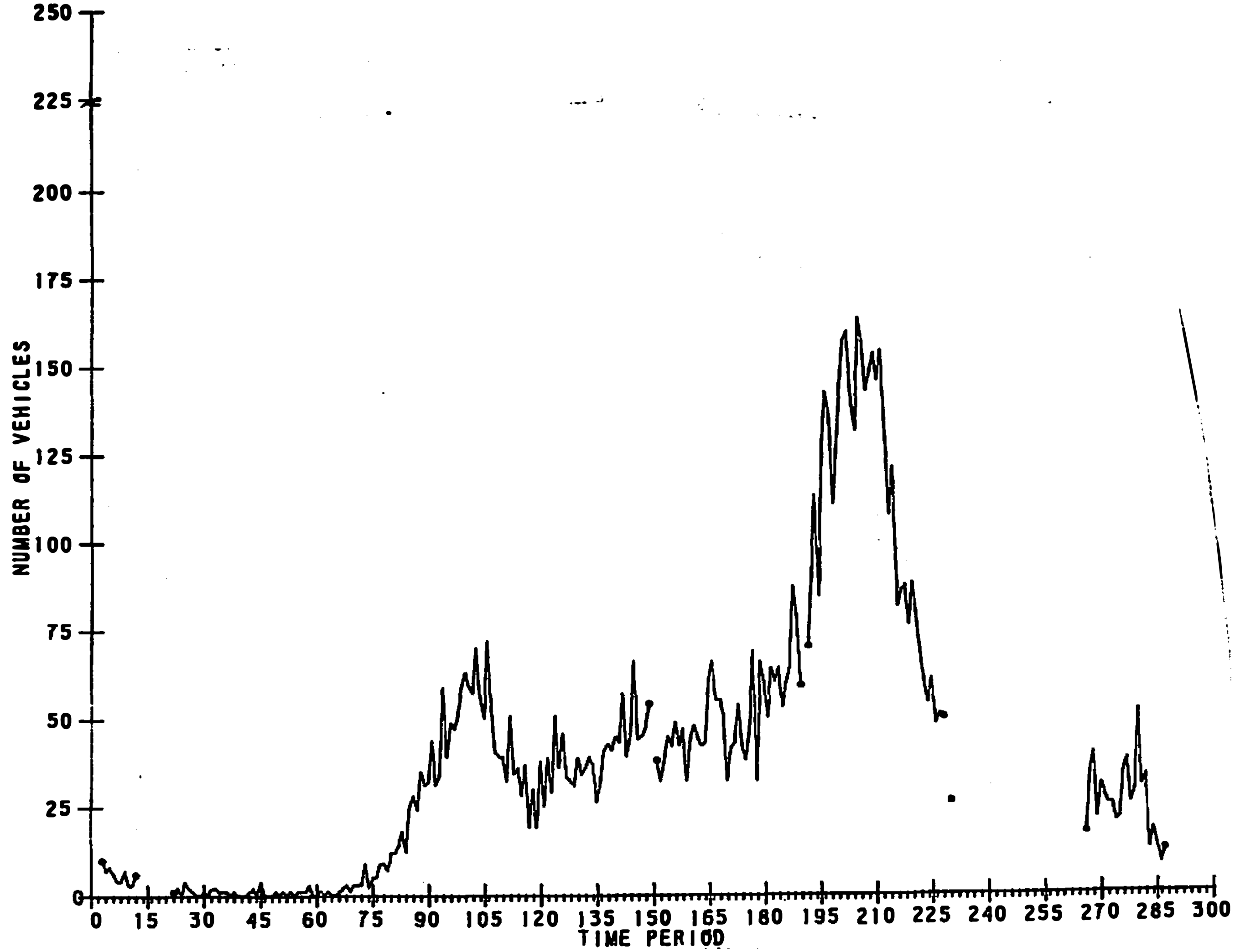

FIGURE D3. VOLUME VS. TIME, SENSOR 1, MONDAY, DAY 1

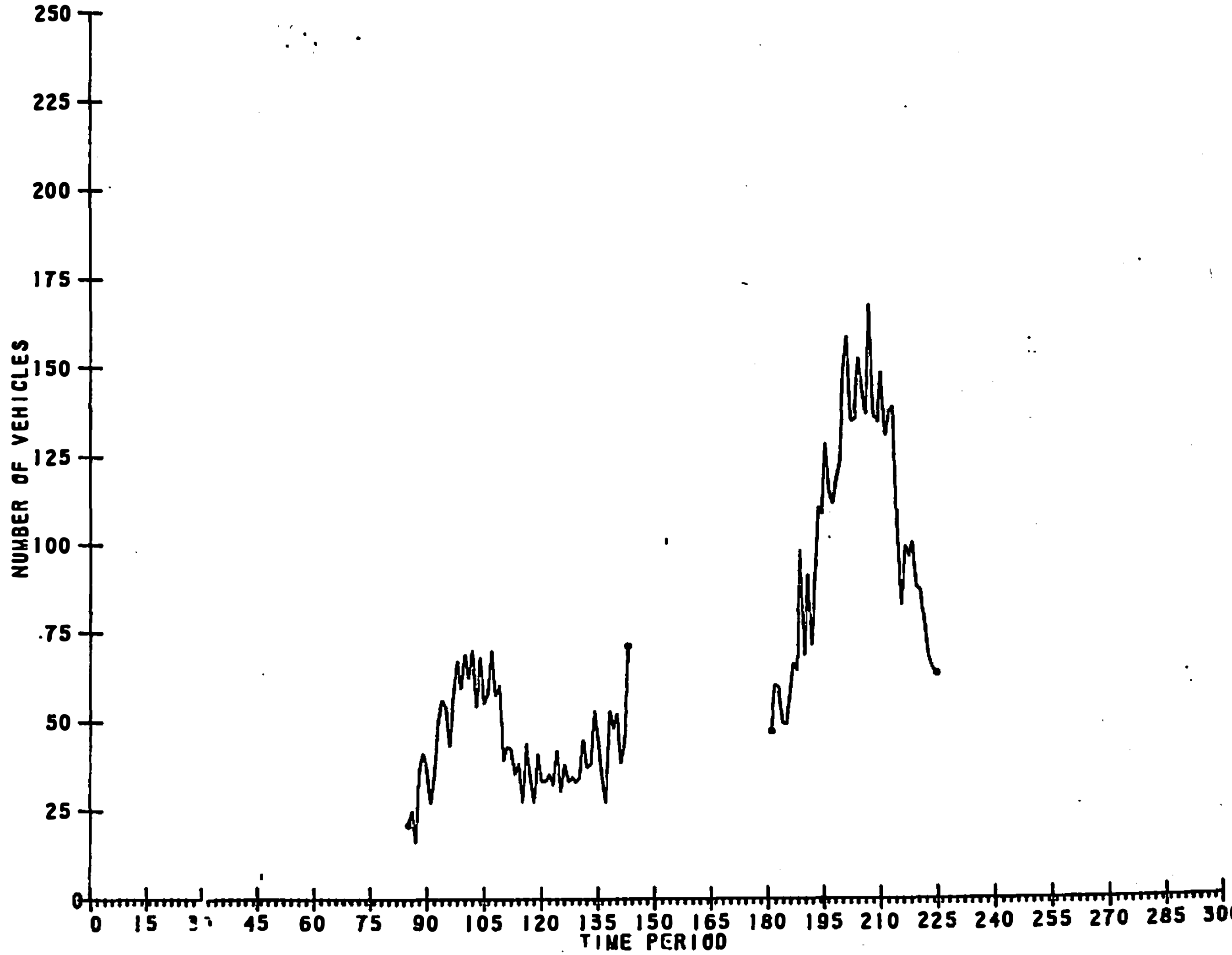

FIGURE D4. VOLUME VS. TIME, SENSOR 1, TUESDAY, DAY 37

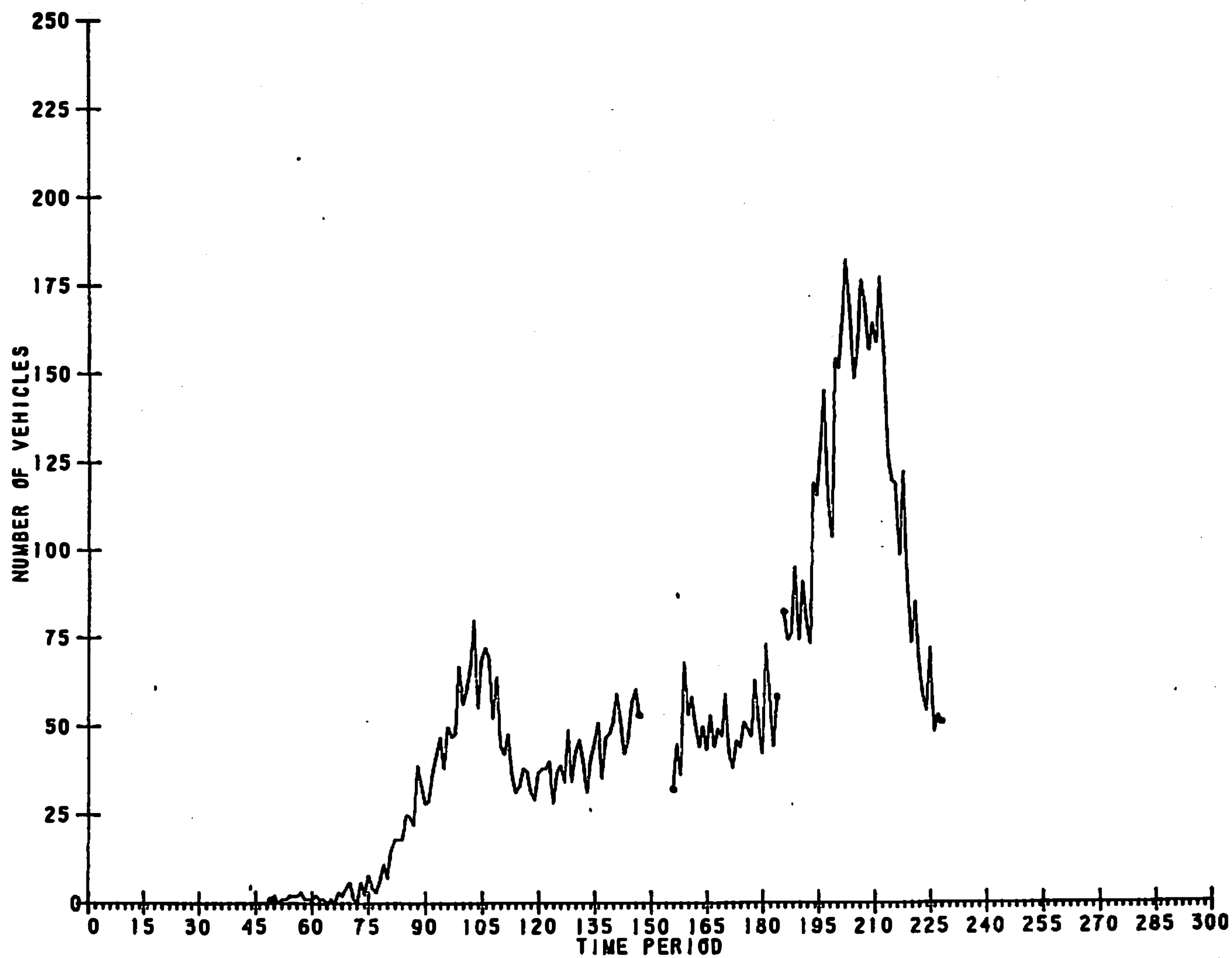

FIGURE D5. VOLUME VS. TIME, SENSOR 1, WEDNESDAY, DAY 31

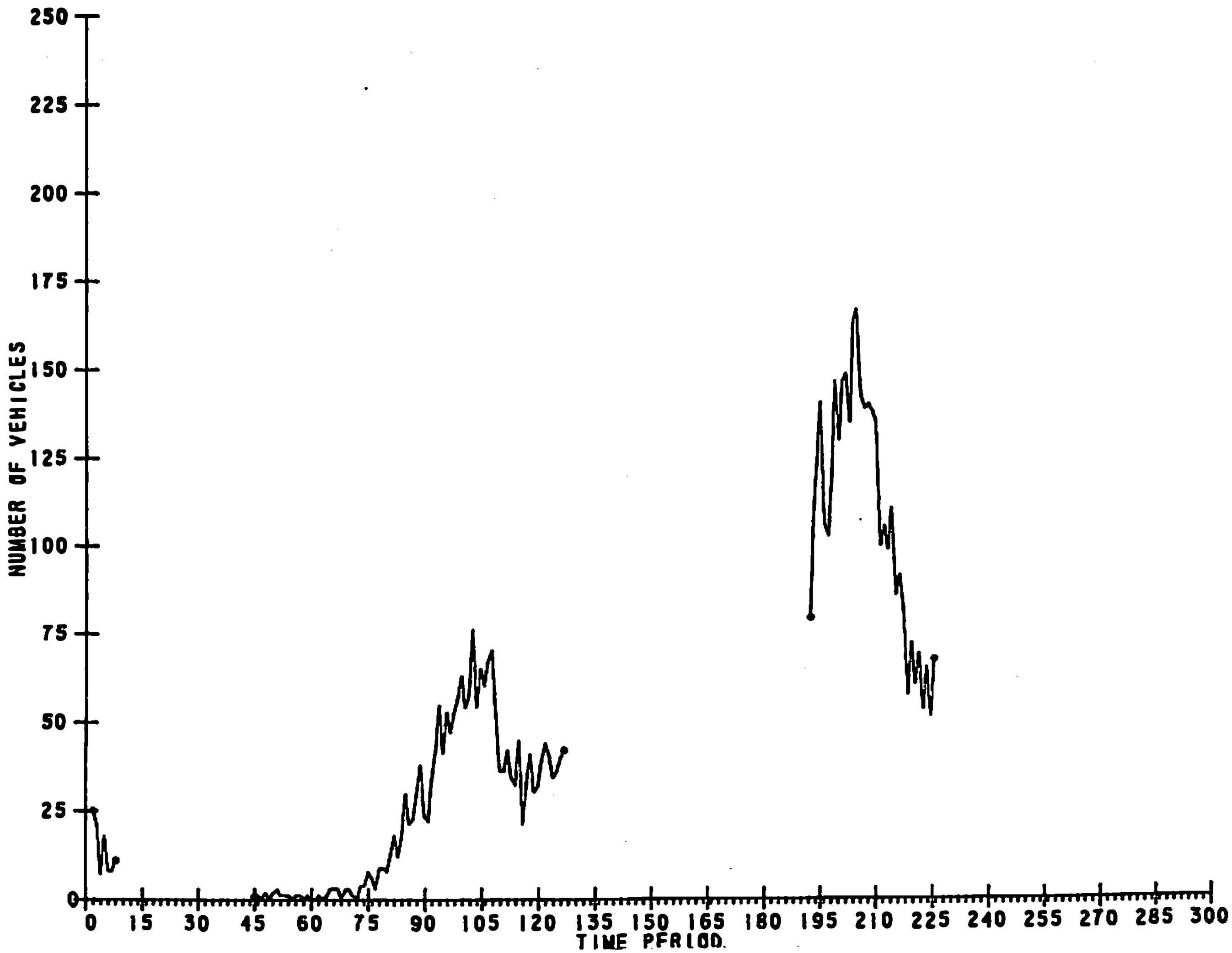

FIGURE D6. VOLUME VS. TIME, SENSOR 1, THURSDAY, DAY 18

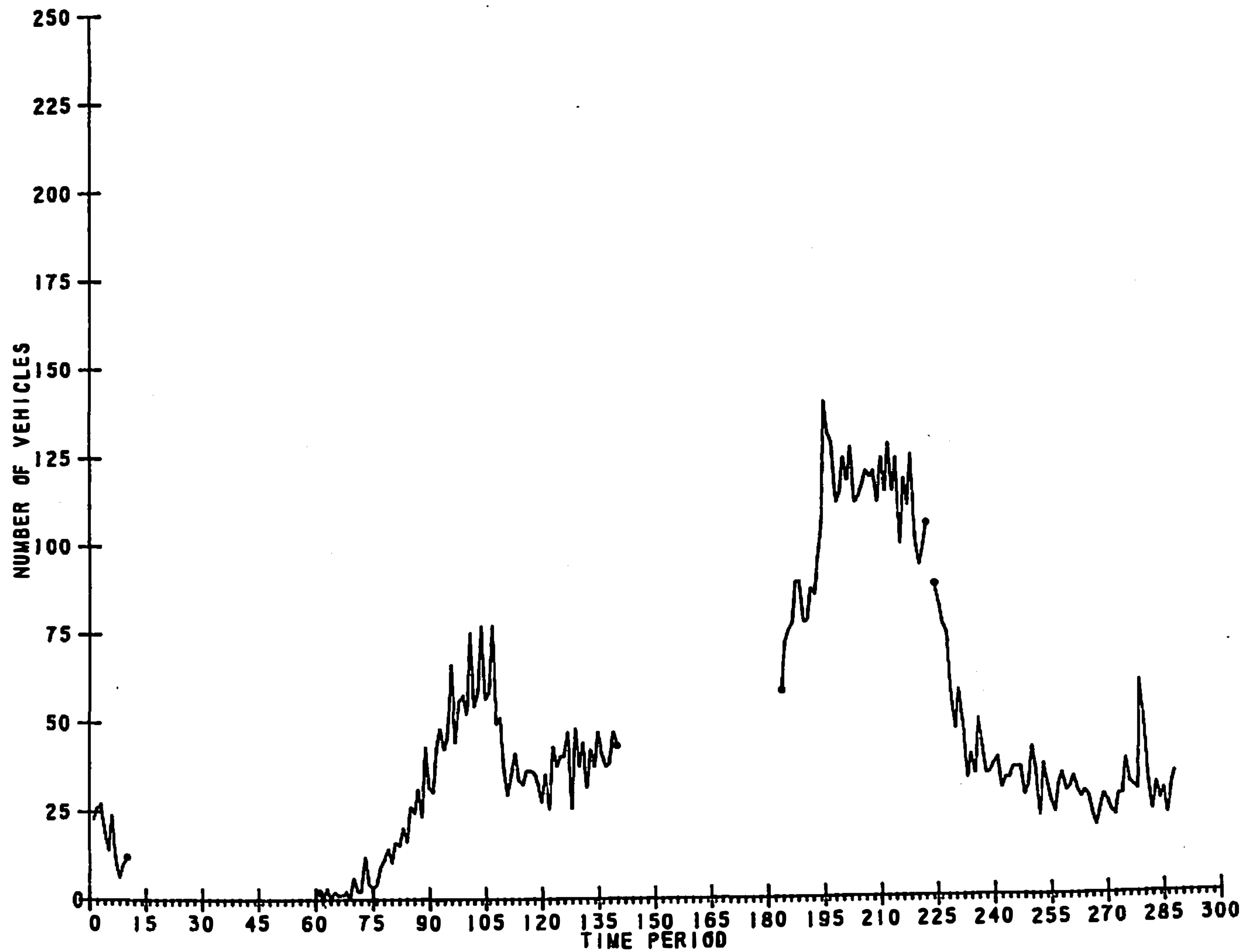

FIGURE D7. VOLUME VS. TIME, SENSOR 1, FRIDAY, DAY 40

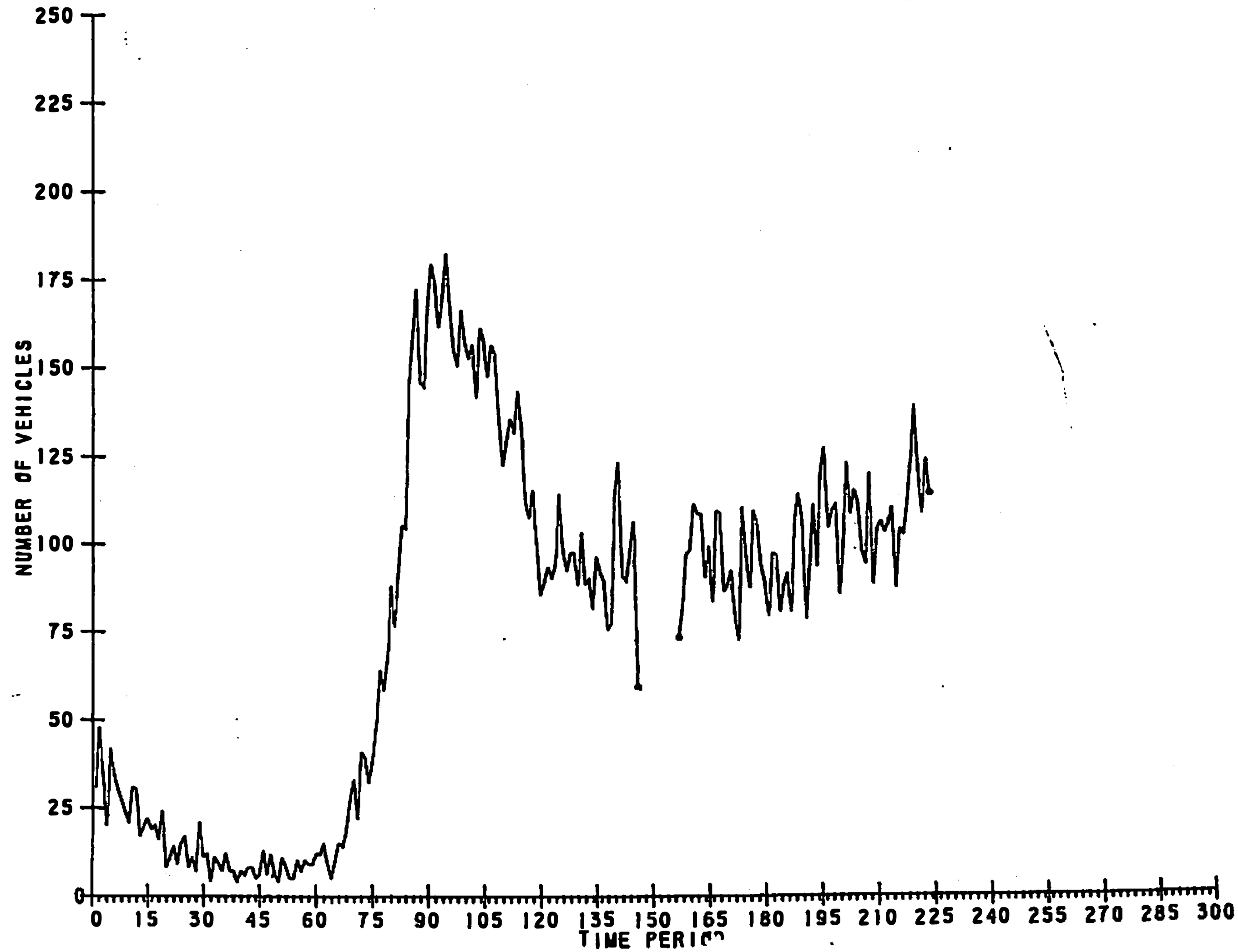

FIGURE D8. VOLUME VS. TIME, SENSOR 2, MONDAY, DAY 29

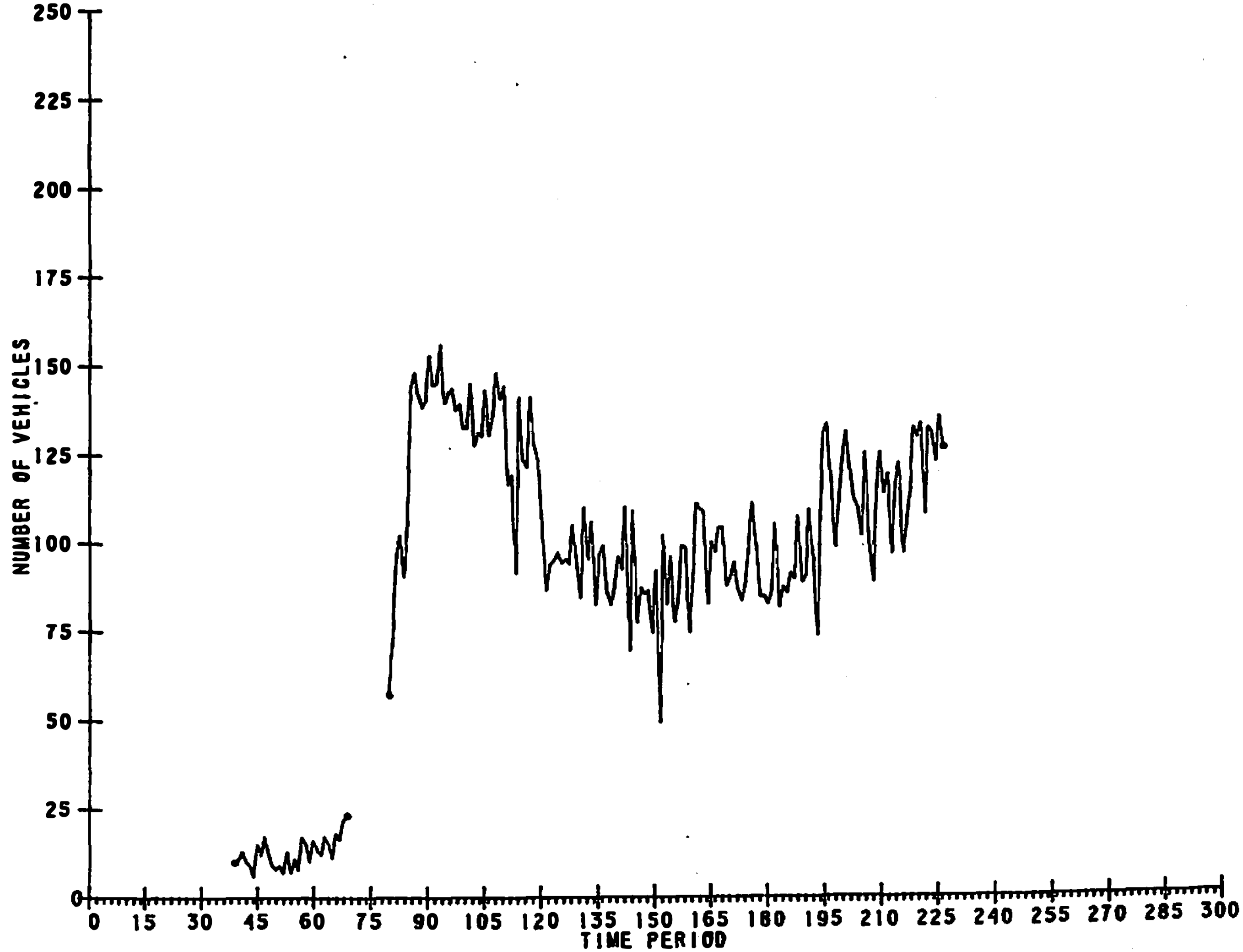

FIGURE D9. VOLUME VS. TIME, SENSOR 2, TUESDAY, DAY 9

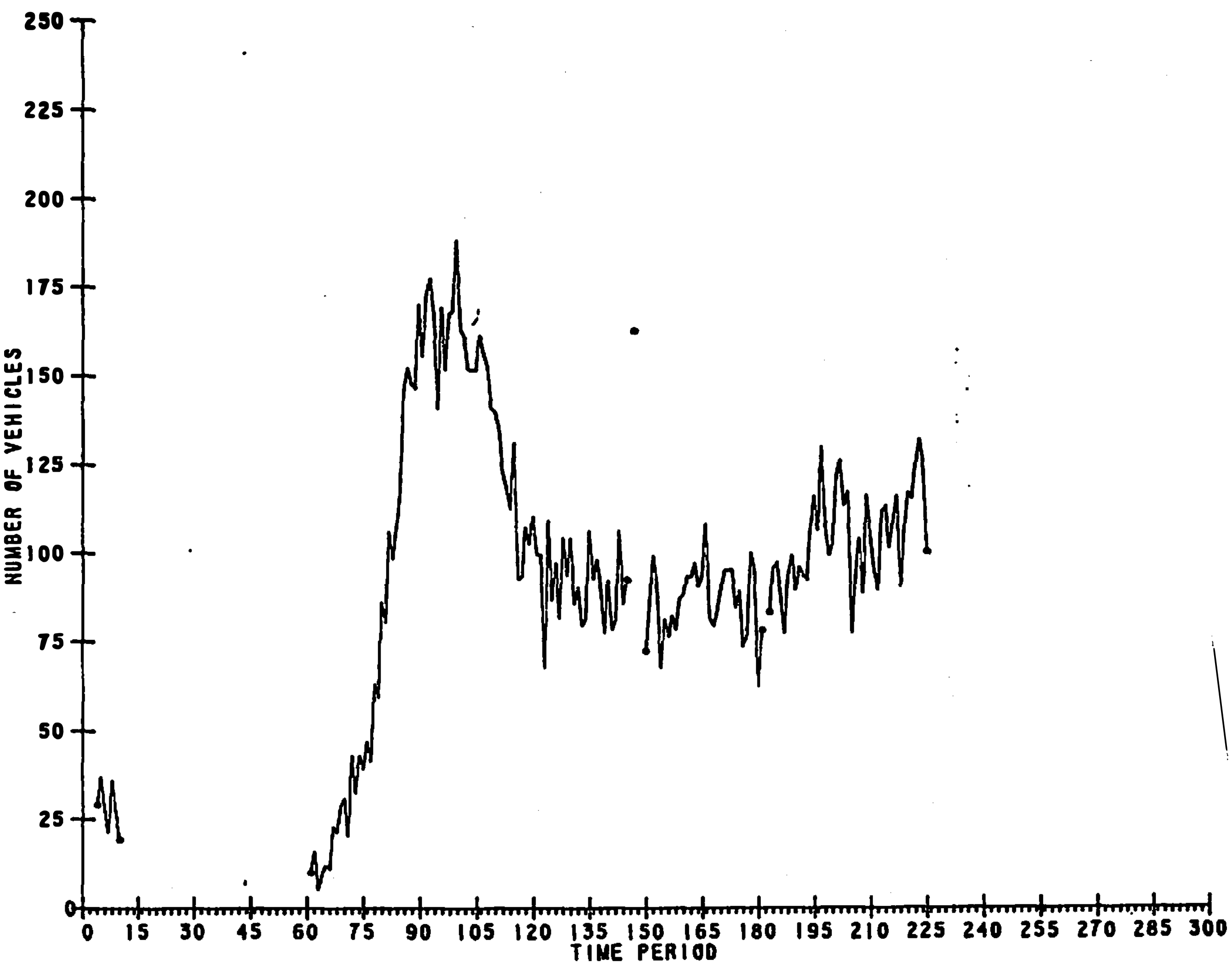

FIGURE D10. VOLUME VS. TIME, SENSOR 2, WEDNESDAY, DAY 24

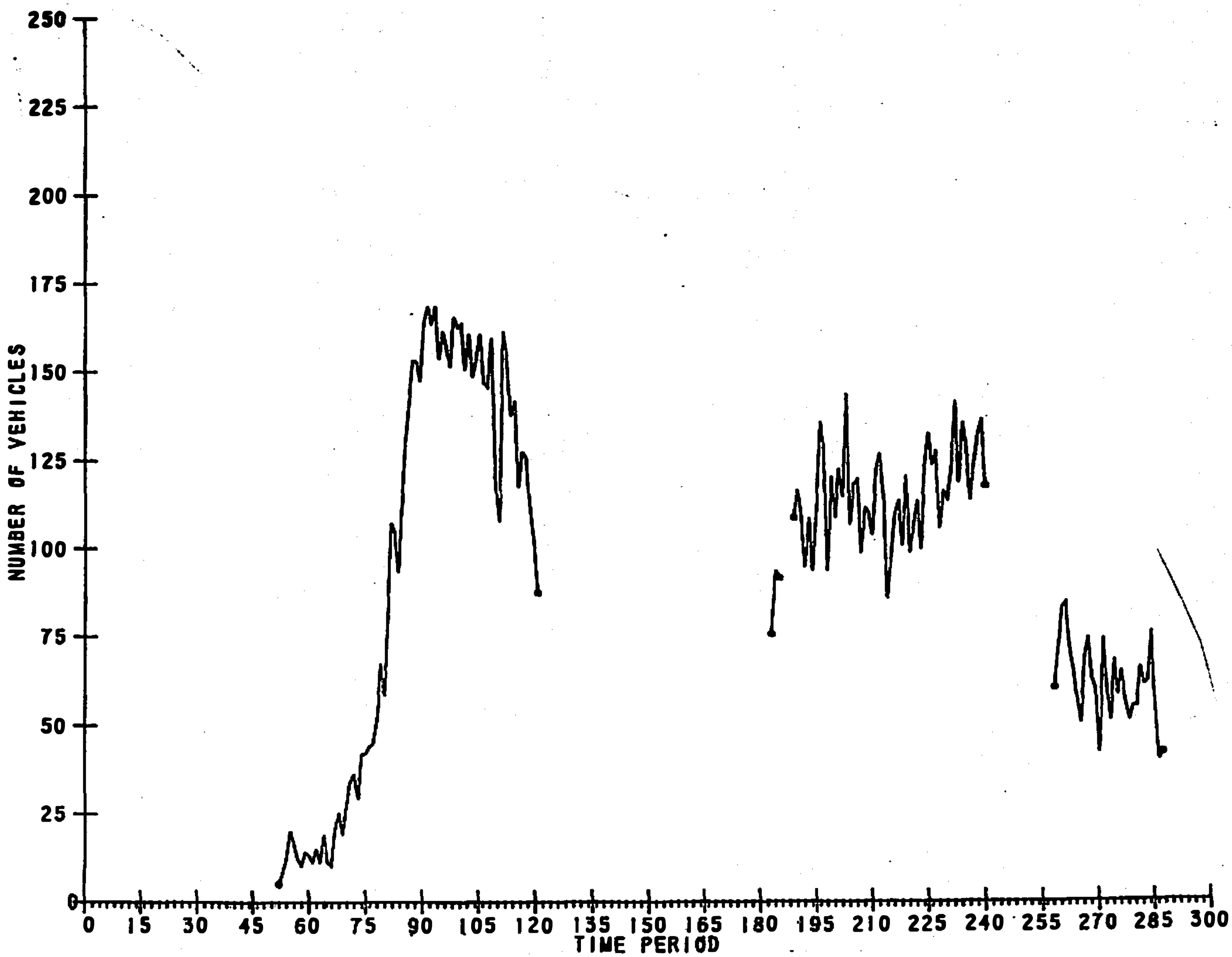

FIGURE D11. VOLUME VS. TIME SENSOR 2, THURSDAY, DAY 39

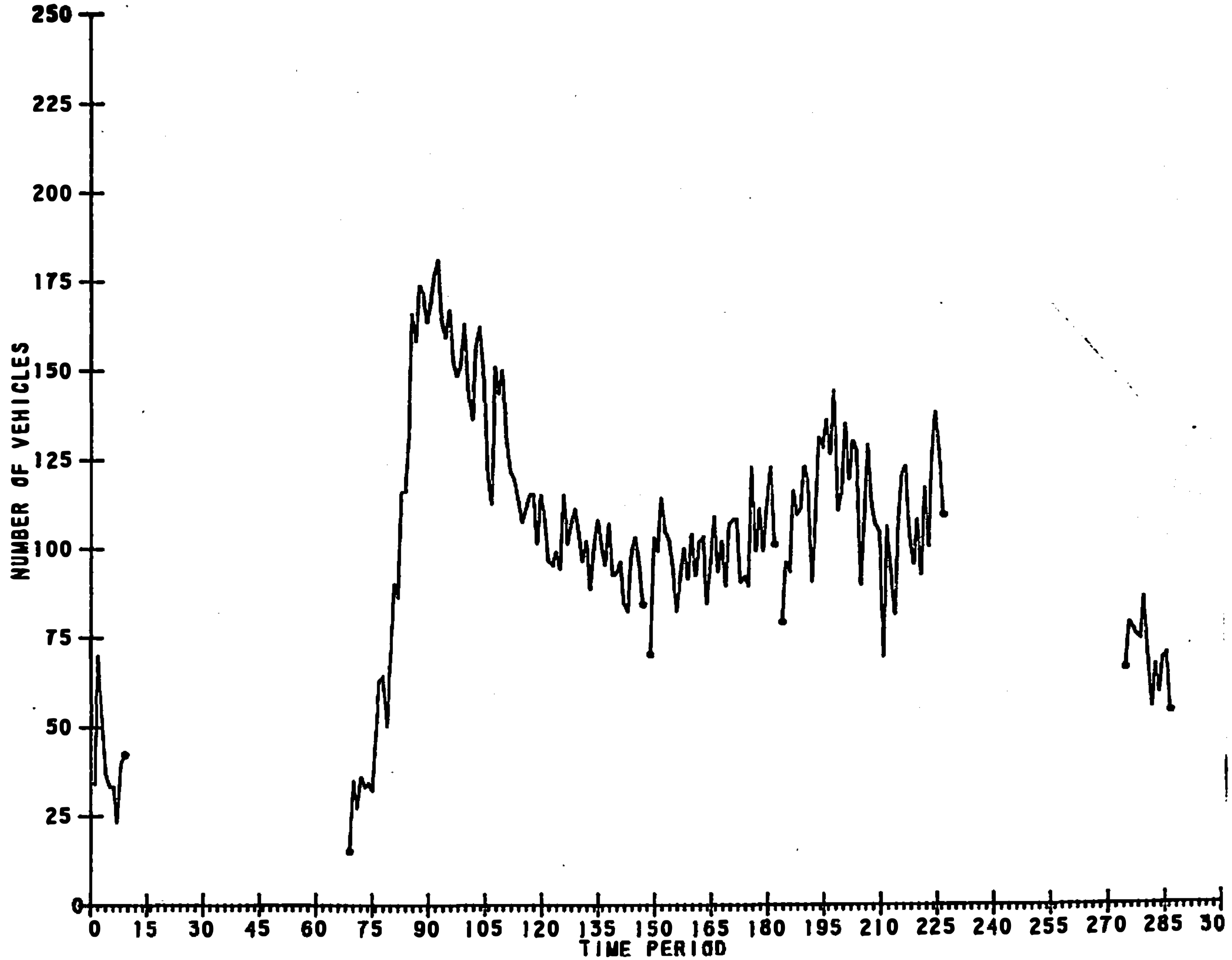

FIGURE D12. VOLUME VS. TIME, SENSOR 2, FRIDAY, DAY 5

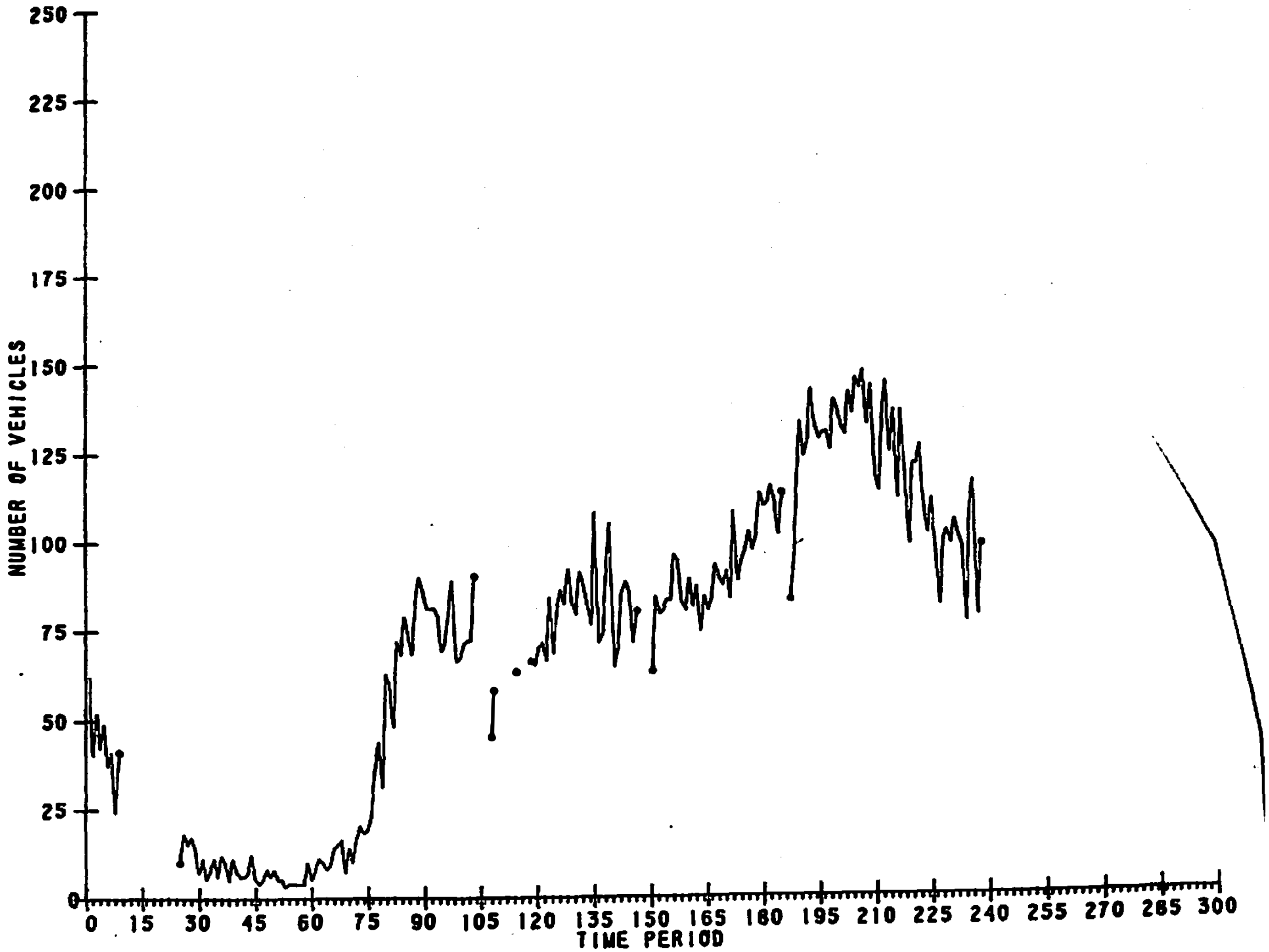

FIGURE D13. VOLUME VS. TIME, SENSOR 3, MONDAY, DAY 8

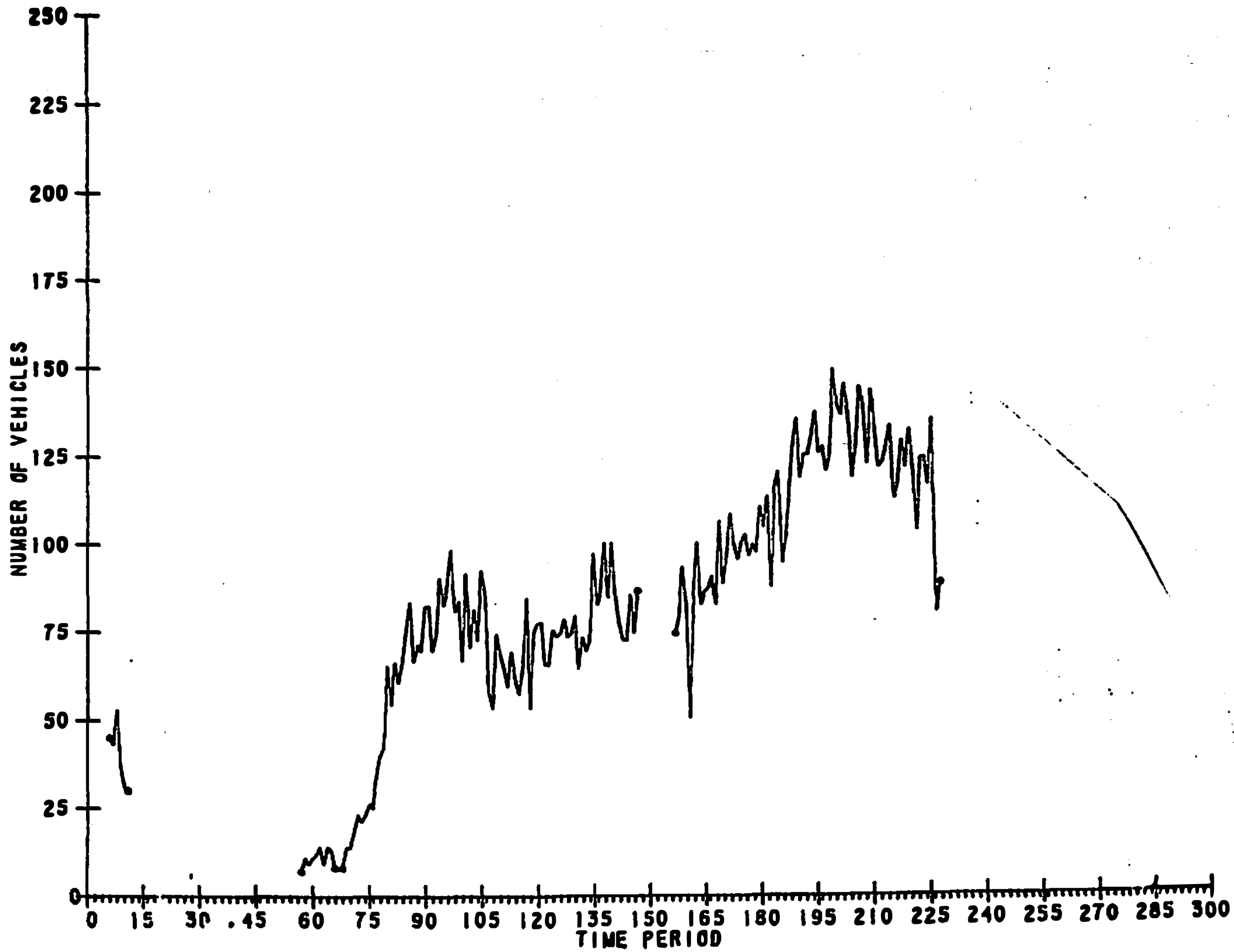

FIGURE D14. VOLUME VS. TIME, SENSOR 3, TUESDAY, DAY 23

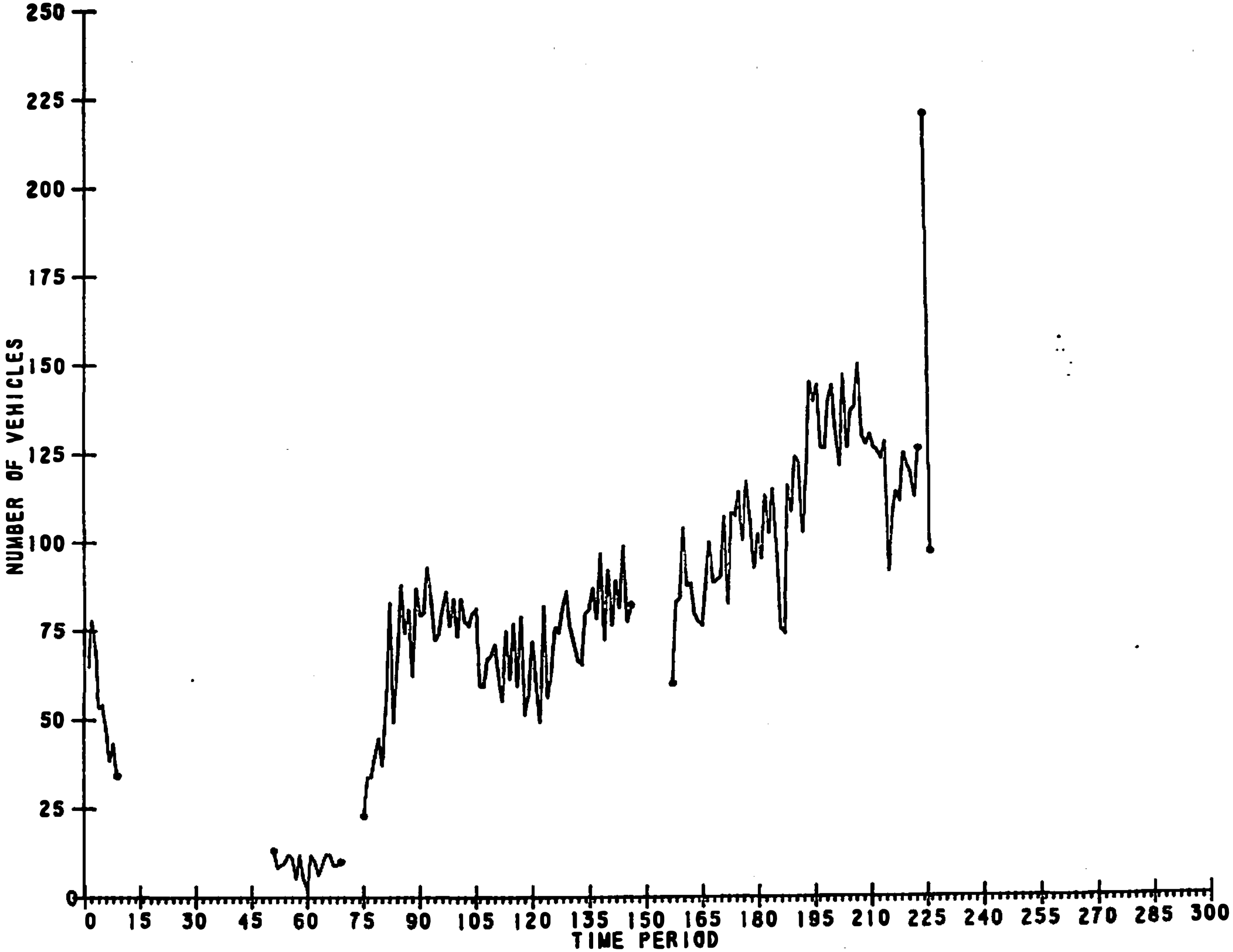

FIGURE D15. VOLUME VS. TIME, SENSOR 3, WEDNESDAY, DAY 17

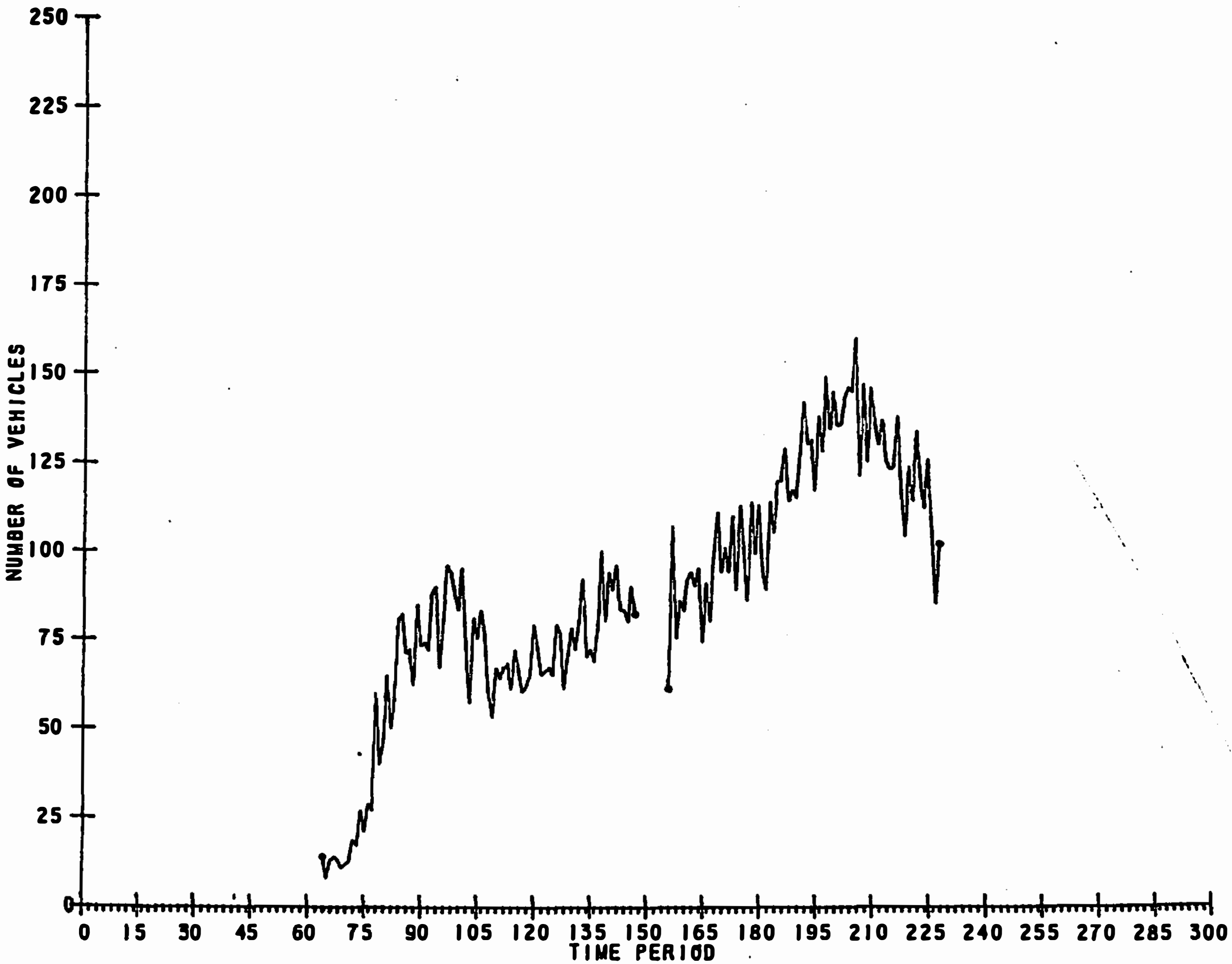

FIGURE D16. VOLUME VS. TIME, SENSOR 3, THURSDAY, DAY 32

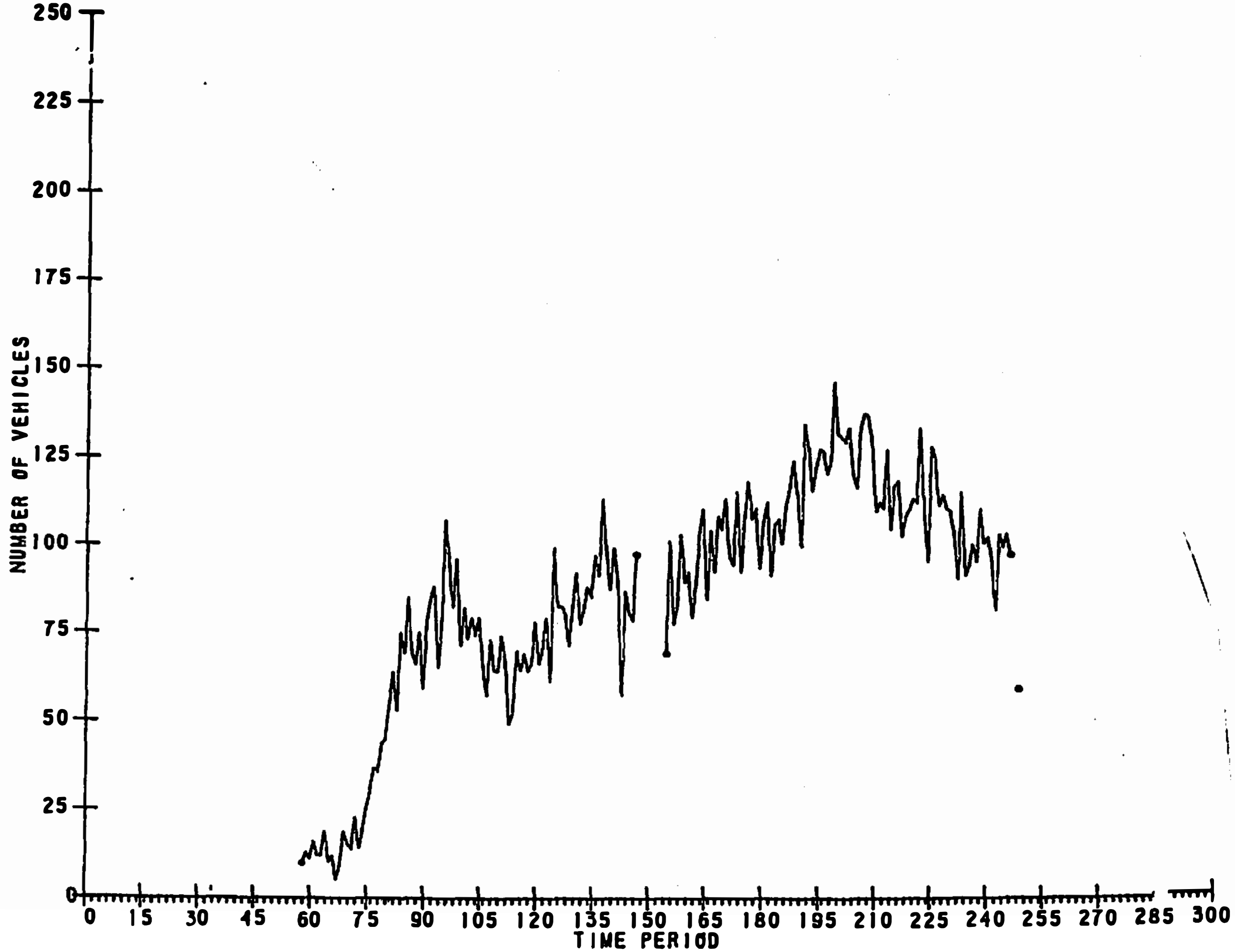

FIGURE D17. VOLUME VS. TIME, SENSOR 3, FRIDAY, DAY 26

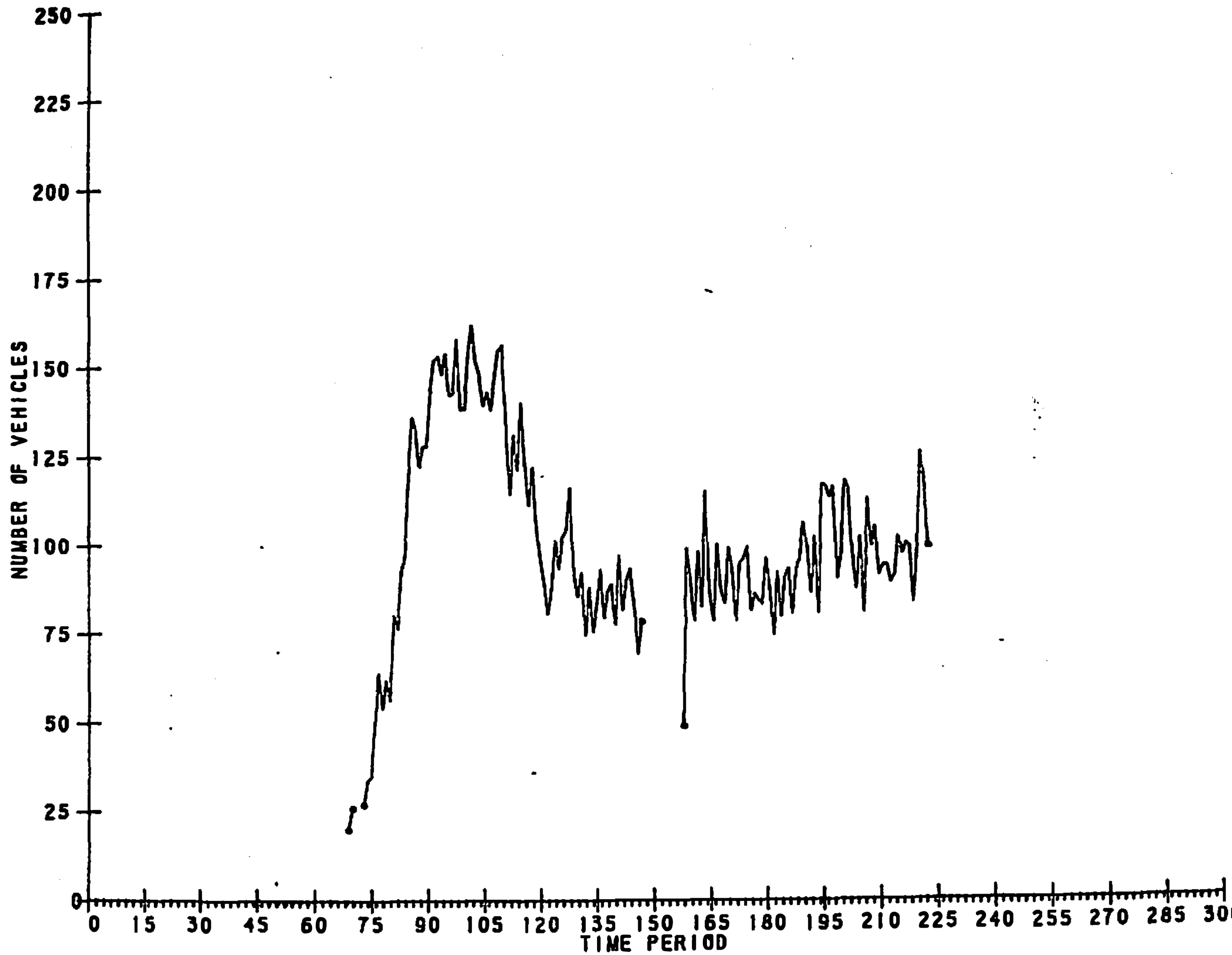

FIGURE D18. VOLUME VS. TIME, SENSOR 4, MONDAY, DAY 36

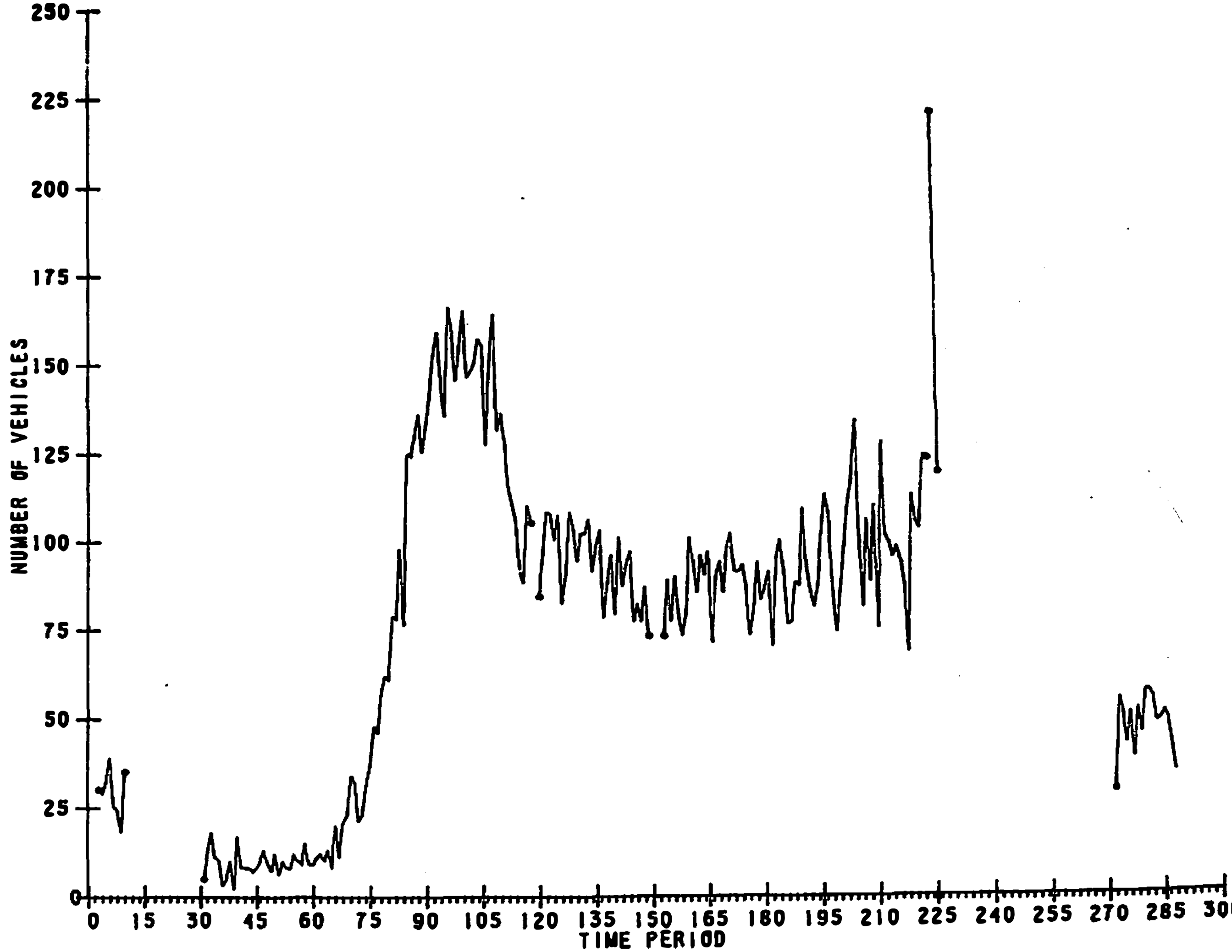

FIGURE D19. VOLUME VS. TIME, SENSOR 4, TUESDAY, DAY 16

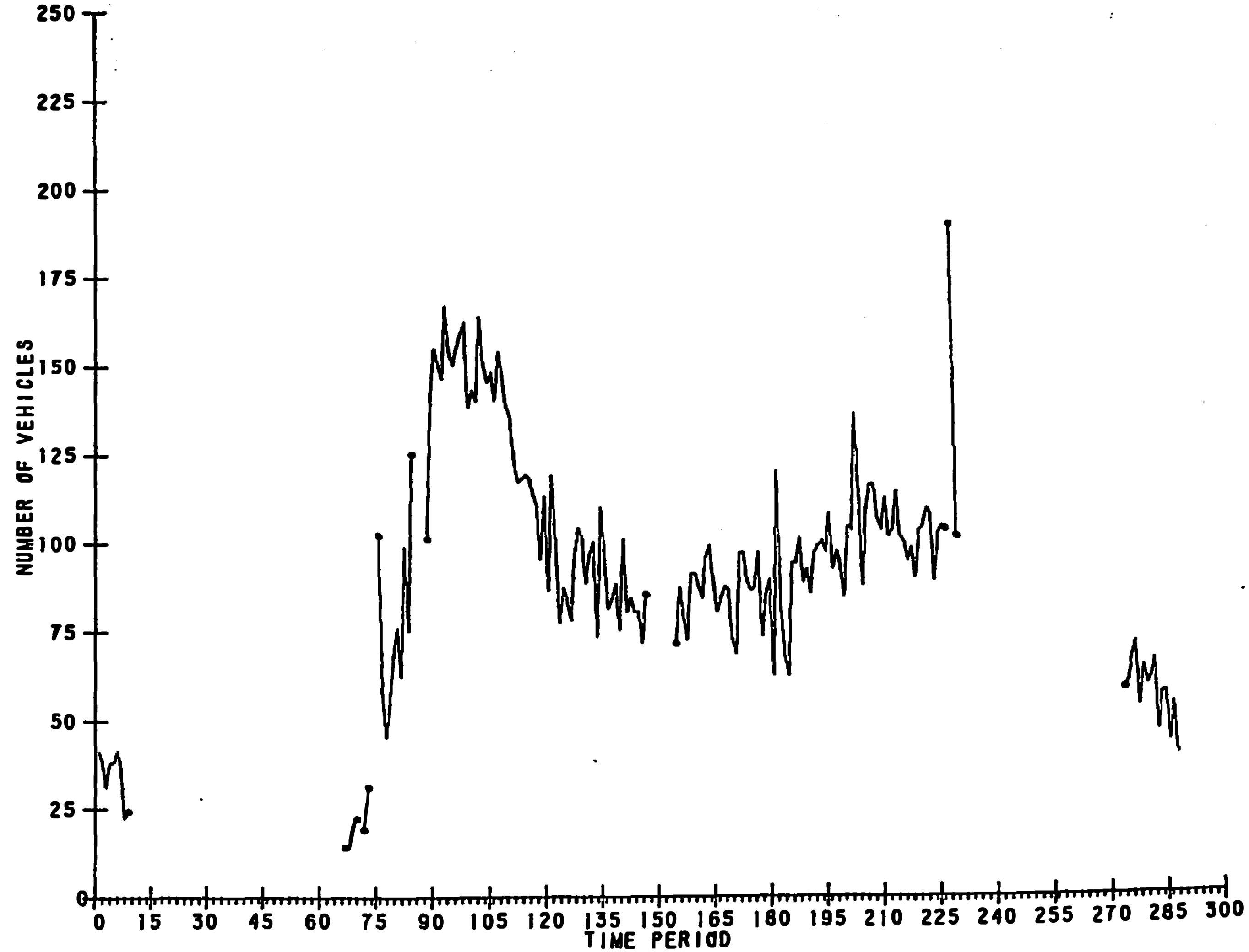

FIGURE D20. VOLUME VS. TIME, SENSOR 4, WEDNESDAY, DAY 3

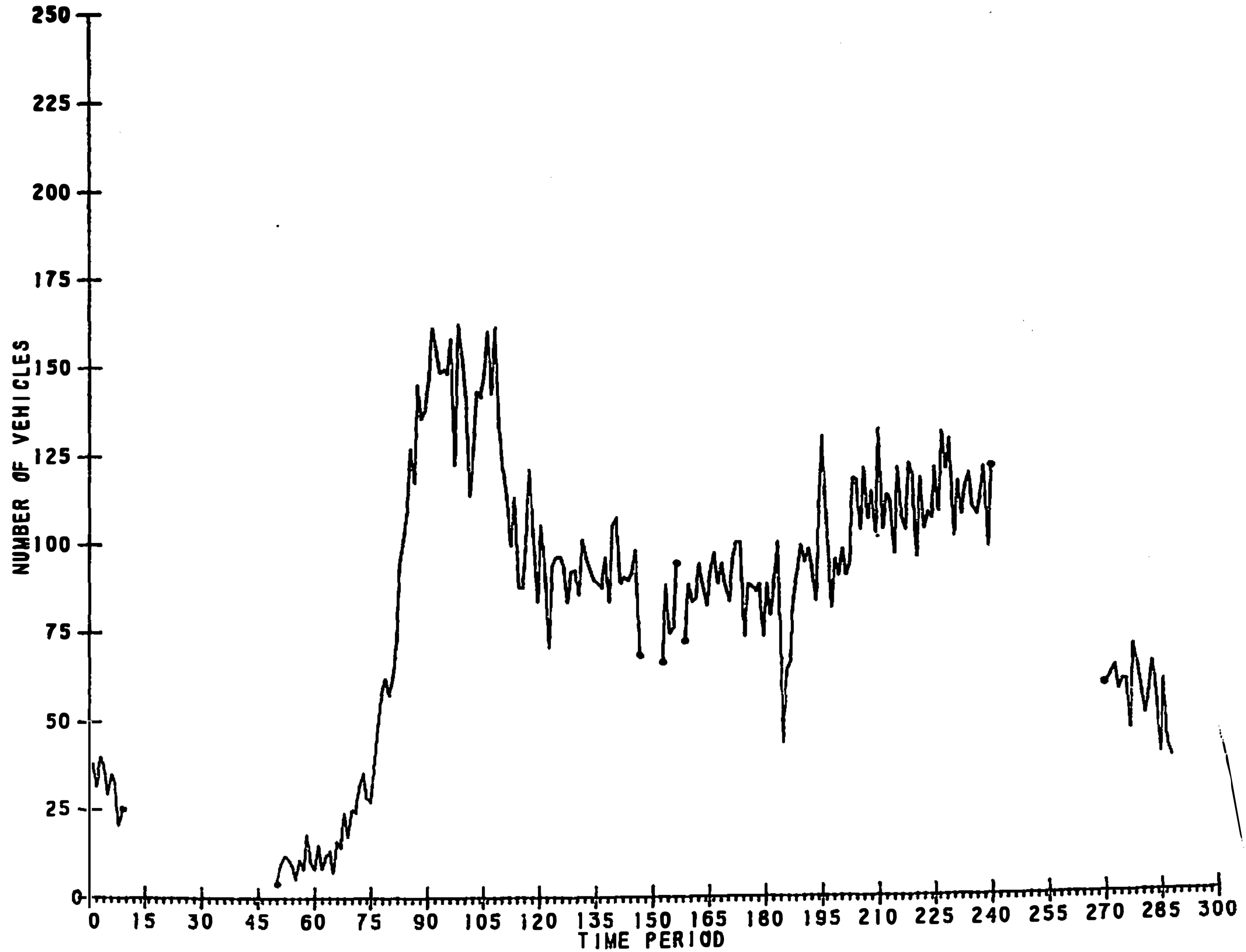

FIGURE D21. VOLUME VS. TIME, SENSOR 4, THURSDAY, DAY 4

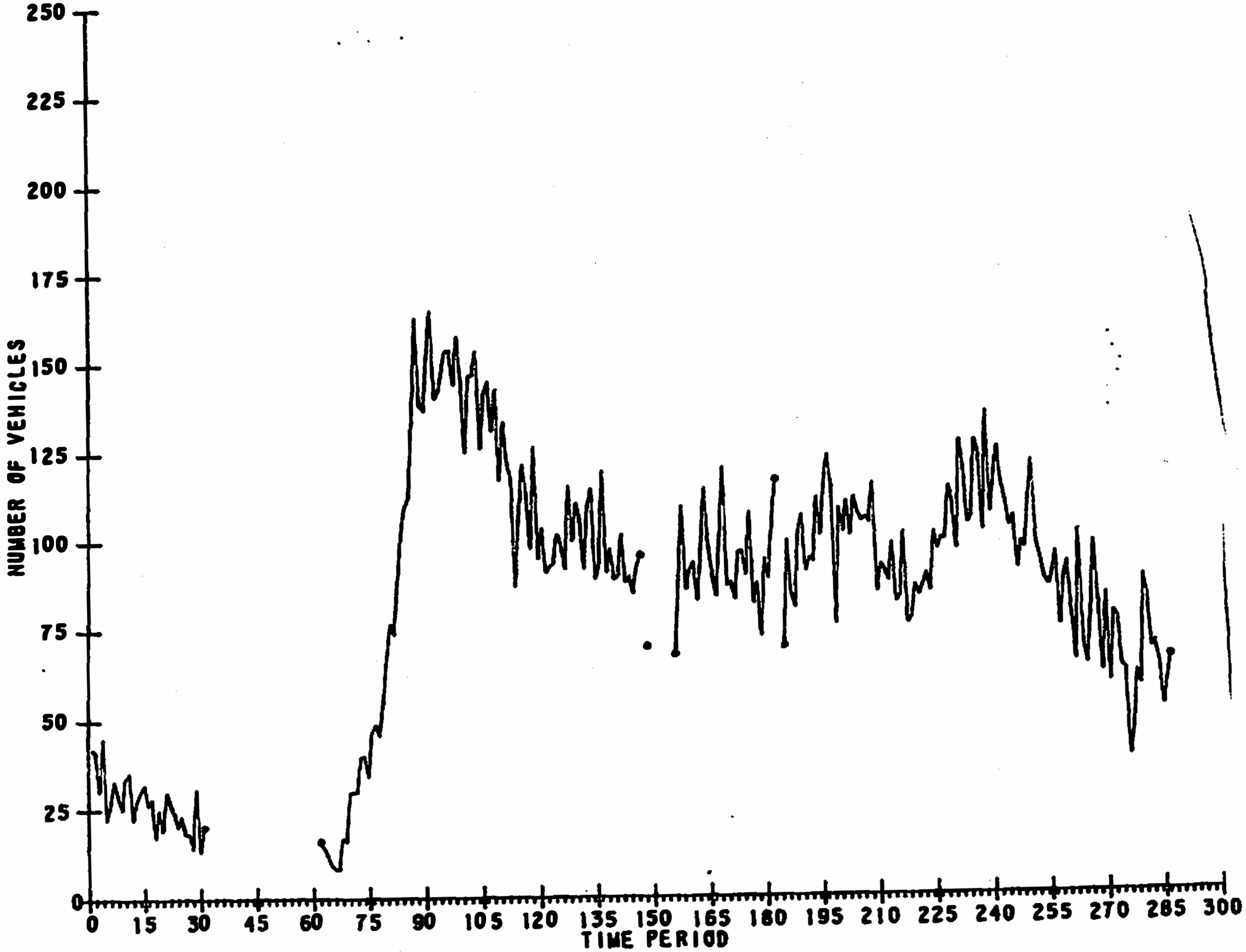

FIGURE D22. TIME VS. VOLUME, SENSOR 4, FRIDAY, DAY 12

APPENDIX E

RELATIONSHIP OF UTCS SECOND AND THIRD GENERATION PREDICTORS TO ARIMA PREDICTORS

In this appendix we show that:

1. The third generation UTCS traffic predictor is of the same form as a Box-Jenkins optimal predictor for an ARIMA(1,1,1) process. In this case the ARIMA(1,1,1) form predictor acts on the sequence of raw traffic volume.
2. The second generation UTCS traffic predictor considered as a predictor of the difference between current day counts and historical average counts is likewise of the form of an optimal ARIMA(1,1,1) predictor.
3. If the historical average traffic counts for use in the second generation UTCS predictor are derived as an exponentially weighted moving average (over past days), then the second generation predictor is equivalent in form to an optimal (seasonal) ARIMA(1,1,1) x $(0,1,1)_{288}$ predictor acting on the raw traffic counts.

E.1 THIRD GENERATION AS ARIMA(1,1,1)

First to derive the connection between third generation and ARIMA predictors, we shall be making transformations in notation in order to achieve conformity with Box-Jenkins notations (and to allow us to make apparent the points of similarity with the second generation UTCS predictor).

First note that equation 2.3 of Part 1 (Sec. 4.2) can be brought into a simpler form using the specific equations (22),

(19) and (18) of that same section:

$$(E.1)\quad \hat{V}_k(i+j) = (1 - \alpha_j)\hat{\mu}_k(i) + \alpha_j V_k(i)$$

In this equation $V_k(i)$ is the volume in time period i at sensor k. $\hat{V}_k(i+j)$ is the predicted volume at time period i+j, and $\mu_k(i)$ is an exponential average of the volume including the current value:

$$(E.2)\quad \hat{\mu}_k(i) = \beta\hat{\mu}_k(i-1) + (1-\beta)V_k(i) \text{ (equation 17, Part 1).}$$

Using E2, E1 can be further transformed into:

$$(E.3)\quad \hat{V}_k(i+j) = (1-\alpha_j)\beta\hat{\mu}_k(i-1) + (\alpha_j + (1-\beta)(1-\alpha_j))V_k(i)$$

Now we make the transformations of notation. Let Z_t denote the volume count during time period t (at the specific sensor) so Z_t replaces $V_k(i)$ (with t replacing i). The exponential average of Z_t will be denoted by $\overline{Z}_t$ so that $\overline{Z}_t$ replaces $\hat{\mu}_k(i)$. The predicted value of Z_{t+j} as determined at time t will be denoted by $\hat{Z}_{t+j}$ and so $\hat{Z}_{t+j}$ replaces $\hat{V}_k(t+j)$. The symbol Θ replaces the symbol β (to denote the same quantity). Finally, λ will replace the quantity $(1-\alpha_j)\ \beta$, or to summarize:

$$(E.4)\quad \hat{Z}_{t+j} = \lambda\overline{Z}_{t-1} + (1-\lambda)Z_t$$

$$(E.5)\quad \overline{Z}_t = \Theta Z_{t-1} + (1-\Theta)Z_t$$

$$(E.6)\quad \Theta = \beta$$

$$(E.7)\quad \lambda = \beta(1-\alpha_j)$$

Equations E4 and E5 define a predictor of the same form as the B-J optimal ARIMA(1,1,1) predictor as is readily verified by referring to Sec. 5.4.6 of Ref. 1.

E.2 SECOND GENERATION UTCS PREDICTOR AS ARIMA(1,1,1)

Next we show the connection between the second generation UTCS predictor and the optimal ARIMA(1,1,1) predictor. Again the UTCS notation will be brought into conformity with the notation to be used for common analysis.

From Sec. 4.1.1 of Part 1 equation 5 can be rewritten (using equations 1, 2, 3 and 4):

$$(E.8)\quad V(t) - m(t) = C(t-1)(\alpha+\gamma) + (1-\alpha-\gamma)r(t-1)$$

and equation (2) (Part 1) can be written (using the fact that $\beta = 1 - \alpha$):

$$(E.9)\quad C(t) = \alpha C(t-1) + (1-\alpha)r(t)$$

Now $r(t) = f(t) - m(t)$ where $f(t)$ is the actual traffic count during time interval t and $m(t)$ is the historical average, i.e., an average (smoothed or unsmoothed, see Appendix F) of traffic counts for the same time of day taken over a number of previous days. The difference, $V(t)$, between the actual and historical counts is what is predicted by the ARIMA(1,1,1) type predictor. The prediction of volume at time t is $V(t)$, therefore $V(t)-m(t)$ is the prediction of $r(t)$ made at time t-1. Thus replacing $r(t)$ by Z_t, $C(t)$ by $\bar{Z}_t$ and $V(t)-m(t)$ by $\hat{Z}_t$ we have:

$$(E.10)\quad \hat{Z}_{t+1} = \lambda\bar{Z}_{t-1}+(1-\lambda)Z_t$$

(E.11) $\bar{Z}_t = \Theta\bar{Z}_{t-1} + (1-\Theta)Z_t$

where

(E.12) $\lambda = 1 - \alpha - \gamma$

and

(E.13) $\theta = \alpha$

Now E10 and E11 are identical to E4 and E5 (with j=1) and so are of the form of optimum ARIMA(1,1,1) predictors.

The parameters λ and Θ like Z_t stand for wholly different quantities than they did in the discussion of UTCS Second Generation predictors earlier.

E.3 SECOND GENERATION AS ARIMA(1,1,1) x $(0,1)_{288}$

Next we show that the aspect of the second generation predictor whereby it works on the difference between actual volume and historical average volume is itself to be found in Box and Jenkins Seasonal ARIMA models. More specifically, if the historical average is formed as an exponentially weighted moving average (to be explained below), then the Second Generation UTCS predictor becomes of the form of an optimal ARIMA(1,1,1) x $(0,1,1)_{240}$ predictor. Before we give the demonstration of this, which is lengthy, we note that the result is not used elsewhere in this report. The primary significance of the result is to suggest that the UTCS Second Generation method of incorporating historical data is consistent with the philosophy of Box and Jenkins who devote a whole chapter to seasonal models. This suggests that dealing with the derived time series consisting of the difference between the actual and historical average volumes is well founded.

Denote the actual traffic volume at time t by Z_t (like in the Third Generation and unlike the treatment of the Second Generation case discussed just above where Z_t represents the difference between the actual and the historical average volume).

It then turns out that for Z_t representing five minute volume counts that Z_{t-288} is the volume count for the same time of day as Z_t (since these are 288 five-minute time periods in one 24-hour day).

Let us denote the exponentially weighted historical average by h_t (h_t replacing m(t)), then

$$\text{(E.14)} \quad h_t = qh_{t-288} + (1-q)Z_{t-288}$$

The quantity q determines the "memory" of the exponential moving average which determines h_t. Explicitly:

$$\text{(E.15)} \quad h_t = (1-q) \sum_{k=1}^{\infty} q^k h_{t-k\cdot 288}$$

A rough idea of the number of past days data which enter into the exponential average h_t is given by $\frac{1}{1-q}$. Thus if q = .95, then $\frac{1}{1-q}$ = 20 and roughly speaking 20 days enter into the historical average. The weight of the 21st previous day's volume (at the same particular time of day, 5-minute period) enters in h_t with a weight which is only 0.36 times the contribution for the same five-minute time period on the first previous day. The 41st day in the past has a contribution equal to .36 that of the 21st day or $(.36)^2$ = .129 times that of the immediately preceding day, etc.

Exponential averages are a common feature in ARIMA predictors; they have, as we see, a smoothly decreasing memory of the more remote past. (However, in some cases we are dealing with memories of minutes rather than days.) The exponential moving average is also useful because of its ease of computation. Considerations should be given such that the historical average for use in the UTCS Second Generation predictor be produced as an exponential moving average. In any case, we proceed to analyze the predictor if it is so constructed.

Letting h_t be given by E14 or E15, we now define:

$$\text{(E.16)}\quad U_t = Z_t - h_{t-288}$$

(Thus U_t here is the equivalent of r(t) in the treatment in Part 1, Sec. 4.1.) We have shown that the Second Generation predictor is equivalent to predicting U_t using an ARIMA(1,1,1) predictor. Equations E10 and E11 give the form of such a predictor but also we have that:

$$\text{(E.17)}\quad \hat{Z}_{t+1} = (1-\lambda)Z_t + (\lambda-\Theta)Z_{t-1} + \Theta\hat{Z}_t$$

is equivalent to E10 and E11, as is easily shown by eliminating $\overline{Z}_t$ between equations E10 and E11.* (See also B-J, Ref. 1, Eq.

---

* a) $\hat{Z}_{t+1} = \lambda\overline{Z}_{t-1} + (1-\lambda)Z_t$

b) $\overline{Z}_t = \Theta\overline{Z}_{t-1} + (1-\Theta)Z_t$

c)→from (a) $\hat{Z}_t = \lambda\overline{Z}_{t-1} + (1-\lambda)Z_{t-1}$

d)→substitute (b) into (a) $\hat{Z}_{t-1} = \lambda(\Theta Z_{t-2} + (1-\Theta)Z_{t-1}) + (1-\lambda)Z_t$

subtract Θx(c) from (d) $\hat{Z}_{t+1} - \Theta\hat{Z}_t = \lambda(\Theta\overline{Z}_{t-2} + (1-\Theta)Z_{t-1})$

$+ (1-\lambda)Z_t - \Theta\lambda Z_{t-2} - \Theta(1-\lambda)Z_{t-1} = (\lambda-\Theta)Z_{t-1} + (1-\lambda)Z_t$

5.4.22; remember $a_t = Z_t - \hat{Z}_t$, $\lambda = \Theta - \phi$).

Since we are now dealing with the difference between the actual volume (now denoted by Z_t) and the historical average h_t, equation E17 becomes:

$$\hat{U}_{t+1} = (1-\lambda)U_t + (\lambda-\Theta)U_{t-1} + \Theta\hat{U}_t$$

Now substituting for U_t from equation E16 we have:

$$\text{(E.18)} \quad \hat{Z}_{t+1} - h_{t+1-288} = (1-\lambda)(Z_t - h_{t-288}) + (\lambda-\Theta)(Z_{t-1} - h_{t-1-288}) + \Theta(\hat{Z}_t - h_{t-288})$$

Equation E18 gives the predicted value of $\hat{Z}_{t+1}$ given Z_t, Z_{t-1}, Z_{t-2}, etc., including the historical averages as defined in E14 or E15 this prediction is established according to the UTCS method using historical averages. Equation E18 is merely an algebraic and notational transformation of equation 23 from Part 1. Our next task is to set down the form of an ARIMA(1,1,1) x $(0,1,1)_{288}$ 1-step predictor and show that it is of the same form as equation E18. From Sec. 9.1 of B-J (Ref. 1), we have that an ARIMA(1,1,1) x $(0,1,1)_{288}$ model can be expressed:

$$\text{(E.19)} \quad (1-B^{288})(1-\phi B)(1-B)Z_t = (1-qB^{288})(1-\Theta B)a_t$$

where q, Θ, $\phi = \Theta - \lambda$ are arbitrary parameters (satisfying certain restrictions; see Ref. 1).

The symbol B represents the backward shift operation; $BZ_t = Z_{t-1}$. Expanding out equation E19 we obtain the equation that it is a shorthand for:

$$(E.20)\quad Z_t - (1+\Phi)Z_{t-1} + \Phi Z_{t-2} - Z_{t-288} + (1-\Phi)Z_{t-1-288}$$

$$- \Phi Z_{t-2-288} = a_t - \Phi a_{t-1} - q a_{t-288} + q\Theta a_{t-1-288}$$

Using the standard BJ rules (i.e., substituting $Z_\tau - \hat{Z}_\tau$ for a_τ; see Eq. 5.1.22, Ref.1) for producing optimal 1-step predictors from ARIMA models, we obtain:

$$(E.21)\quad \hat{Z}_t = (1+\Phi)Z_{t-1} - \Phi Z_{t-2} + Z_{t-288}(1-\Phi)Z_{t-1-288}$$

$$+ \Phi Z_{t-2-288} - \Theta(Z_{t-1}-\hat{Z}_{t-1}) - q(Z_{t-288}-\hat{Z}_{t-288})$$

$$+ q\Theta(Z_{t-1-288}-\hat{Z}_{t-1-288})$$

It is now a matter of algebra to show that E21 is equivalent to E18 together with E16 (remember that $\Phi=\Theta-\lambda$). The parameter h_t must be eliminated between E18 and E16. The chore is rather tedious but the principle is the same as deriving E17 from E10 and E11. The trick is to subtract from equation E18 the same equation evaluated at t = t-288 and multiply by q:

$$(E.22)\quad \hat{Z}_{t+1} - q\hat{Z}_{t+1-288} - h_{t+1-288} + qh_{t+1-2\cdot 288})$$

$$= (1-\lambda)(Z_t - Z_{t-288} - h_{t-288} + qh_{t-2\cdot 288})$$

$$+ (\lambda-\Theta)(Z_{t-1} - Z_{t-1-288} - h_{t-1-288} - qh_{t-1-2\cdot 288})$$

$$+ \Theta(\hat{Z}_t - \hat{Z}_{t-288} - h_{t-288} + qh_{t-2\cdot 288})$$

Now use the fact that $h_t - qh_{t-288} = (1-q)Z_t$ (obtained immediately from equation 16) to eliminate adjacent pairs of terms of this form, e.g., $-h_{t+1-288} + qh_{t+1-2\cdot 288}$ is replaced by $-(1-q)Z_{t+1-288}$. Equation E21 (evaluated at t=t+1) results.

Equation E21 can thus be regarded as the explicit form of the UTCS Second Generation predictor when the historical average is calculated as an exponentially weighted moving average. This has advantages of computational simplicity as well as theoretical reasonableness. The rather high degree of complexity (cf. Eq. 3 of main text, Part 2, which specifies the same predictor) results from eliminating all implicit intermediate quantities -- for actual implementation the implicit forms are simpler and more convenient. In this discussion Z_t has represented the actual 5-minute traffic volume count. Earlier in this Appendix and elsewhere throughout the report (as noted in context), Z_t represents the <u>difference</u> between the actual volume and the historical average <u>if</u> historical data are being used (e.g., Second Generation) -- <u>but</u> Z_t always represents the volume data themselves <u>if</u> historical data are <u>not</u> being used (e.g, Third Generation). The reason for the shift is so that Z_t always represents the basic time series to be predicted (based on its current and past values), whatever the context.

APPENDIX F

FOURIER SMOOTHING OF HISTORICAL DATA

This appendix discusses the methods used to obtain the coefficients for the Fourier smoothing curves, examines the results obtained by varying the number of Fourier coefficients and partially tests the results using predicting algorithms.

F.1 METHODS FOR OBTAINING FOURIER COEFFICIENTS

The data as discussed in Appendix D, has daily general shapes peculiar to each sensor (i.e., peaks and valleys occur at about the same time each day with some variation in their heights and depths). Hence, it is reasonable to use a mathematical expression in sines and cosines with the length of a day (288 5-minute time intervals, see Appendix D) as their period to represent the common daily trend plus a random error $\varepsilon_{\ell j}$.

This is written as follows:

F1) $$V_{\ell j} = \tilde{V}_{\ell} + \varepsilon_{\ell j}$$

where $V_{\ell j}$ is the volume counted in the ℓ'th interval of time of the j'th day, and

F2) $$\tilde{V}_{\ell} = a_o + \sum_{i=1}^{L} [a_i \cos(\frac{2\pi i\ell}{288}) + b_i \sin(\frac{2\pi i\ell}{288})]$$

where a_o, a_i's and b_i's are the 2L + 1 "Fourier" coefficients. A standard model is to assume that the expected value of $\varepsilon_{\ell j}$ is zero. The $\varepsilon_{\ell j}$'s are uncorrelated and have equal variance σ^2.

The unbiased estimate of σ^2 i.e., pure noise variance S^2p is given by

$$\text{F3)} \quad S^2p = \sum_{\ell=1}^{288} \sum_{j=1}^{n_\ell} (\bar{V}_\ell - V_{\ell j})^2 / \sum_{\ell=1}^{288} (n_\ell - 1) ,$$

where n_ℓ is the number of days which have data in the ℓ'th interval and where

$$\text{F4)} \quad \bar{V}_\ell = \sum_{j=1}^{n_\ell} V_{\ell j}/n_\ell$$

Thus, $\bar{V}_\ell$ is the average count in the ℓ'th time interval. Since it is convenient to work with $\bar{V}_\ell$ and n_ℓ, equation (F1) is averaged over the n_ℓ days, i.e.,

$$\text{F5)} \quad \bar{V}_\ell = \frac{n_\ell \tilde{V}_\ell}{n_\ell} + \frac{\sum_{j=1}^{n_\ell} \varepsilon_{\ell j}}{n_\ell} = \tilde{V}_\ell + \bar{\varepsilon}_\ell$$

With VAR=variance and COV=covariance, then

$$\text{F6)} \quad \text{VAR}(\bar{V}_\ell) = \text{VAR}(\tilde{V}_\ell) + 2\text{COV}(\tilde{V}_\ell, \bar{\varepsilon}_\ell) + \text{VAR}(\bar{\varepsilon}_\ell)$$

$$= 0 + 0 + \frac{\sigma^2}{n_\ell} = \frac{\sigma^2}{n_\ell} ,$$

and

$$\text{F7)} \quad \text{COV}(\bar{V}_\ell, \bar{V}_{\ell'}) = \text{COV}(\tilde{V}_\ell, \tilde{V}_{\ell'}) + \text{COV}(\tilde{V}_\ell, \bar{\varepsilon}_{\ell'})$$

$$+ \text{COV}(\tilde{V}_{\ell'}, \bar{\varepsilon}_\ell) + \text{COV}(\bar{\varepsilon}_\ell, \bar{\varepsilon}_{\ell'}) = 0$$

Thus, the (weighted) sum of squares is given by

F8) $SS = \sum_{\ell=1}^{288} n_\ell(\overline{V}_\ell - \tilde{V})^2 = \sum_{\ell=1}^{288} n_\ell(\overline{\varepsilon}_\ell)^2$

(Ref.: The Analysis of Variance by Scheffé, pages 19 and 20.)
The sum of squares is minimized with respect to a_o, a_i and b_i by differentiating equation (F8) with respect to a_o, a_i and b_i and setting the resulting partial derivative equal to zero. This leads to the following equations in the unknowns a_o, a_i and b_i.

$$a_o \sum_{\ell=1}^{288} n_\ell + \sum_{i=1}^{L} a_i \sum_{\ell=1}^{288} n_\ell \cos\left(\frac{2\pi i\ell}{288}\right) + \sum_{i=1}^{L} b_i \sum_{\ell=1}^{288} n_\ell \sin\left(\frac{2\pi i\ell}{288}\right)$$

$$= \sum_{\ell=1}^{288} n_\ell \overline{V}_\ell \quad ,$$

$$a_o \sum_{\ell=1}^{288} n_\ell \cos\left(\frac{2\pi I\ell}{288}\right) + \sum_{i=1}^{L} a_i \sum_{\ell=1}^{288} n_\ell \cos\left(\frac{2\pi i\ell}{288}\right)\cos\left(\frac{2\pi I\ell}{288}\right)$$

F9) $$+ \sum_{i=1}^{L} b_i \sum_{\ell=1}^{288} n_\ell \sin\left(\frac{2\pi i\ell}{288}\right)\cos\left(\frac{2\pi I\ell}{288}\right) = \sum_{\ell=1}^{288} n_\ell \overline{V}_\ell \cos\left(\frac{2\pi I\ell}{288}\right) \quad ,$$

$$a_o \sum_{\ell=1}^{288} n_\ell \sin\left(\frac{2\pi I\ell}{288}\right) + \sum_{i=1}^{L} a_i \sum_{\ell=1}^{288} n_\ell \cos\left(\frac{2\pi i\ell}{288}\right)\sin\left(\frac{2\pi I\ell}{288}\right)$$

$$+ \sum_{i=1}^{L} b_i \sum_{\ell=1}^{288} n_\ell \sin\left(\frac{2\pi i\ell}{288}\right)\sin\left(\frac{2\pi I\ell}{288}\right) = \sum_{\ell=1}^{288} n_\ell \overline{V}_\ell \sin\left(\frac{2\pi I\ell}{288}\right) \quad ,$$

$$1 \le I \le L.$$

The variance estimate (i.e., Fourier residual variance) derived from the above sum of squares is given by

$$F10) \quad S^2_{2L+1} = \frac{\sum_{\ell=1}^{288} n_\ell(\tilde{V}_\ell - \bar{V}_\ell)^2}{288 - (2L+1)}$$

The expected value of S^2_{2L+1}, $E(S^2_{2L+1})$, is

$$\sigma^2 + \left(\sum_{\ell=1}^{288} n_\ell [E(V_\ell) - \tilde{V}_\ell]^2\right)/[288 - (2L+1)],$$

where $E(V_\ell)$ is the expected value of V_ℓ i.e., the true means of at the time interval ℓ. Thus, when the second term is zero, $E(S^2_{2L+1}) = \sigma^2$ and the data has a good fit in $\tilde{V}_\ell$. Thus, S^2_{2L+1} can be compared to the pure noise or error variance S^2P (equation F3). The expected value of S^2p is known to be equal to σ^2 regardless of the fit used (Ref. Applied Linear Statistical Models by Neter and Wasserman pages 117-119). The F statistic given by:

$$F11) \quad F = S^2_{2L+1}/S^2p$$

is used to test the fit of $\tilde{V}_\ell$. (An excessively large value means an unsatisfactory fit.)

F.2 THE EFFECTS OF VARYING THE NUMBERS OF FOURIER COEFFICIENTS

For Part 1 of the data (see Appendix D), the number of Fourier coefficients considered was 11 through 21, 31 and 41 for all sensors. In addition, for sensor 4, calculations for 51 and 81 coefficients were made. The results are shown in Figure F1. This shows the Fourier residual standard deviation S_{2L+1} (section F.1) versus the number of Fourier coefficients. The pure error standard deviation Sp is plotted in Figure F1 to show the

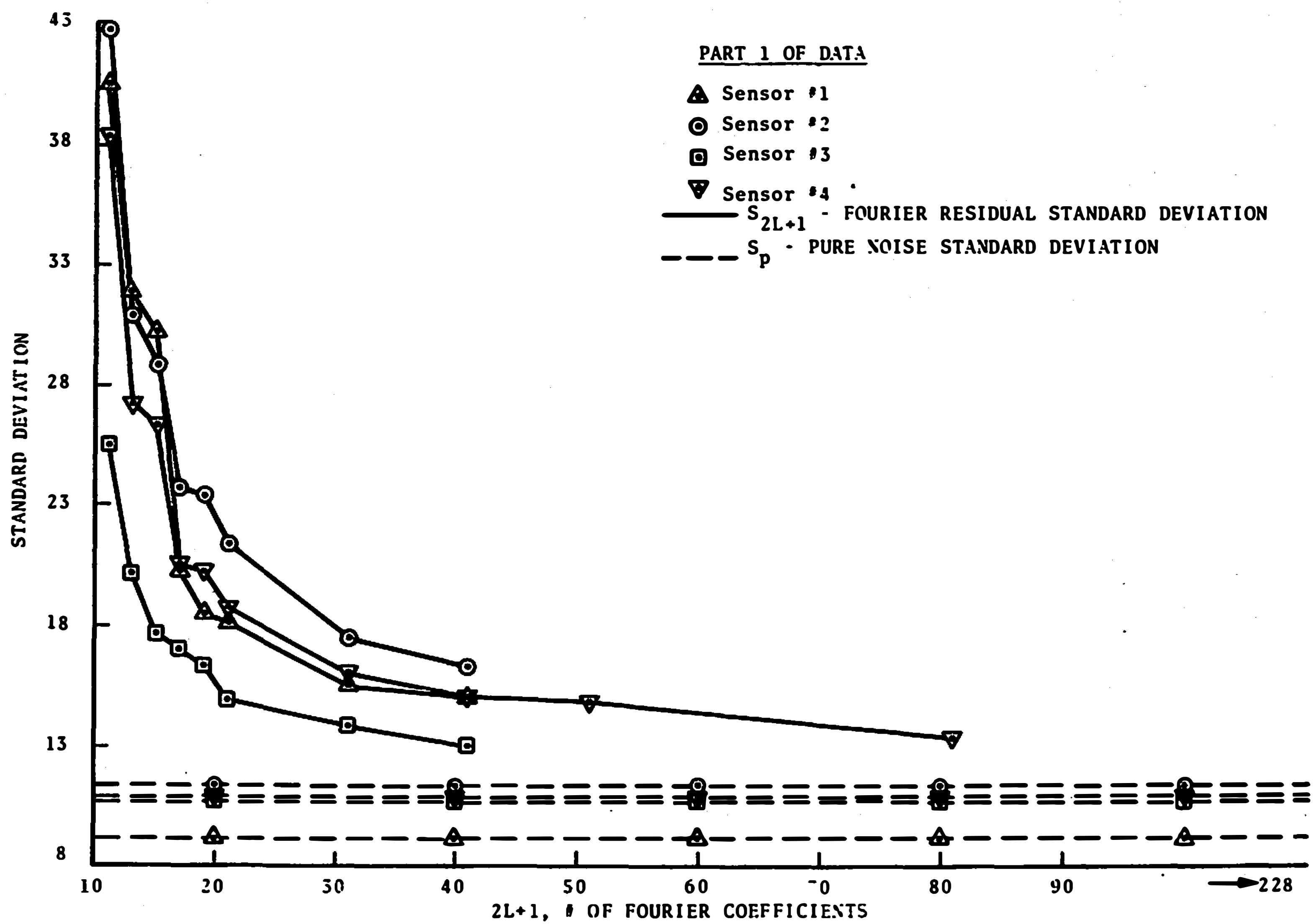

FIGURE F1. FOURIER RESIDUAL STANDARD DEVIATION AND PURE NOISE STANDARD DEVIATION VS. NUMBER OF FOURIER COEFFICIENTS limit to which S_{2L+1} approaches as the number, 2L+1, of Fourier coefficient goes to 288. The calculations were repeated for Part 2 of the data with the number of Fourier coefficients considered being 11 through 21, 31 and 41. These results are shown in Figure F2. By the F Statistic (equation F11), all values of S^2_{2L+1} shown are significantly larger than the corresponding S^2p value except for sensor 3, 41 Fourier coefficient for Part 2. This tends to indicate that large numbers of Fourier coefficients are needed for "good fit." It can be noticed from the Figures F1 and F2 that the rate of change of S_{2L+1} with respect to the number of Fourier coefficients is quite rapid when the number is less than 21 in contrast to when the number of coefficients is greater than 21. This fact is used as the basis for some predictor calculations discussed in Section F.3. It was originally planned to use both parts of the data in the predictors (i.e., Part 2 of the data to predict Part 1 of the data and Part 1 to predict Part 2). As an indication of the expected effectiveness of the various smoothed averages when calculated on one part of the data and used to predict on the other part of the data, two measures are introduced. These measures used the weighted sum of squares of the differences between the smoothed values computed on Part 1 (Part 2) and the actual averages computed on Part 2 (Part 1). The measure, $S_{i,j,2L+1}$ (see equation F12), which is formed from this sum of squares, has as its denominator the same quantity, 288-(2L-1), as does S_{2L+1} and is thus easily compared to S_{2L+1}.

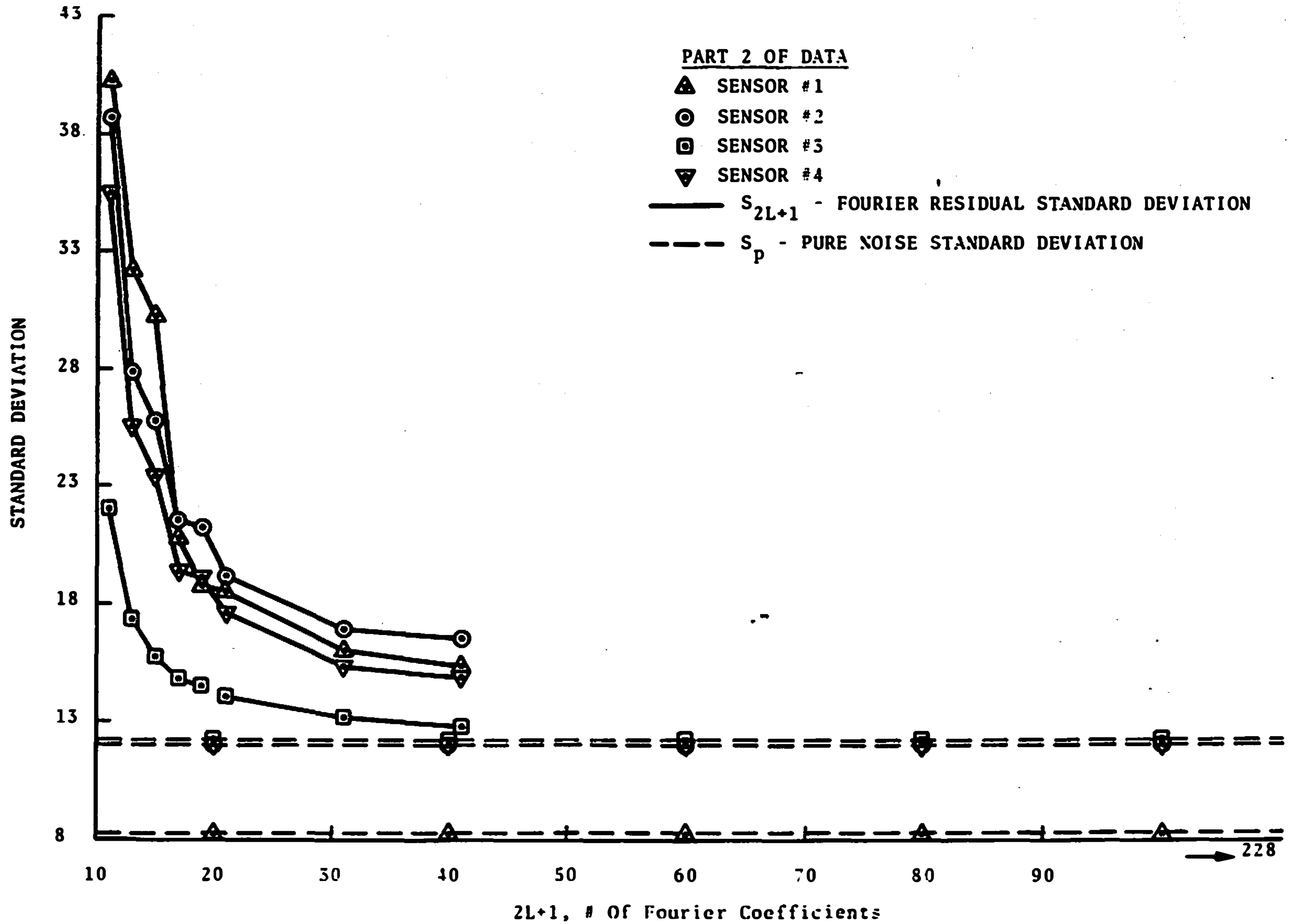

FIGURE F2. FOURIER RESIDUAL STANDARD DEVIATION AND PURE NOISE STANDARD DEVIATION VS. NUMBER OF FOURIER COEFFICIENTS

The other measure $E_{i,j,2L+1}$ (see equation F13), has this sum of squares divided by $\sum_{\ell=1} n_{\ell i}$ and is thus the square root of the weighted mean of the squared differences.

The definitions are as follows:

$$\text{F12)} \quad S_{i,j,2L+1} = \left(\sum_{\ell=1}^{288} n_{\ell i}(\tilde{V}_{\ell j} - \bar{V}_{\ell i})^2/[288 - (2L+1)]\right)^{1/2}$$

$$\text{F13)} \quad \epsilon_{i,j,2L+1} = \sum_{\ell=1}^{288} n_{\ell i}(\tilde{V}_{\ell j} - \bar{V}_{\ell i})^2/[\sum_{\ell=1}^{288} n_{\ell i}])^{1/2}$$

where i = 1, j = 2 and i = 2, j = 1. Here $\tilde{V}_{\ell 1}$ denotes $\tilde{V}_\ell$ calculated on Part 1 of the data, $\tilde{V}_{\ell 2}$ denotes $\tilde{V}_\ell$ calculated on Part 2, $\bar{V}_{\ell 1}$ is $\bar{V}_\ell$ calculated on Part 1, and $\bar{V}_{\ell 1}$ is $\bar{V}_\ell$ is calculated on Part 2.

As would be expected, $S_{1,2,2L+1}$ in Figure F3 and $S_{2,1,2L+1}$ in Figure F4, are higher point by point than the corresponding S_{2L+1} in Figure F1 and S_{2L+1} in Figure F2. Sensor 4 in Figure F4 seems to indicate that a limit for $S_{2,1,2L+1}$ of about 18.5 is reached for $31 \leq 2L+1 \leq 81$ for that particular sensor. Much more information can be perceived from Figures F5 and F6. The four horizontal dashed lines in these two figures are the values obtained when $\bar{V}_\ell$ of Part j is used for $\tilde{V}_{\ell j}$ in the equation for $E_{i,j,2L+1}$ for each sensor. These results are called $E_{i,j,288}$. For all four sensors for both Part 1 and Part 2, the two figures indicate that the average curve $\bar{V}_\ell$ would give better results in the predictor than $\tilde{V}_\ell$ of 21 coefficients which in turn would give better predictor results than the $\tilde{V}_\ell$ for lesser numbers of

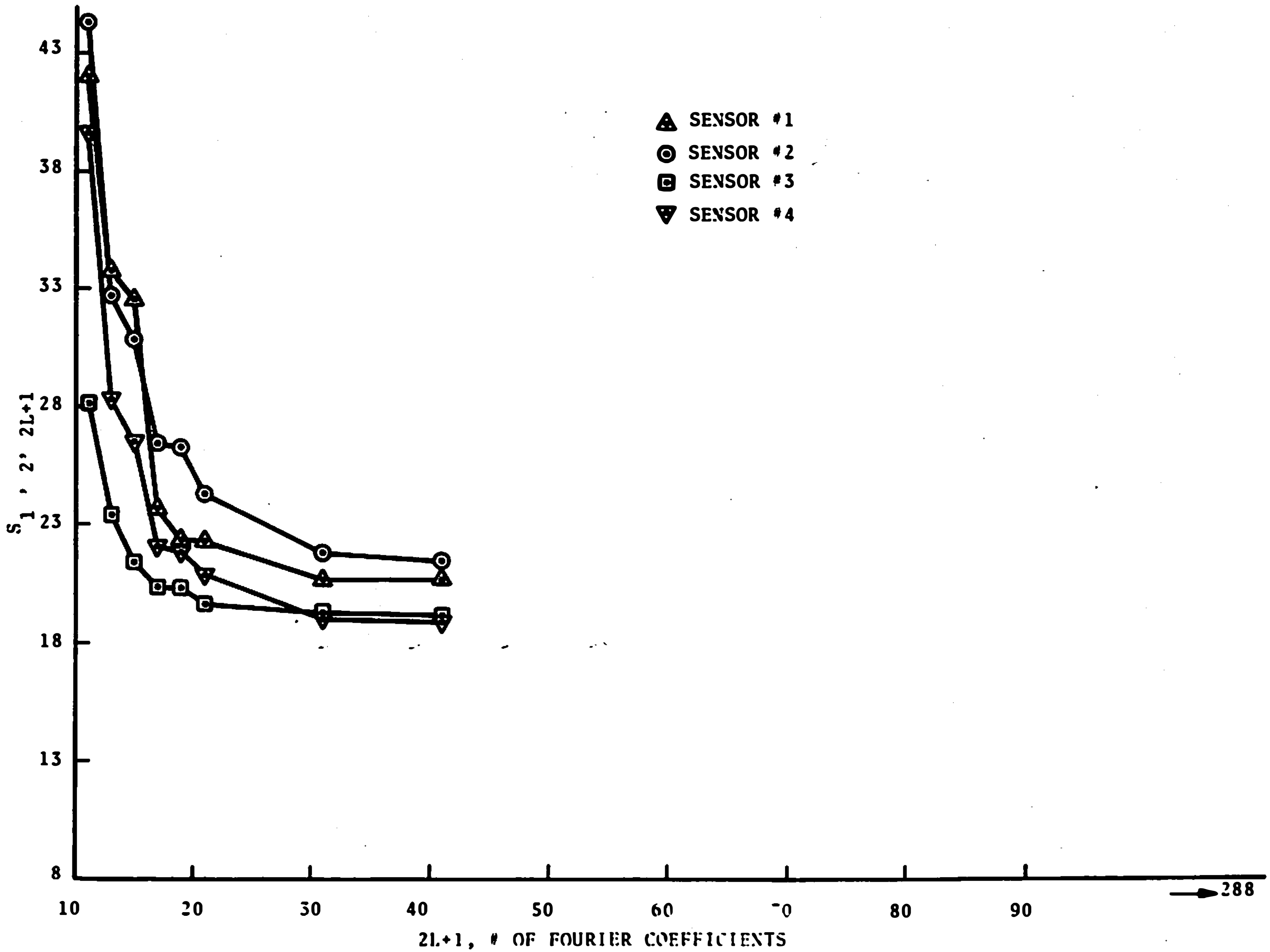

FIGURE F3. $S_{1,2,2L+1}$ VS. NUMBER OF FOURIER COEFFICIENTS

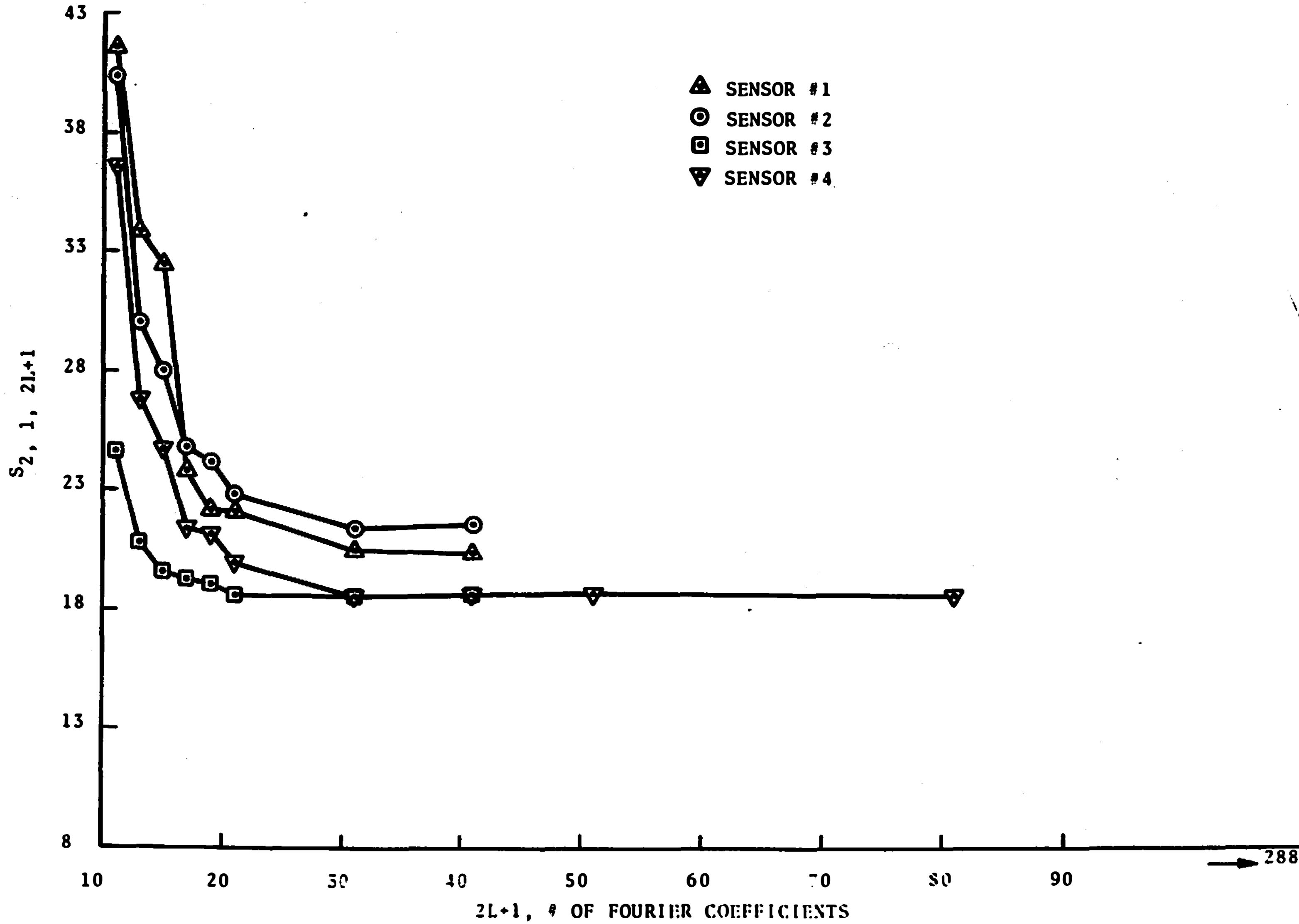

FIGURE F4. $S_{2,1,2L+1}$ VS NUMBER OF FOURIER COEFFICIENTS

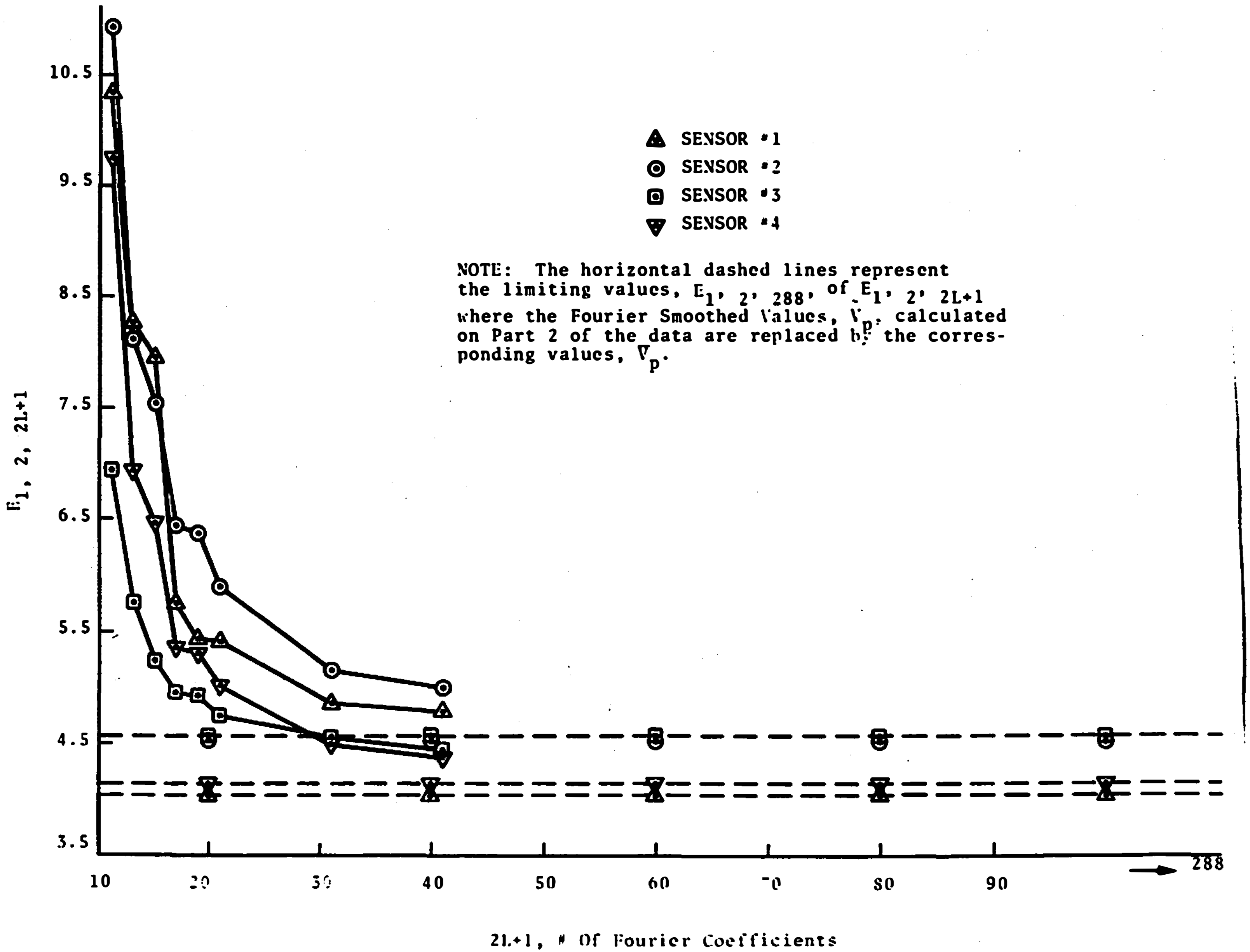

FIGURE F5. ROOT WEIGHTED MEAN SQUARE OF THE DIFFERENCES BETWEEN FOURIER SMOOTHED ESTIMATE OF PART 1 AVERAGES AND PART 2 (STRAIGHT) AVERAGES VS. NUMBERS OF FOURIER COEFFICIENTS

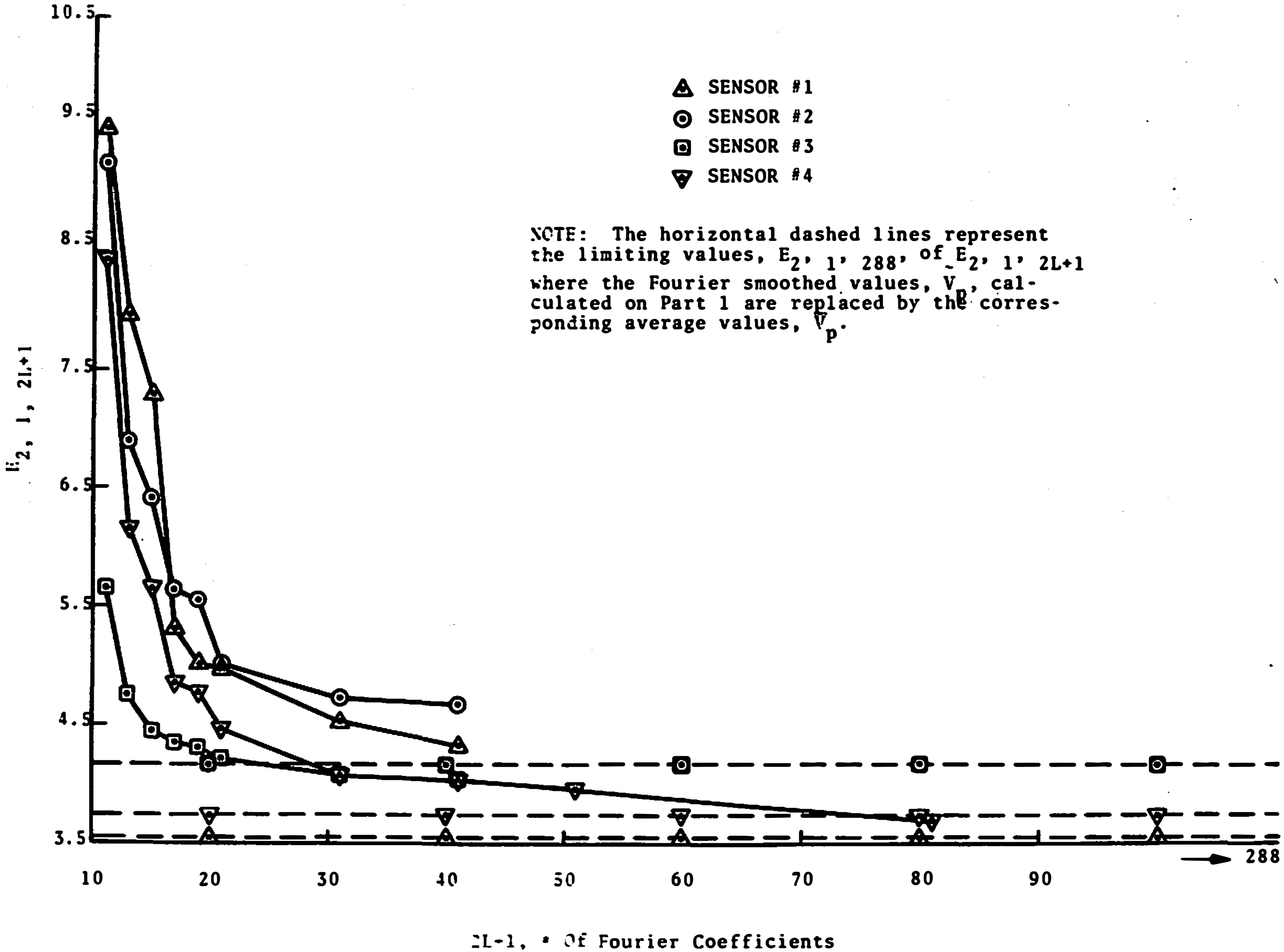

FIGURE F6. ROOT WEIGHTED MEAN SQUARE OF THE DIFFERENCES BETWEEN FOURIER SMOOTHED ESTIMATE OF PART 2 AVERAGES AND PART 1 (STRAIGHT) AVERAGES VS. NUMBER OF FOURIER COEFFICIENTS

coefficients. This is shown to be true in Section F.3 with some calculations.

Figure F5 indicates that a fit, $\tilde{V}_\ell$, of 41 Fourier coefficients is better for prediction than the average curve, $\bar{V}_\ell$, at sensor 3. Similarly from Figure F6, a fit, $\tilde{V}_\ell$, of 31 and 41 Fourier coefficients for sensor 3, and 81 Fourier coefficients for sensor 4 should be better for use with a predictor. This was not verified with actual predictor calculations. But results in Table F2 in section F.3 when compared with Table F1 in this section suggest this is true, although not significantly. The measure in Table F1 shows how less accurate the fits of 2L+1 coefficients are relative to the average curve. It is to be noticed that the entry -2.641% means that it is 2.641% better to use the fit of 41 coefficients of Part 2 than the average curve, $\bar{V}_\ell$, of Part 2 to approximate the average curve of Part 1. (In other words, the error of using 41 Fourier coefficients to determine the average is 2.641% less than the error using the straight historical average, certainly a very small difference from a practical point of view.) It is seen that, except perhaps for very high levels of smoothing, the straight average generally gives more accurate predictions.

Table F1

$$100 \times (E_{1,2,2L+1} - E_{1,2,288})E_{1,2,288}$$

2L+1/Sensor	1	2	3	4
41			-2.641%	
21	34.41%	29.91%	3.773%	21.77%
13	105.82%			

F.3 SOME PREDICTOR RESULTS

All predictor calculations in this paper were predictions on Part 1 of the data. This section discusses some of the results for the Second Generation predictor described by the following equations and discussed in the body of the report.

$$\text{F14)}\quad \bar{Z}_{t-2,2L+1} = \alpha\bar{Z}_{t-3,2L+1} + (1-\alpha)Z_{t-2,2L+1}$$

$$\hat{Z}_{t,2L+1} = \beta\bar{Z}_{t-2,2L+1} + (1-\beta)Z_{t-1,2L+1}$$

where $Z_{t,2L+1} = V_t - \tilde{V}_{t,2L+1}$,

V_t is the volume count at time interval t in Part 2 of the data and $\tilde{V}_{t,2L+1}$ is the volume count at time interval t of the Fourier fit of 2L+1 coefficients for Part 2 of the data. The notation, V_{t-2} means the count at two time intervals before time interval t. The α's and β's were optimized for each sensor and for each particular "Fourier" fit. The criteria for optimizing the α's and the β's is to find their values which give the minimal value, N_{2L+1}, of the root mean square of the error (error = $Z_t - \hat{Z}_t$). The N_{2L+1}'s are smaller for all the sensors when the average curve is used for $\tilde{V}_\ell$ in the predictor than when $\tilde{V}_\ell$ is the fit of 21 Fourier coefficients. This is shown in Table F2. This table shows that a fit of 13 Fourier coefficients is 10.5% less accurate at sensor 1 than when using the (unsmoothed) average curve such as $\tilde{V}_\ell$. In the Table, N_{288} is the value obtained when the average curve, $\bar{V}_\ell$, is used for $\tilde{V}_\ell$.

Table F2

$100 \times (N_{2L+1} - N_{288})/N_{288}$

2L+1/Sensor	1	2	3	4
21	5.4%	3.7%	.25%	1.9%
13	10.5%			

Note that when the common entries in Table F1 and Table F2 are compared, although the magnitudes are different by about a power of 10, Table F1 does indicate what is to be expected in Table F2. For example, at sensor 1, using the average V_ℓ for $\tilde{V}_\ell$ is better than using the fit of 21 Fourier coefficients, etc.

F.4 SUMMARY

In conclusion, it is best to use a relatively high fit of the historical data to obtain the best predictor results. This fit can be chosen by an examination of the data like that given in Figures F5 and F6. If having such a high fit is not feasible (because of high costs of computation, etc.), then it is best to use the straight curve than to use a Fourier fit of small number of coefficients. The aforementioned examination is less expensive and time consuming than to actually use predictors on the data to arrive at the same conclusions.

This study has indicated that the average curve leads to more accurate predictors than do Fourier fits of the average curve using up to 30 or 40 or even more coefficients. Fourier fits using even higher numbers of coefficients may lead to better predictors but are apparently only very slightly better than the average curve.