In presenting the dissertation as a partial fulfillment of the requirements for an advanced degree from the Georgia Institute of Technology, I agree that the Library of the Institute shall make it available for inspection and circulation in accordance with its regulations governing materials of this type. I agree that permission to copy from, or to publish from, this dissertation may be granted by the professor under whose direction it was written, or, in his absence, by the Dean of the Graduate Division when such copying or publication is solely for scholarly purposes and does not involve potential financial gain. It is understood that any copying from, or publication of, this dissertation which involves potential financial gain will not be allowed without written permission.

PSEUDO-RANDOM NUMBER GENERATORS HAVING SPECIFIED PROBABILITY DENSITY FUNCTION AND AUTOCORRELATION

A THESIS

Presented to

The Faculty of the Division of Graduate

Studies and Research

by

David Allen Conner

In Partial Fulfillment

of the Requirements for the Degree

Doctor of Philosophy in the School

of Electrical Engineering

Georgia Institute of Technology

August, 1971

PSEUDO-RANDOM NUMBER GENERATORS HAVING SPECIFIED

PROBABILITY DENSITY FUNCTION AND AUTOCORRELATION

Approved:

Chairman

Date approved by Chairman: August 20, 1971

ACKNOWLEDGMENTS

It is a pleasure to express my sincere appreciation to my thesis advisor, Dr. Joseph L. Hammond, Jr. for his assistance and encouragement in the development of this thesis. I also wish to thank Drs. H. Allen Ecker and James R. Rowland for their service as members of the reading committee and Claudine Taylor for typing this thesis.

A special acknowledgment is given to the Army Research Office--Durham for supporting me on a consulting contract to the Advanced Sensors Laboratory of the U. S. Army Missile Command at Redstone, Alabama, that allowed me to complete a portion of the research discussed herein.

Finally, I wish to convey special thanks to my wife, Jerry, and to two of my sons, Wesley and Jeffrey, for their understanding and encouragement which made this work possible.

TABLE OF CONTENTS

TABLE OF CONTENTS (CONTINUED)

TABLE OF CONTENTS (CONTINUED)

LIST OF TABLES

LIST OF TABLES (CONTINUED)

LIST OF TABLES (CONTINUED)

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS (CONTINUED)

Figure | Page

SUMMARY

The objective of this research is to develop a procedure for generating pseudo-random processes with specified probability density function and autocorrelation for use in digital simulation. As part of the work, typical random sequences are generated and their properties subjected to statistical tests to study the accuracy of the method.

The technique that is developed, referred to as the Predistorted Transformed Gaussian Method, produces the required output sequence from a given input sequence with two operations, namely a linear memory filter and a nonlinear zero-memory filter. The input sequence has a Gaussian distribution function with non-orthogonal values. This technique forms the desired density and autocorrelation from the input sequence using a linear memory filter to introduce the autocorrelation and a zero-memory filter to transform the probability density function.

To generate a sequence having the desired probability density function and autocorrelation, the design is carried out in three steps, namely, (1) design of the nonlinear zero-memory filter, (2) calculation of the autocorrelation required as input to the zero-memory filter, and (3) design of the linear memory filter. The nonlinear zero-memory filter is designed using the cumulative distribution functions of the input Gaussian sequence and the specified output sequence. The autocorrelation of the Gaussian sequence used as input to the nonlinear zero-memory filter is obtained by deriving an input/output autocorrelation relationship

for the zero memory filter using the design equation for the filter and an expression for the joint probability density function for the input sequences. The linear memory filter is designed using a Modified Pakov Method which employs optimization techniques.

This method for generating pseudo-random sequences having a specified probability density function and autocorrelation is applicable to all density functions for which an inverse cumulative distribution function can be obtained explicitly. The accuracy of the design is dependent upon the ability to design the linear memory filter.

The method was tested for ten specific design cases. The cases include specified output probability densities of the Uniform Density, the Random Telegraph Signal Density, the Chi-square Density with one degree of freedom, and the Rayleigh Density. Specified autocorrelations included exponential autocorrelations, triangular autocorrelations, and sin(x)/x autocorrelations. All cases passed the autocorrelation tests at a $\gamma = .95$ level of significance. The best results for the probability density tests occurred at a $\gamma = .06$ level of significance. The least accurate results for which the probability density tests passed occurred at a $\gamma = .95$ level of significance. Eight cases gave results in this region of significance. All ten cases gave results within a $\gamma = .99$ level of significance.

CHAPTER I

INTRODUCTION

Definition of the Problem

Digital computer simulation of a system requires a mathematical algorithm for modeling the system and an algorithm to generate the required system inputs. Many systems of interest have inputs modeled as stationary random processes with both the probability density function and the autocorrelation specified.

The objective of this thesis research is to develop a procedure for generating pseudo-random processes with specified probability density function and autocorrelation for use in digital simulation. As part of the work, typical random processes are generated and their properties subjected to statistical tests to study the accuracy of the method.

Origin and History

In his tutorial paper on pseudo-random number generators, Chambers (1) indicated that significant interest in the generation of random numbers on digital computers began over twenty years ago. This work, which has for the most part been based upon recurrence relations involving integers, deals with a number of interrelated problems.

A basic problem is the construction of pseudo-random number generators which produce sequences of independent variables uniform on the interval [0, 1]. The generation is usually based upon one of two methods: the mixed congruential method or the multiplicative congruential method. The principle involved is to generate each value of the variable in turn by an operation involving the previously generated value. A typical algorithm is given by

$$X_{i+1} = aX_i + c \text{ (moludo m)}. \tag{1.1}$$

where X_{i+1} is the newly generated value, X_i is the previously generated value, and a, c, and m are scalar constants. If $c = 0$, the method is called the multiplicative congruential method. If $c \neq 0$, the method is called the mixed congruential method. The relative advantages and disadvantages of these two methods have been thoroughly explored by Chambers and Hull, et al (1, 2). Tests have been developed to check the sequence of numbers for randomness (1, 3) and independence (1, 5, 6).

A second problem area, discussed by Muller (7), includes several techniques which produce sequences of numbers having independent Gaussian distributions. Of these, three are found in common usage. They are the direct approach, which transforms two sequences of numbers uniform on the interval [0, 1] into two Gaussian number sequences by use of trignometric and logarithmic functions; the inverse approach, which transforms a sequence of numbers uniform on the interval [0, 1] by use of the error function integral; and the central limit approach, which sums 12 or more independent uniform values from a sequence on the interval $[-1, 1]$.

Correlated Gaussian sequences are developed by Levin, Gevy, and Pakov (8, 9, 10, 11). These methods make use of the fact that linear operations on Gaussian processes do not change the nature of the probability density function. Independent sequences of numbers $N[0, 1]$ are weighted and summed as

$$y_i = a_1 x_i + a_2 x_{i-1} + a_3 x_{i-2} + \dots + a_n x_{i-n} + \dots \tag{1.2}$$

to produce the desired correlated Gaussian sequence of numbers.

Marsaglia et al and Bankovi (13, 14, 15, 16) discuss the construction of pseudo-random number generators having an exponential distribution. Sequences of numbers are obtained by performing a discrimination action on values from a distribution uniform on the interval $[0, 1]$. At this time, this method has not been investigated with respect to autocorrelation.

Curtis (17) designs pseudo-random number generators which possess an exponential autocorrelation. A random sequence $\{y_n\}$ is generated such that

$$y_{n+\tau} = y_n e^{-k\tau} + z_\tau \tag{1.3}$$

where $e^{-k\tau}$ is the desired autocorrelation and z_τ is a random variable whose moment generating function, $\Phi_z(s)$, is related to the moment generating function of y, $\Phi_y(s)$, by the relation

$$\Phi_{z_\tau}(s) = \frac{\Phi_y(s)}{\Phi_y(se^{-k\tau})} . \tag{1.4}$$

Two methods have been devised for the generation of pseudo-random sequences of numbers having a specified autocorrelation which is the specific problem treated by this thesis. These methods pertain to (1) the generation of correlated Gaussian sequences and (2) the generation of densities having exponential autocorrelations. As a means of approximating autocorrelations other than exponential and densities other than Gaussian, Gujar and Kavanagh (18) propose that a system for generating correlated Gaussian sequences be modified by attaching a zero-memory device. As shown in Figure 1, the linear memory filter is designed to give the desired autocorrelation at its output. The Gaussian sequence is then transformed by means of the zero-memory device to give an output sequence having the required probability density but with no further attention to the autocorrelation.

Broste (19) in a letter to the editor suggests that the method of Gujar and Kavanagh be modified by predistorting the autocorrelation of the Gaussian sequence so as to control the output autocorrelation as well as the output probability density.

In a separate study Nuttal (20) attempts to achieve a desired autocorrelation using only nonlinear zero-memory networks with an input having specified first and second-order statistics. In general, it is impossible to design the zero-memory

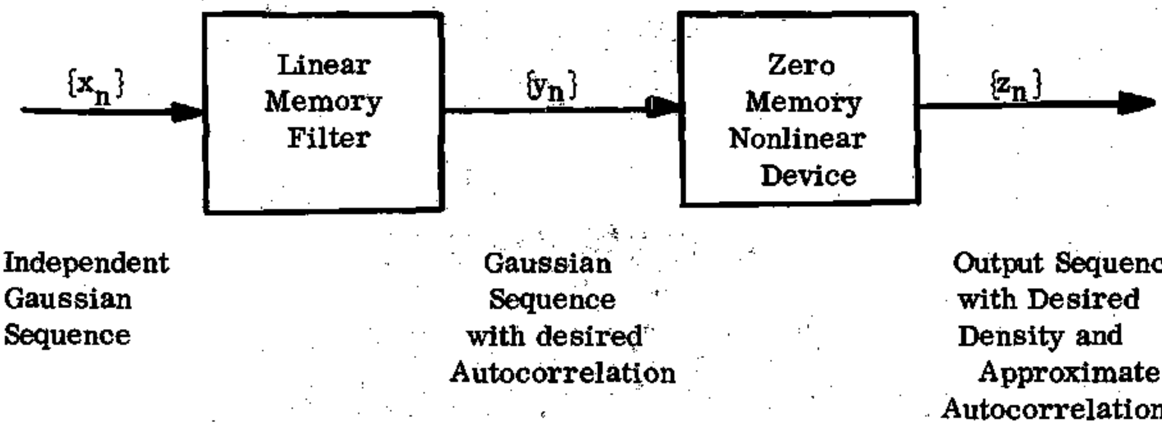

Figure 1. Transformed Gaussian Method.

network to realize the specified output when the input second order statistics is fixed.

Results of the studies by Gujar and Kavanagh, Broste, and Nuttal on the specific thesis problem of generating sequences of pseudo-random numbers with prescribed probability density and autocorrelation can be summarized as follows: sequences of numbers having specified probability density and autocorrelation may be generated from independent Gaussian sequences by alternately controlling the autocorrelation and the probability density characteristics through the use of memory and zero-memory filters in a manner that allows each operation to compensate for the inadequacies of the other operations on the input sequence of numbers.

Applications of Pseudo-Random Sequences with Specified Probability Density Function and Autocorrelation

A significant need for improved techniques in random number generation is in the area of radar system simulation.* A current area of research relates to the development of radar systems to be used in the tracking, identifying, and destroying of enemy aircraft flying at ground level. In simulating angle data, correlated uniform sequences of numbers are needed. A variety of combinations of specified density and autocorrelation functions arise in simulating the radar cross-section data and the ground clutter data. Examples are the Rayleigh density function (21) for the former application and the Weibull density function (22) for the latter application.

Other applications for which simulation studies make use of correlated pseudo-random sequences are:

a) life testing of products where wear-out failure is involved;

b) Monte Carlo analysis of systems where the system components and/or disturbance inputs are random;

c) systems which analyze human responses to new situations based upon previous pattern behavior; and

d) process control such as paper mills, tire factories, or any production line control.

A particularly intriguing application deals with learning systems which must recognize input trends and adapt accordingly.

---

*The author was made aware of this problem during recent employment by the RF Technology Group of the Advanced Sensors Directorate of the U. S. Army Missile Command in Redstone, Alabama.

Overview of Thesis

The research results presented in this thesis deal with the development and testing of a design procedure for digital generation of pseudo-random number sequences with a specified probability density and specified autocorrelation. Chapter II presents the preliminary design considerations. Chapter III presents a detailed discussion of the design considerations required for implementing the Predistorted Transformed Gaussian Method. Chapter IV gives a specific design procedure along with a discussion of the properties and limitations of the procedure.

In order to adequately test the variables generated by the method, consideration must be given to statistical tests and the manner in which they are performed on the random number generator. Such topics are discussed in Chapter V. A summary of the results of the statistical tests as applied to the variables generated in a number of cases studied is given in Chapter VI. Finally, the conclusions derived from the study are presented in Chapter VII.

CHAPTER II

PRELIMINARY DESIGN CONSIDERATIONS

This chapter presents the preliminary approaches considered for the random number generation; formulates the problem mathematically in view of the most favorable approach, the Predistorted Transformed Gaussian Method; and considers the digital simulation of a continuous system when discrete random number sequences are used as inputs.

Preliminary Approaches

This section describes the preliminary design schemes considered for digital generation of pseudo-random numbers with specified probability density and specified autocorrelation. The schemes considered are the Moment Generating Function Method, the Rejection Method, the Transformed Gaussian Method, the Ordering Method, and the Predistorted Transformed Gaussian Method.

Moment Generating Function Method

The moment generating function method, developed by Curtis (17) and discussed in some detail in Chapter I, generates sequences with the aid of a second random variable z_τ which is formed by using the moment generating function of the density being generated. To develop z_τ, the desired density must either have a tractable moment generating function or be Laplace transformable. The latter constraint permits only density functions with positive values to be used. In general, this method is limited to a few continuous density functions, e.g. Gaussian (which can be obtained by other means), exponential, and gamma.

Rejection Method

This method* is similar to the method used for generating exponential distributions discussed in Chapter I. A sampling element is used that possesses a control on the input, as illustrated in Figure 2. The output value resulting from each input value is examined to determine if it is acceptable for the desired output density function; if not, the associated input value is rejected, and a new input value is generated. The objective of the sampler is to prevent the output sequence from having a Gaussian distribution by selecting from the output of the linear memory filter a collection of values having the desired distribution. This technique is unacceptable because (1) the input and output sequences are forced to be nonstationary and (2) the output values near zero are not usually obtainable.

Transformed Gaussian Method

As illustrated in Chapter I in Figure 1, this method (18) produces a Gaussian distributed sequence with the desired autocorrelation and then generates the desired density by passing the sequence through a zero-memory filter. The non-linear zero-memory device degrades the autocorrelation of the Gaussian sequence in proportion to the amount of the nonlinearity. The less similarity the desired density has with a Gaussian distribution, the more corrupted the autocorrelation. As an example, only marginally acceptable results are obtained for the autocorrelation

---

*The Rejection Method was initially proposed as a solution by B. F. Pope of the U. S. Army Missile Command in Redstone, Alabama.

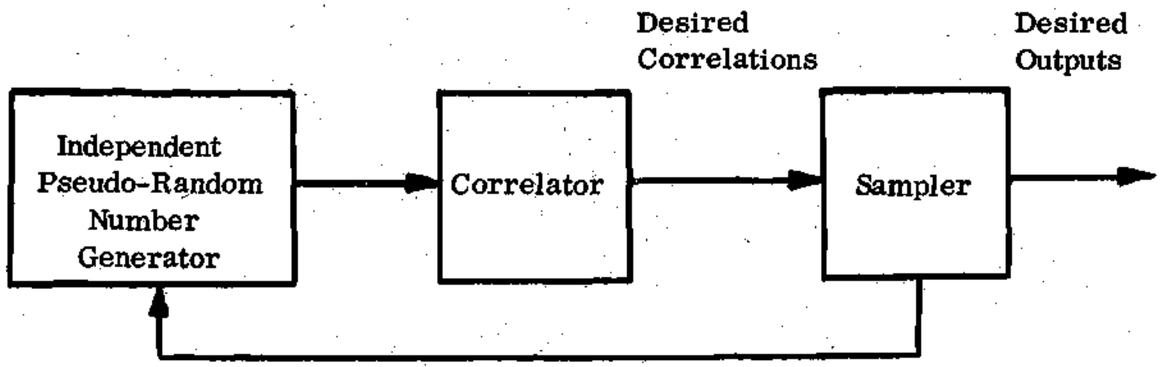

Figure 2. Rejection Method.

of the uniform density. The results are completely unsatisfactory for the auto-correlation of distributions such as the exponential density or the chi-square density having one degree of freedom.

Ordering Method

The ordering method* replaces the nonlinear zero-memory element in the Transformed Gaussian Method by an ordering scheme, as shown in Figure 3. The entire sequence of correlated Gaussian values is first generated. Once the Gaussian sequence has been produced, the values of the sequence are ordered from minimum to maximum. During the ordering process, information concerning the location of the value in the original sequence is retained. A sequence having the same number of values is then generated for the desired distribution, ordered from minimum to

---

*This technique was suggested by F. M. Holliday of the U. S. Army Missile Command and tested jointly with him. The results were later incorporated, in part, in Mr. Holliday's Master's Thesis (23).

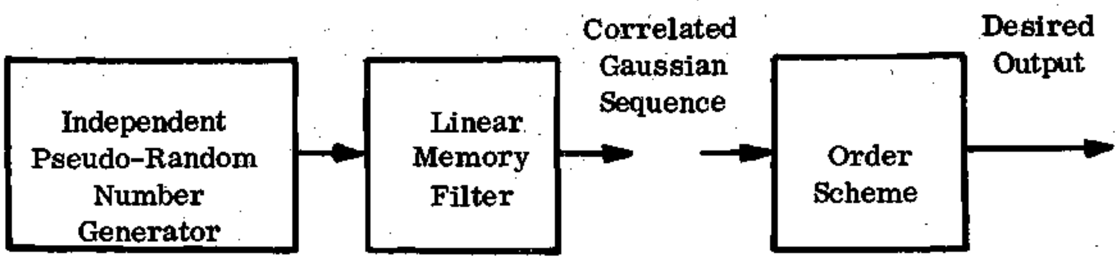

Figure 3. Ordering Method.

maximum. These new values are compared with the ordered Gaussian sequence so that each value from the desired distribution replaces its counterpart in the Gaussian distribution as it appeared in the original sequence.

Evaluation of this method showed that its results were no better, and in fact, essentially the same as for the Transformed Gaussian Method.

Predistorted Transformed Gaussian Method

This method, as presented by Broste* (19), makes use of two effects observed in the Transformed Gaussian Method. First, linear memory filters are used to introduce autocorrelation; but in so doing, the output sequence of the filter is forced to have a Gaussian distribution. Second, zero-memory filters are used to transform probability density functions, with the autocorrelation being corrupted in the process. The interaction of these two effects must be considered if improvement is to be made. The autocorrelation of the output sequence of the linear

*The work by Broste was done concurrently with this research. Since the author served for a time as consultant to the group that Broste worked with, Broste had access to the preliminary results of this research.

memory filter must be distorted to compensate for the corruption introduced by the zero-memory element. Figure 4 illustrates this modification.

The Gaussian input is produced using standard techniques (7). The linear memory filter introduces a portion of the autocorrelation needed to obtain the desired output autocorrelation. This correlated Gaussian sequence is passed through the nonlinear zero-memory device to achieve the specified probability density function and the specified autocorrelation function.

The Predistorted Transformed Gaussian Method is chosen for implementation and refinement since it seems to present the least shortcomings and has the capability of producing the largest class of random sequences having specified probability densities and specified autocorrelations.

Mathematical Formulation

This section presents a mathematical description of the system for implementing the Predistorted Transformed Gaussian Method. The input and output sequences are described, and the method by which the sequences are transformed is characterized. A class of applications for which the procedure may be used is also discussed.

Method of Transformation

Using Figure 4 as a reference, the input to the Predistorted Transformed Gaussian Method is a stationary sequence $\{x_n\}$ of Gaussian random variables. Thus, the joint density is given by (12)

$$p(x_1, x_2, \ldots, x_n) = \frac{1}{(2\pi)^{n/2}\sqrt{|\bar{K}|}} e^{-\frac{1}{2|\bar{K}|}\sum_{i=1}^{n}\sum_{j=1}^{n} \Delta_{ij}x_i x_j} \qquad (2.1)$$

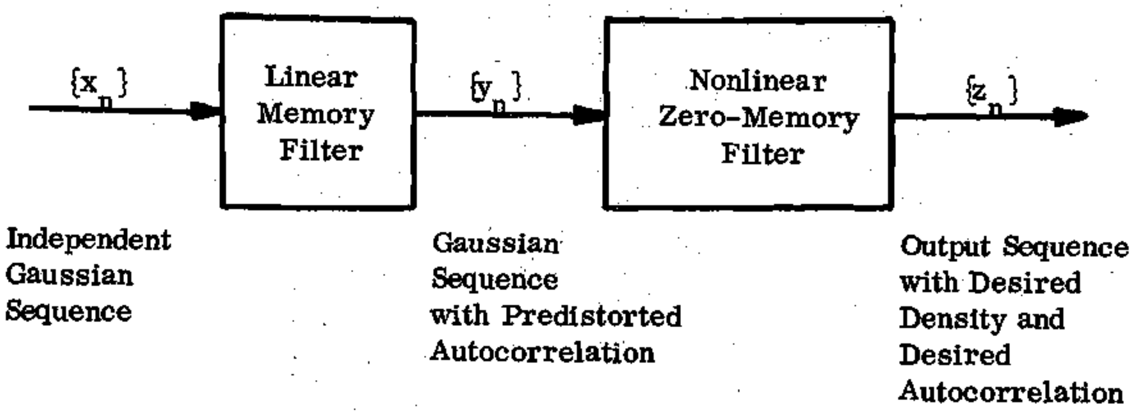

Figure 4. Predistorted Transforded Gaussian Method.

where $|\bar{K}|$ is the determinant of the covariance matrix $\bar{K}$ and Δ_{ij} is the cofactor of the element K_{ij}. Since the process is Gaussian, the first and second-order moments, $E[x_i]$ and $E[x_i x_j]$ completely describe the process (12) and are ideally given by

$$E[x_i] = 0 \quad , \tag{2.2}$$

and

$$E[x_i x_j] = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases} \quad . \tag{2.3}$$

Practically, these parameters can have values other than those given by equations (2.2) and (2.3). If the mean is non-zero, a linear zero-memory operation may be performed on the sequence to adjust the mean to zero. If the second-order moments are non-ideal, the system design can include sufficient compensation to overcome these deficiencies.

The required output is a stationary sequence $\{z_n\}$ with a specified marginal density $p(z_n)$ and a specified autocorrelation $E[z_i z_j]$. Thus, a choice of the linear memory filter and the nonlinear zero-memory filter must be made to produce this result.

The linear memory filter, which transforms the sequence $\{x_n\}$ into the sequence $\{y_n\}$, can be represented by the equation

$$y_j = G_1(x_j, x_{j-1}, x_{j-2}, \ldots, x_{j-k}). \tag{2.4}$$

A special choice of G_1 can be made to relate the correlated sequence $\{y_n\}$ to the input sequence $\{x_n\}$ by

$$y_j = \sum_{i=1}^{k} \alpha_i x_{j-i+1} \tag{2.5}$$

where the α_i's are constants. If the transformation of (2.5) is used, $\{y_n\}$ is a Gaussian random sequence since linear operations on Gaussian processes do not change the nature of the probability density function.

The correlated Gaussian sequence $\{y_n\}$ is next processed through a non-linear zero-memory operation to produce the desired output sequence $\{z_n\}$. Thus the output is given by

$$z_j = G_2[y_j] = G_2[G_1(x_j, x_{j-1}, \ldots, x_{j-k})]. \tag{2.6}$$

For the choice of G_1 given in equation (2.5),

$$z_j = G_2[y_j] = G_2\left[\sum_{i=1}^{k} \alpha_i x_{j-i+1}\right]. \tag{2.7}$$

The autocorrelation of the output sequence $\{z_n\}$ can be given by

$$E\left[z_j z_{j+\tau}\right] = E\left\{G_2\left[\sum_{i=1}^{k} \alpha_i x_{j-i+1}\right] G_2\left[\sum_{i=1}^{k} \alpha_i x_{j-i+1+\tau}\right]\right\} = E\left\{G_2\left(y_j\right) G_2\left(y_{j+\tau}\right)\right\}. \quad (2.8)$$

Application in the Simulation of Continuous Systems

The research discussed in this thesis pertains to the generation of sequences of random numbers which can be used as inputs in simulations. These discrete sequences can be used (in a more or less standard manner) in the simulation of continuous systems. For any given application specific problems arise when selecting the proper number of values for the sequence.

The number of values in the sequence must be chosen to properly describe the significant information in the high frequency region. This choice must be made prudently so that too many samples are not chosen making the data redundant. The selection of the number of samples should be based upon the system cutoff frequency or Nyquist frequency and the length of the time record desired.

If the continuous system being simulated produces output information based upon key random properties of the sequence, the sequence should have a sufficient number of values to allow estimates of these random properties to be made that are within the desired error bounds. The desired error bound, the type of estimator, and the specific random property being estimated will aid in the determination of the length of time record required.

In simulating a continuous white noise process, with an autocorrelation function given by

$$R_c(\tau) = \sigma_c^2\, \delta(\tau), \tag{2.9}$$

the discrete white noise case is of the triangular form given by

$$R_d(\tau) = \begin{cases} \sigma_d^2\left(1 - \frac{|\tau|}{T}\right) & \text{for } |\tau| \leq T \\ 0 & \text{elsewhere} \end{cases} \tag{2.10}$$

Rowland (24) indicates that for a good simulation the equation

$$\sigma_d^2 = \frac{\sigma_c^2}{T} \tag{2.11}$$

will hold provided the higher order frequency effects may be neglected. This will be true if

$$T \leq .1 \frac{1}{f_n} \tag{2.12}$$

where f_n is the highest frequency which the system will pass.

The mathematical formulation of the Predistorted Transformed Gaussian Method for generating random number sequences having specified probability densities and specified autocorrelations will be used as a basis for the design considerations presented in the next two chapters.

CHAPTER III

DETAILED DESIGN CONSIDERATIONS

Based upon the preliminary analysis just completed on the Predistorted Transformed Gaussian Method, design of a system to generate stationary discrete random sequences requires three steps: determination of the nonlinear zero-memory filter, which transforms a correlated Gaussian input sequence into the sequence having the desired output density; determination of the autocorrelation of the correlated Gaussian sequence, given the desired autocorrelation of the output sequence; and determination of the linear memory filter, restricted to

$$y_j = G_1(x_j) = \sum_{i=1}^{k} \alpha_i x_{j-i+1} . \tag{3.1}$$

Each step of the system design is discussed in detail in this chapter.

Design of Nonlinear Zero-Memory Filter

Although the input sequence to the filter may have any mean and variance, the sequence is specified to have a Gaussian density with a zero mean and unity variance. This specification permits the greatest ease in the design of the nonlinear zero-memory filter and can be achieved by restricting the input sequence of the system to have a zero mean and unity variance. The output sequence has a specified probability density and autocorrelation. The nonlinear zero-memory filter can be designed by first transforming the Gaussian sequence $\{y_n\}$ into an intermediate uniform sequence $\{w_n\}$, and then transforming this uniform sequence into the desired output sequence. This operation can be characterized in terms of cumulative distribution functions as illustrated in Figure 5. The random sequence $\{w_n\}$ has values given by

$$w_i = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{y_i} e^{-\frac{y^2}{2}} dy = P_y(y_i). \tag{3.2}$$

The output sequence $\{z_n\}$ may also be related to the uniform sequence by means of a cumulative distribution notation

$$w_i = P_z[z_i]. \tag{3.3}$$

Taking the inverse,

$$z_i = P_z^{-1}[w_i] \tag{3.4}$$

which allows the output sequence to be related directly to the input sequence by

$$z_i = P_z^{-1}[P_y(y_i)]. \tag{3.5}$$

Since $\{y_n\}$ is Gaussian, its cumulative distribution can be written as an error function

$$\mathrm{Erf}(y_i) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{y_i} e^{-\frac{y^2}{2}} dy \tag{3.6}$$

so that

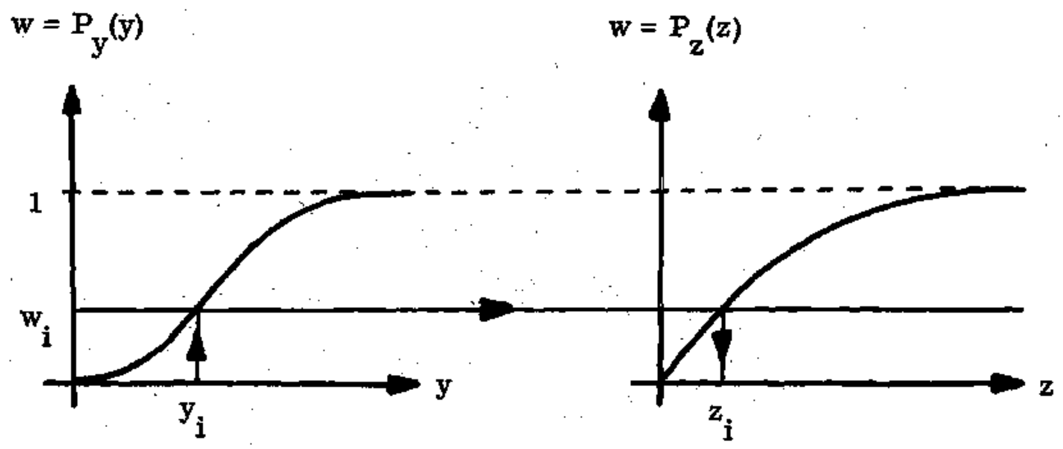

Figure 5. Design of Nonlinear Zero-Memory Filter.

$$z_i = P_z^{-1}\left[\mathrm{Erf}\,(y_i)\right] . \qquad (3.7)$$

Several examples of output densities can be considered to illustrate the performance of this design step. The uniform density requires that $p_z(z_i) = \frac{1}{a}$ for $0 \leq z_i \leq a$ so that

$$z_i = a[\mathrm{Erf}(y_i)] . \qquad (3.8)$$

The random telegraph signal density requires that all positive input values be transformed to an output value of +1 and that all negative input values be transformed to output value of -1. The random telegraph signal distribution is given by

$$p_z(z_i) = \begin{cases} \frac{1}{2} & z_i = 1 \\ \frac{1}{2} & z_i = -1 \end{cases} \qquad (3.9)$$

so that each value of the sequence is given by

$$z_i = P_z^{-1}(w_i) = \begin{cases} -1 & w_i < .5 \\ 0 & w_i = .5 \\ +1 & w_i > .5 \end{cases} \tag{3.10}$$

Using the notation

$$\text{Sgn } u_i = \begin{cases} 1 & u_i > 0 \\ 0 & u_i = 0 \\ -1 & u_i < 0 \end{cases} . \tag{3.11}$$

equation (3.10) becomes

$$z_i = \text{Sgn}\,[\text{Erf}(y_i) - .5], \tag{3.12}$$

or

$$z_i = \text{Sgn}\,(y_i). \tag{3.13}$$

The chi-square density with one degree of freedom has a cumulative distribution of the form

$$P_z(z_i) = \frac{1}{\sqrt{2\pi z}} \int_0^{z_i} e^{-z/2}\, dz . \tag{3.14}$$

The inverse distribution is thus given by

$$z_i = \{\text{Erf}^{-1}[\text{Erf}(y_i)]\}^2 = y_i^{\,2}. \tag{3.15}$$

An exponential density having a cumulative distribution of the form of

$$P_z(z_i) = 1 - e^{-z_i} \qquad z_i \geqq 0 \tag{3.16}$$

requires, for example, a zero-memory element of the form of

$$z_i = -\log[1 - \mathrm{Erf}(y_i)] \, . \tag{3.17}$$

In the formulation of the problem in Chapter II it is pointed out that the nonlinear zero-memory filter might not always exist and can be a possible restriction on the problem. The method for designing the zero-memory filter as shown here is achievable for any specified output marginal density. It is instructive to point out that a table can always be constructed that relates z_i to y_i.

Calculation of Input Autocorrelation to Zero-Memory Filter

The most difficult problem in the system design is the determination of the input correlation function to the zero memory device. A significant amount of work has been performed by Deutsch, Thomas, Thomson, Bonet, and Baum (25-29) in the area of how to determine the relationship between input and output autocorrelation in applications using the assumption that the input sequence has a Gaussian distribution.

Three methods of determining the input/output autocorrelation relationship have been developed. The discussion of these methods which follows makes use of the notation of Figure 6 where the notations $R_y(\tau)$ and $R_z(\tau)$ will represent the collections of second-order statistics $E\{y_j y_{j+\tau}\}$ and $E\{z_j z_{j+\tau}\}$ respectively. Each collection of second-order statistics can also be written in the form

$$R(\tau) = \sigma^2 \rho(\tau) + m^2 \tag{3.18}$$

where $\rho(\tau)$ is the collection of the normalized second-order statistics, σ^2 is the variance, and m is the mean. In each method g(y) is a single-valued transformation of

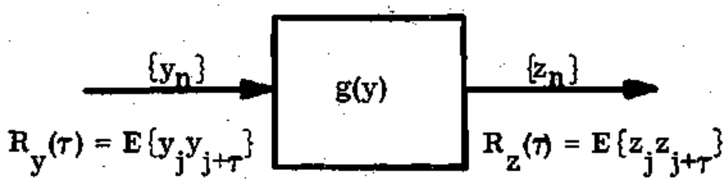

Figure 6. Nonlinear Zero-memory Filter Notation.

$\{y_n\}$ to $\{z_n\}$.

The direct method, sometimes called the density function method, allows the determination of the output autocorrelation in terms of the joint density of the input and the transformation g(y). The result (26) is given by

$$R_z(\tau) = \frac{1}{2\pi\sqrt{1-\rho_y^{\,2}(\tau)}} \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(y_1)g(y_2)$$

$$\exp\left[-\frac{y_1^{\,2}-2\rho_y(\tau)y_1y_2+y_2^{\,2}}{2(1-\rho_y^{\,2}(\tau))}\right] dy_1dy_2 \tag{3.19}$$

where y_1 and y_2 are dummy variables of integration. The only unknown in the right-hand side of equation (3.19) is the input normalized autocorrelation $\rho_y(\tau)$.

Equation (3.19) relates $R_z(\tau)$ as a function of $\rho_y(\tau)$ To obtain $\rho_y(\tau)$ as a function of $R_z(\tau)$ using a direct method, a knowledge of the joint density of the output sequence would be required. As the joint density is not specified, equation (3.19) must be used. In many instances, the relationship of the inverse of the result of equation (3.19) may be found explicitly. As an example, a specified uniform density as output requires an input/output normalized autocorrelation relationship of the form (26)

$$\rho_z(\tau) = \frac{6}{\pi} \arcsin \frac{\rho_y(\tau)}{2} \tag{3.20}$$

which yields

$$\rho_y(\tau) = 2 \sin\left[\frac{\pi}{6} \rho_z(\tau)\right]. \tag{3.21}$$

A specified random telegraph signal density as output requires an input/output normalized autocorrelation relationship of the form (26)

$$\rho_z(\tau) = \frac{2}{\pi} \arcsin[\rho_y(\tau)] \tag{3.22}$$

which yields

$$\rho_y(\tau) = \sin\left[\frac{\pi}{2} \rho_z(\tau)\right]. \tag{3.23}$$

A specified output of a chi-square density with one degree of freedom requires an input/output autocorrelation relationship of the form (26)

$$R_z(\tau) = R_y^{\,2}(0) + 2R_y^{\,2}(\tau) \tag{3.24}$$

which yields

$$R_y(\tau) = \left\{\tfrac{1}{2}\left[R_z(\tau) - R_y^{\,2}(0)\right]\right\}^{\frac{1}{2}}. \tag{3.25}$$

In some cases it is easier to obtain $\rho(\tau)$ by expressing equation (3.19) in terms of a joint characteristic function given by

$$R_z(\tau) = \frac{1}{(2\pi)^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \mathfrak{G}(v_1)\mathfrak{G}(v_2)\Phi(v_1, v_2)dv_1dv_2 \tag{3.26}$$

where

$$\mathfrak{G}(w) = \mathfrak{F}[g(y)] \tag{3.27}$$

and

$$\Phi_y(v_1, v_2) = E\left\{e^{j(v_1y_1+v_2y_2)}\right\}. \tag{3.28}$$

This method is called the characteristic function method.

A case which illustrates this point is the class of non-linear zero-memory filters known as Full-wave Even $v^{\underline{th}}$ Law Devices. This class possesses a zero-memory filter which transforms the Gaussian sequence in the manner of (26)

$$z_i = g(y_i) = c\,|y_i|^v. \tag{3.29}$$

For equation (3.28) $\Phi_y(v_1, v_2)$ can be approximated by a series so that the integral can be evaluated to give

$$R_z(\tau) = 4 \sum_{\text{even}} a_{kv}^2 \frac{\rho_y^k(\tau)}{k!} \tag{3.30}$$

where

$$a_{kv} = \frac{c}{2\pi}\,\Gamma(1+v)\Gamma\left(\frac{k-v}{2}\right)\left[\frac{1}{2}\right]^{\frac{v-k}{2}} \sin\pi\left(\frac{k-v}{2}\right). \tag{3.31}$$

The series form of equation (3.30) prohibits the determination of the inverse relationship. To circumvent this problem, values may be assigned to $\rho_y(\tau)$ and the expression for equation (3.30) tabulated numerically. The result is a table of input second-order statistics with corresponding output second-order statistics. For each specified output second-order statistic, the corresponding input second-order statistic may be determined from the table.

A third method, the series method, allows the determination of the output autocorrelation when the forms of equations (3.19) and (3.26) yield a tractable relationship to integrate. A series expansion (of the Gram-Charlier form) of the input density is developed such that (26)

$$p_y(y_1, y_2) = p_y(y_1)p_y(y_2) \sum_{n=0}^{\infty} A_n \theta_n(y_1)\Psi_n(y_2) \tag{3.32}$$

where

$$A_n = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} p_y(y_1, y_2)\, \theta_n(y_1)\, \Psi_n(y_2) dy_1 dy_2 \tag{3.33}$$

and $\theta_n(y_1)$ and $\Psi_n(y_2)$ are polynomials of order n. Then when $p_y(y_1, y_2)$ is symmetric

$$R_z(\tau) = \sum_{n=0}^{\infty} c_n^{\;2} A_n \tag{3.34}$$

where

$$c_n = \int_{-\infty}^{\infty} g(y)\, p_1(y)\theta_n(y) dy \tag{3.35}$$

where

$$p_1(y) = \frac{1}{\sqrt{2\pi\sigma}} e^{\frac{-y^2}{2\sigma^2}} \tag{3.36}$$

with ρ being the normalized correlation coefficient.

A special case of the series method* makes use of Hermetian polynomials (27). When the input Gaussian signal has a mean of zero and variance of unity,

$$\rho_y(\tau) = R_y(\tau) \ . \tag{3.37}$$

The approximate output is given by

$$R_z(\tau) = \sum_{k=0}^{\infty} c_k{}^2 \rho_y{}^k(\tau) \tag{3.38}$$

where

$$c_k = (k!)^{-\frac{1}{2}} \int_{-\infty}^{\infty} p_y(y)\, H_k(y) \rho_y(y) dy \tag{3.39}$$

where $H_k(y)$ is the Hermite Polynomial of order k given by the recursive relation

$$H_{k+1}(x) = xH_k(x) - \frac{d}{dx} H_k(x). \tag{3.40}$$

The first three polynomials are

$$\left\{ \begin{array}{l} H_0 = 1 \\ H_1 = x \\ H_2 = x^2 - 1 \end{array} \right\} \ . \tag{3.41}$$

---

*The work by Broste (19) makes use of this method.

The use of the series method yields an approximate result the quality of which is dependent on a large number of terms.

Experience in using this design step indicates that the zero-memory filter tends to decorrelate the random sequence. As a result, this design step will usually result in the second-order statistics for the input sequence having a larger amplitude than the corresponding second-order statistics for the output sequence.

Design of Linear Memory Filter

This section presents a detailed discussion of the design of the linear memory filter. Three design procedures are discussed, namely Levin's Method, Gery's Method, and Pakov's Method. In addition, a modification of Pakov's Method is presented. A discussion follows on the implementation of the latter method which is the best design approach.

Design Methods

Levin's Method. This method (9) makes use of a simple recursive formula to generate Gaussian sequences, having either a specified autocorrelation function or power spectrum function, from an independent Gaussian sequence. An inherent advantage of the method is that initial conditions are so chosen that no transient accompanies the starting of the output sequence.

The method develops the recursion formula by making use of certain z-transform concepts. The principle involves the solution of the convolution relationship

$$R_y(\tau) = h(\tau)*h(-\tau) \tag{3.42}$$

which has a z-transform given by

$$\Phi_y(z) = H(z)\,H(z^{-1}) \;. \tag{3.43}$$

H(z) is found by factoring equation (3.43). Since

$$H(z) = \sum_{n=0}^{\infty} h(n)\, z^{-n} \;, \tag{3.44}$$

h(n) may be obtained by long division. The recursive formula for generating the first K-1 output values of the sequence is given by

$$y(n) = \sum_{m=0}^{n} h(m)\, x(n-m) + \xi_n \qquad n \leq K-1 \tag{3.45}$$

where ξ_n is a random variable representing the influence of all x(n) for $n < 0$. All values for $n \geq K$ are given by the recursive relationship

$$\begin{aligned} y(n) = &-b_1 y(n-1) - \ldots - b_k y(n-K) + a_0\, x(n) \\ &+ a_1 x(n-1) + \ldots + a_k x(n-K) \end{aligned} \tag{3.46}$$

where the coefficients are taken from the z-transfer function when written in the form

$$H(z) = \frac{y(z)}{x(z)} = \frac{a_0 + a_1 z^{-1} + \ldots + a_K z^{-K}}{1 + b_1 z^{-1} + \ldots + b_K z^{-K}} \;. \tag{3.47}$$

Gery's Method. This method (10) develops a transformation matrix $\bar{G}$ which transforms independent Gaussian sequences represented by $\bar{X}$ into Gaussian sequences represented by $\bar{Y}$ having the desired autocorrelation. Thus,

$$\bar{Y} = \bar{G}\bar{X} . \tag{3.48}$$

The variance-covariance matrix $\bar{R}$ is given by

$$R = \bar{G}\bar{G}' . \tag{3.49}$$

The required design is the solution of equation (3.49) for $\bar{G}$. It should be noted that the first few values generated are transient in nature and must be discarded.

If the solution of $\bar{G}$ is assumed to be a triangular form $\bar{G}\,(g_{ij})$, the specified covariance matrix $\bar{R}$ is of the form (r_{ij}). Then

$$\left.\begin{aligned}
g_{11} &= \sqrt{r_{11}} \\
g_{ij} &= \frac{r_{1j}}{g_{11}} \\
g_{ii} &= \sqrt{\left(r_{ii} - \sum_{m=1}^{i-1} g_{mi}^{\;2}\right)} \qquad \text{for } i > 1 \\
g_{ij} &= \begin{cases} \left(r_{ij} - \sum_{m=1}^{i-1} \dfrac{g_{mi}g_{mj}}{g_{ii}}\right. & \text{for } j < i \\[2ex] 0 & \text{for } i > j \end{cases}
\end{aligned}\right\} \tag{3.50}$$

so that

$$\left.\begin{aligned} y_1 &= g_{11}x_1 \\ y_2 &= g_{12}x_1 + g_{22}x_2 \\ &\cdot \\ &\cdot \\ &\cdot \\ y_k &= g_{1k}x_1 + g_{2k}x_2 + \ldots + g_{kk}x_k. \end{aligned}\right\} \quad (3.51)$$

Without modification this method is only suited to generating a sequence of finite (and practically very short) length.

Pakov's Method. This method (11) characterizes the design problem in terms of the N equations which result from the expression of equation (3.42). The filter is characterized by N filter weights, $\alpha_1, \alpha_2, \alpha_3, \ldots, \alpha_i, \ldots, \alpha_N$. The relationship of these weights to the normalized autocorrelation is given by the N equations

$$\left.\begin{aligned} f_1 &= \alpha_1^{\;2} + \alpha_2^{\;2} + \ldots + \alpha_N^{\;2} - 1 = 0 \\ f_2 &= \alpha_1\alpha_2 + \alpha_2\alpha_3 + \ldots + \alpha_{N-1}\alpha_N - \rho(\tau) = 0 \\ f_3 &= \alpha_1\alpha_3 + \alpha_2\alpha_4 + \ldots + \alpha_{N-2}\alpha_N - \rho(2\tau) = 0 \\ &\cdot \\ &\cdot \\ &\cdot \\ f_{N-1} &= \alpha_1\alpha_{N-1} + \alpha_2\alpha_N - \rho[(n-2)\tau] = 0 \\ f_N &= \alpha_1\alpha_N - \rho[(N-1)\tau] = 0 \end{aligned}\right\} \quad (3.52)$$

The filter is designed by solving these N equations simultaneously. The desired output values are given in terms of the filter weights as

$$y_j = \sum_{i=1}^{N} \alpha_i x_{j-i+1} \tag{3.53}$$

It should be noted that the first N values of y_j are transient in nature and must be discarded.

Pakov outlines three approaches for solution of the equation. The first, the "simplified method of direct solution," consists of a simple method for calculating filter weights when the desired output sequence has an exponential autocorrelation. This method yields suitable accuracy for engineering purposes. It is obvious that a method of direct solution can also be developed for output sequence having triangular autocorrelations. The limitation with this approach is that the required autocorrelation for the linear memory filter output is rarely ever exponential or triangular. When an exponential or a triangular autocorrelation is specified for the output sequence, the resulting autocorrelations at the output of the linear memory filter is usually a perturbed form of the output autocorrelation. As a result, a "direct method" would give only approximate results. The second approach, the "iteration method," solves the nonlinear equations describing the relationship of the filter weights to the autocorrelation in an iterative fashion until the change in the filter weights is within the desired accuracy. The third approach, the "Newton's Method of successive approximations," reduces the nonlinear system of equations to a corresponding linear system of equations which can be solved by ordinary methods. The implementation of this method is tedious and time consuming.

Modified Pakov Method

A solution is required which will give real values to α_i for each autocorrelation specified. Nakamura (30) gives a necessary, but not sufficient, condition that equation (3.52) has an exact solution for real α_i of

$$1 + 2\sum_{\tau=1}^{N-1} \rho(\tau) \geq 0 \quad . \tag{3.54}$$

An empirical study indicated that additional restrictions exist. Consider, for example, the two weight case for which $\rho(1)$ is specified ($\rho(0)$ is required to be 1). In this case equation (3.52) reduces to

$$f_1 = \alpha_1^2 + \alpha_2^2 - 1 = 0 \tag{3.55}$$

and

$$f_2 = \alpha_1\alpha_2 - \rho(1) = 0 \quad . \tag{3.56}$$

A plot of these equations is given in Figure 7 for $\rho_2 = 0.5$ showing the solution to be $\alpha_1 = \alpha_2 = .707$. Note that for $\rho(1) > .5$ there will be no real solution.*

* Experience has indicated that the triangular autocorrelation serves as a good rule of thumb for indicating whether or not a given autocorrelation will yield a realizable linear memory filter. Let $\rho(\tau)$ represent a given autocorrelation function where $R(\tau) = m^2$ for $\tau \geq T$. There will be a good likelihood that the linear memory filter will not have an exact design if

$$\rho(\tau) > 1 - \frac{|\tau|}{T}$$

for any discrete $\tau < T$. It should be reemphasized that this is only a rule of thumb.

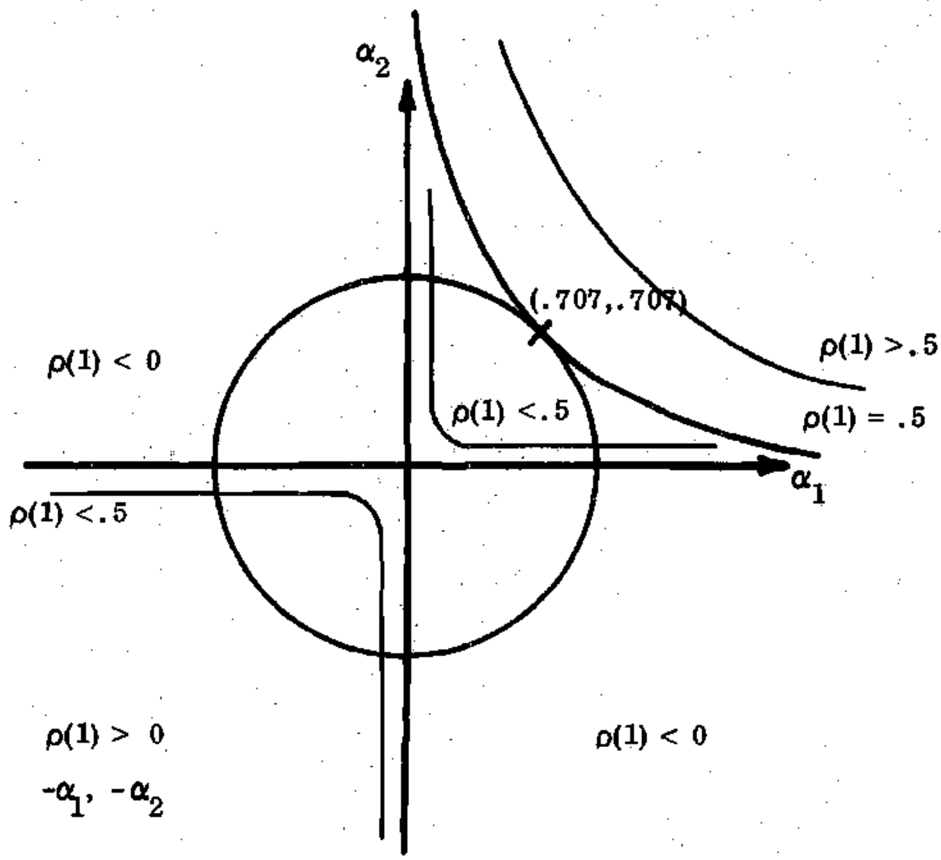

Figure 7. Linear Memory Filter Design Example.

To circumvent the inability to solve the filter equations exactly in certain cases, an optimization approach can be used to select real filter weights which come as close as possible to satisfying the N equations specified in (3.52). This approach can be carried out by minimizing a functional F defined as (31)

$$\sum_{i=1}^{N} f_i^2 = F \tag{3.57}$$

where the f_i have the significance given in equation (3.52). Standard minimization schemes, such as the Fletcher-Powell Technique (31), can be employed to obtain a solution.

Four cases were tested for which the equations had a solution. The optimization approach yielded results that compared favorably with the iteration approach as illustrated in Tables 16 through 19 of Appendix II. It is concluded that the optimization approach may be used in all cases with confidence. The value of the functional minimum for the final solution can be used as an indicator as to whether or not the equations could have been solved by the iterative method. Experience indicates that, in general, for thirty filter weights, a "minimum" $F < 10^{-6}$ corresponds to a set of equations that could be solved by Pakov's Method in an iterative manner or using Newton's Method of successive approximations. A "minimum" $F > 10^{-6}$ corresponds to a set of equations which could not be solved directly. Again, it should be emphasized that the use of the value of 10^{-6} is merely a rule-of-thumb.

An empirical study was performed on the effect that increasing the number of filter weights has on the functional minimum. Four cases were studied that could not be solved using Newton's Method of Successive Approximation. The results are given in Tables 27 and 28 of Appendix II. In all four cases two questions were raised. First, what effect does an increase in the number of filter weights have on the total value of the functional minimum? Second, what effect does an increase in the number of filter weights have on the error introduced by particular equations? The initial minimization for each case used N filter weights. Each minimization effected thereafter increased the number of weights, and hence the number of equations, by N. The functional minimums for each minimization are denoted in the Tables by F. To determine the effect that the number of filter weights had upon the error introduced by particular equations, the error introduced by the original N equations in each case was tabulated for each minimization. This error is denoted in the Tables by A. For each case of the triangular autocorrelation in Table 27, the value of A decreases as the number of filter weights increases until the optimization approach ceases to converge when the number of filter weights is on the order of 30. For the case of the sin (x)/x autocorrelation in Table 28, A gradually increases as the number of filter weights increases. It was concluded that for the triangular case, optimization can be improved by increasing the number of filter weights to the order of 25.

Determination of Filter Weights for Modified Pakov's Method

Pakov's Method uses the assumption that the input sequence is orthogonal, that is

$$E[x_i x_j] = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases} \quad (3.58)$$

In practice, the input sequence is not exactly orthogonal and the modified method is developed without this assumption.

Let

$$y_j = \sum_{i=1}^{N} \hat{\alpha}_i x_{j-i+1} \quad (3.59)$$

represent the output of the linear memory filter. Then

$$E\left[y_j y_{j-\tau}\right] = E\left[\sum_{i=1}^{N} \hat{\alpha}_i x_{j-i+1} \sum_{i=1}^{N} \hat{\alpha}_i x_{j-i+1-\tau}\right]. \quad (3.60)$$

Expanding and collecting terms for the case where $\{x_n\}$ has a zero mean and unity variance yields

$$E\left[y_j y_{j-\tau}\right] = \rho_x(\tau-N+1)\left[\hat{\alpha}_1\hat{\alpha}_N\right] + \rho_x(\tau-N+2)\left[\sum_{i=1}^{2}\hat{\alpha}_i\hat{\alpha}_{i+N-2}\right]$$

$$+ \ldots + \rho_x(\tau-2)\left[\sum_{i=1}^{N-2}\hat{\alpha}_i\hat{\alpha}_{i+2}\right] + \rho_x(\tau-1)\left[\sum_{i=1}^{N-1}\hat{\alpha}_i\hat{\alpha}_{i+1}\right]$$

$$+ \rho_x(\tau)\left[\sum_{i=1}^{N}\hat{\alpha}_i^{\,2}\right] + \rho_x(\tau+1)\left[\sum_{i=1}^{N-1}\hat{\alpha}_i\hat{\alpha}_{i+1}\right]$$

$$+ \rho_x(\tau+2)\left[\sum_{i=1}^{N-2}\hat{\alpha}_i\hat{\alpha}_{i+2}\right] + \ldots + \rho_x(\tau+n-2)\left[\sum_{i=1}^{2}\hat{\alpha}_i\hat{\alpha}_{i+N-2}\right]$$

$$+ \rho_x(\tau+N-1)\left[\hat{\alpha}_1\hat{\alpha}_N\right] \quad . \tag{3.61}$$

It should be noted that the summation terms are the same as those of equation (3.52). It is convenient to introduce the notation

$$\left.\begin{aligned}
a_0 &= \sum_{i=1}^{N} \hat{\alpha}_i^2 \\
a_1 &= \sum_{i=1}^{N} \hat{\alpha}_i \hat{\alpha}_{i+1} \\
a_2 &= \sum_{i=1}^{N-2} \hat{\alpha}_i \hat{\alpha}_{i+2} \\
&\cdot \\
&\cdot \\
&\cdot \\
a_{N-2} &= \sum_{i=1}^{2} \hat{\alpha}_i \hat{\alpha}_{i+N-2} \\
a_{N-1} &= \hat{\alpha}_1 \hat{\alpha}_N
\end{aligned}\right\} \qquad (3.62)$$

Now equation (3.61) can be written in the form

$$\begin{aligned}
E[y_j y_{j-\tau}] = {} & a_0 \rho_x(\tau) + a_1[\rho_x(\tau+1) + \rho_x(\tau-1)] \\
& + a_2[\rho_x(\tau+2) + \rho_x(\tau-2)] + \dots \\
& + a_{N-2}[\rho_x(\tau+N-2) + \rho_x(\tau-N+2)] \qquad (3.63) \\
& + a_{N-1}[\rho_x(\tau+N-1) + \rho_x(\tau-N+1)].
\end{aligned}$$

Equation (3.63) can be expressed compactly in matrix notation where $\bar{P}$ is an N valued column matrix given by

$$\bar{P} = \begin{bmatrix} E[y_j^{\,2}] \\ E[y_j y_{j+1}] \\ E[y_j y_{j+2}] \\ \cdot \\ \cdot \\ \cdot \\ E[y_j y_{j+N-1}] \end{bmatrix}, \tag{3.64}$$

$\bar{R}$ is a 3N-3 valued column matrix given by

$$\bar{R} = \begin{bmatrix} E\left[x_j x_{j-N+1}\right] \\ E\left[x_j x_{j-N+2}\right] \\ \vdots \\ E\left[x_j x_{j-2}\right] \\ E\left[x_j x_{j-1}\right] \\ E\left[x_j^{\,2}\right] \\ E\left[x_j x_{j+1}\right] \\ E\left[x_j x_{j+2}\right] \\ \vdots \\ E\left[x_j x_{j+N-2}\right] \\ E\left[x_j x_{j+N-1}\right] \\ \vdots \\ E\left[x_j x_{j+2N-3}\right] \\ E\left[x_j x_{j+2N-2}\right] \end{bmatrix} \tag{3.65}$$

and $\bar{Q}$ is a (N) x (3N-3) matrix given by

$$
\bar{Q} = \begin{bmatrix}
a_{N-1} & a_{N-2} & a_{N-3} \cdots & a_2 & a_1 & a_0 & a_1 & a_2 & a_3 & \cdots a_{N-2} & a_{N-1} & 0 & \cdots 0 & 0 \\
0 & a_{N-1} & a_{N-2} \cdots & a_3 & a_2 & a_1 & a_0 & a_1 & a_2 & \cdots a_{N-3} & a_{N-2} & a_{N-1} \cdots & 0 & 0 \\
0 & 0 & a_{N-1} \cdots & a_4 & a_3 & a_2 & a_2 & a_0 & a_1 & \cdots a_{N-4} & a_{N-3} & a_{N-2} \cdots & 0 & 0 \\
 & & \cdot & & & & & & & & & & & \\
 & & \cdot & & & & & & & & & & & \\
 & & \cdot & & & & & & & & & & & \\
0 & 0 & 0 & \cdots 0 & a_{N-1} & a_{N-2} & a_{N-3} & a_{N-4} & a_{N-5} \cdots & a_0 & a_1 & a_2 & \cdots a_{N-1} & 0 \\
0 & 0 & 0 & \cdots 0 & 0 & a_{N-1} & a_{N-2} & a_{N-3} & a_{N-4} \cdots & a_1 & a_0 & a_1 & \cdots a_{N-2} & a_{N-1}
\end{bmatrix} \tag{3.66}
$$

Equation (3.54) becomes

$$\bar{P} = \bar{Q}\bar{R} \tag{3.67}$$

Solution of equation (3.67) for a_i allows the modified Pakov Method to be expressed as

$$\left.\begin{aligned}
f_1 &= \hat{\alpha}_1^2 + \hat{\alpha}_2^2 + \ldots + \hat{\alpha}_N^2 - a_0 = 0 \\
f_2 &= \hat{\alpha}_1\hat{\alpha}_2 + \hat{\alpha}_2\hat{\alpha}_3 + \ldots + \hat{\alpha}_{N-1}\hat{\alpha}_N - a_1 = 0 \\
&\;\;\vdots \\
f_{N-1} &= \hat{\alpha}_1\hat{\alpha}_{N-1} + \hat{\alpha}_2\hat{\alpha}_N - a_{N-2} = 0 \\
f_N &= \hat{\alpha}_1\hat{\alpha}_N - a_{N-1} = 0
\end{aligned}\right\} \tag{3.68}$$

This chapter has discussed in detail the design considerations for the Predistorted Transformed Gaussian Method for the generation of stationary discrete random sequences. The next chapter will survey the concepts presented in this chapter by giving a specific approach to follow in making the design along with a means of assessing the error of the method for generating the desired output sequence.

CHAPTER IV

FINAL DESIGN PROCEDURE

This chapter presents the final design procedure along with an assessment of the error of the method for generating the desired output sequence.

Design Method

The final design procedure consists of the following steps.

Step 1. Determination of the nonlinear zero-memory filter: The zero-memory filter is designed by using the relationship

$$z_i = P_z^{-1}\left[\mathrm{Erf}(y_i)\right] \tag{4.1}$$

where P_z^{-1} is the inverse of the cumulative distribution function, $P_z(z)$ is specified for the output sequence. This design can be achieved for all density functions for which an inverse function P_z^{-1} exists.

Step 2. Determination of the autocorrelation of the Gaussian sequence $\{y_n\}$: Determination of the input second-order statistics to the zero-memory filter can be illustrated in principle as follows. The general input-output autocorrelation relationship for the zero-memory filter is derived using the Direct Method given by

$$R_z(\tau) = \frac{1}{2\pi\sqrt{1-\rho_y^2(\tau)}}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(y_1)g(y_2)\exp\left[-\frac{y_1^2-2\rho_y(\tau)y_1y_2+y_2^2}{2(1-\rho_y^2(\tau))}\right]dy_1dy_2 \ . \tag{4.2}$$

Since this equation is in the form

$$R_z(\tau) = \mathcal{H}\,[\rho_y(\tau)], \tag{4.3}$$

the inverse can be taken in principle to yield

$$\rho_y(\tau) = \mathcal{H}^{-1}\,[R_z(\tau)]. \tag{4.4}$$

To implement this approach in practice, three alternative methods can be used, namely analytic implementation, characteristic function implementation, or brute force implementation using series or a tabular method. The direct method for calculating the input/output autocorrelation can prove to be an untractable mathematical step. In such cases, the Characteristic Function Method described in Chapter III can be substituted. If this approach also proves to be untractable mathematically, the Series Method can be used using Hermetian polynomials or some other series of the Gram-Charlier Form. The series approach is used at the sacrifice of having to calculate a large number of terms to gain satisfactory accuracy. When input/output autocorrelations, of the form of equation (4.3), result that do not have an inverse form that can be readily found, a table of input/output autocorrelation values can be developed. Each input second-order statistic can be found by entering the table for each output second-order statistic and reading off the corresponding input second-order statistic.

This design step can be achieved for all zero-memory devices and all output autocorrelations. For example, the brute force approach would seem to always be possible. Deutsch, Thomas, and Baum (25, 26, 29) have catalogued a large group of zero-memory devices for which the input/output autocorrelations have been determined.

Step 3. Determination of the linear memory filter: The determination of the linear memory filter requires seven steps.

1. Determine the number of filter coefficients $\hat{\alpha}_i$ to be used in the design. Call this number N.

2. Calculate the normalized second-order statistics of the input sequence. The second-order statistics from $\tau = 1 - N$ to $\tau = 2N - 2$ are used to form a matrix $\bar{R}$ given by

$$\bar{R}^T = [\rho_x(1-N) \quad \rho_x(2-N) \quad \ldots \quad \rho_x(0) \quad \ldots \quad \rho_x(2N-2)]. \tag{4.5}$$

3. Form a matrix $\bar{P}$ representing the desired output second-order statistics given by

$$\bar{P}^T = [\rho_y(0) \quad \rho_y(1) \quad \cdots \quad \rho_y(N-1)]. \tag{4.6}$$

4. Form a matrix $\bar{Q}$ which will be composed of the correlation values for which the linear memory filter will be designed given by

$$\bar{Q} = \begin{bmatrix} a_{N-1} & a_{N-2} \cdots a_0 & a_1 & a_2 & \cdots a_{N-1} & 0 & \cdots 0 & 0 \\ 0 & a_{N-1} \cdots a_1 & a_0 & a_1 & \cdots a_{N-2} & a_{N-1} & \cdots 0 & 0 \\ 0 & 0 \quad \cdots a_2 & a_1 & a_0 & \cdots a_{N-3} & a_{N-2} & \cdots 0 & 0 \\ & \vdots & & & & & & \\ 0 & 0 \quad \cdots a_{N-2} & a_{N-3} & a_{N-4} \cdots a_1 & a_2 & \cdots a_{N-1} & 0 \\ 0 & 0 \quad \cdots a_{N-1} & a_{N-2} & a_{N-3} \cdots a_0 & a_1 & \cdots a_{N-2} & a_{N-1} \end{bmatrix}. \tag{4.7}$$

5. The resulting relationship between matrices $\bar{P}$, $\bar{Q}$, and $\bar{R}$ is given by

$$\bar{P} = \bar{Q}\bar{R} \tag{4.8}$$

This equation relates a set of N equations having N unknown a_i's. Solve this set of equations for the a_i's.

6. Form a set of filter design equations given by

$$\left.\begin{aligned} f_1 &= \hat{\alpha}_1^2 + \hat{\alpha}_2^2 + \ldots + \hat{\alpha}_N^2 - a_0 = 0 \\ f_2 &= \hat{\alpha}_1\hat{\alpha}_2 + \hat{\alpha}_2\hat{\alpha}_3 + \ldots + \hat{\alpha}_{N-1}\hat{\alpha}_N - a_1 = 0 \\ &\vdots \\ f_{N-1} &= \hat{\alpha}_1\hat{\alpha}_{N-1} + \hat{\alpha}_2\hat{\alpha}_N - a_{N-2} = 0 \\ f_N &= \hat{\alpha}_1\hat{\alpha}_N - a_{N-1} = 0 \end{aligned}\right\} \quad . \tag{4.9}$$

7. These equations can now be combined to give a composite functional

$$F = \sum_{i=1}^{N} f_i^2$$

which can be optimized to yield the values of $\hat{\alpha}_i$ to be used as the filter coefficients.

This design procedure will yield a design for the linear memory filter for all specified output autocorrelations. For some specified output autocorrelations the filter design will be approximate as the solutions of the equations of (4.9) obtained by minimization do not intersect for the resulting values of $\hat{\alpha}_i$.

Assessment of Error for Design Method

The accuracy of the design method is dependent upon the ability to minimize the functional F in the design of the linear memory filter. An assessment of the error of the design method can be based upon the value of F with the design procedure being considered to be exact for functional minimums less than 10^{-6}. As the functional minimum increases above 10^{-6}, the design method has a degree of error introduced by the linear memory filter design. Satisfactory design results can be obtained up to a value of functional minimum on the order of 10^{-1}. Designs having values of the functional minimum above 10^{-1} should be judged on an individual basis as to the accuracy of the output autocorrelation.

An additional error may be introduced if a proper choice of input sequence is not made. Some sequences require a large number of values in a localized region of the probability density curve. Such sequences require a prudent choice of the input sequence to insure that the sequence is distributed properly to give the specified output sequence.

This chapter has presented a detailed outline of the design procedure along with an error assessment of the design method. The next chapter will present the statistical tests used to evaluate the performance of this design method when used for several interesting cases.

CHAPTER V

STATISTICAL TESTS FOR EVALUATING PERFORMANCE

The objective of the research discussed in this thesis is to develop a technique for generating stationary discrete random sequences. The design technique, discussed in Chapters III and IV, has been implemented for several interesting cases. This chapter presents a discussion of the statistical tests used to evaluate the performance of the designs for these cases by first identifying the critical statistics to be tested and then discussing the tests to be used.

Classification of Critical Statistics

The Predistorted Transformed Gaussian Method for generating stationary discrete random sequences uses an input Gaussian sequence $\{x_n\}$ to produce an intermediate correlated Gaussian sequence $\{y_n\}$ and the desired output sequence $\{z_n\}$. Three properties describe the sequences (32). The first property, the mean square property, is characterized by two statistics, the mean which gives a static description of the sequence and the variance which gives a dynamic description of the sequence. The second property, the probability density function property, relates the probability that a value of the sequence will be within some defined range. The third property, autocorrelation, describes the dependence of each value in the sequence on all other values in the sequence. In addition, the characteristic of stationarity is used to characterize the sequence. The characteristic of a sequence being stationary implies that the random properties of the sequence do not vary significantly due to translation in position within the sequence.

The random properties of each sequence are calculated using the values of the sequence. The mean square statistics are calculated by averaging N values of the sequence such that the mean of the sequence $\{u_n\}$ is given by

$$\bar{u} = \langle u_n \rangle_N, \tag{5.1}$$

and the variance is given by

$$s^2 = \langle (u_n - \bar{u})^2 \rangle_N . \tag{5.2}$$

The probability density property is determined by calculating either the density function or the cumulative distribution function for N values of the sequence. The autocorrelation property is determined by using $N+\tau$ values of the sequence to calculate the second-order statis tics given by

$$E[u_j u_{j+\tau}] = \frac{1}{N} \sum_{j=1}^{N} u_j u_{j+\tau} . \tag{5.3}$$

Standard statistical tests can be used to study some of the properties of the sequences produced by the Predistorted Transformed Gaussian Method. The mean of a Gaussian sequence can be tested using the Student-t Test for Means (32). The variance of a Gaussian sequence can be tested using the Chi-square test for Variances (32). The probability density property can be tested for most sequences using either the Pearson Chi-square Goodness-of-Fit Test (32, 33) or the Kolmogorov

Test (33, 34). The chi-square goodness-of-fit test compares the density function of the generated sequence with the required density function. The Kolomogorov Test compares the cumulative distribution function of the generated sequence with the required cumulative distribution function.

Two newly developed statistical tests can be used to study the autocorrelation property. The first test, developed by Patel (35), tests the second-order statistics without using information concerning the variance. The second test, presented for the first time in this thesis, tests the second-order statistics of Gaussian sequences. This latter test can be extended to test the autocorrelation property of the output sequence for the design method being tested.

Statistical Tests

Because the calculated statistics for each sequence are determined using a finite number of values from the sequence, the calculated statistics will vary from the theoretical values by a small amount. Each statistical test cited in the previous section compares calculated statistics with the theoretical values by developing a random variable which describes the degree of agreement or disagreement. A rational basis is used to determine from this random variable whether or not the sequence being tested possesses the theoretical value required.

The decision as to whether or not the test sequence possesses the specified theoretical statistic is based upon a consideration of the probability density function of the random variable describing the degree of agreement or disagreement. Figure 8 is used to illustrate the mechanics of this decision. Φ is the calculated

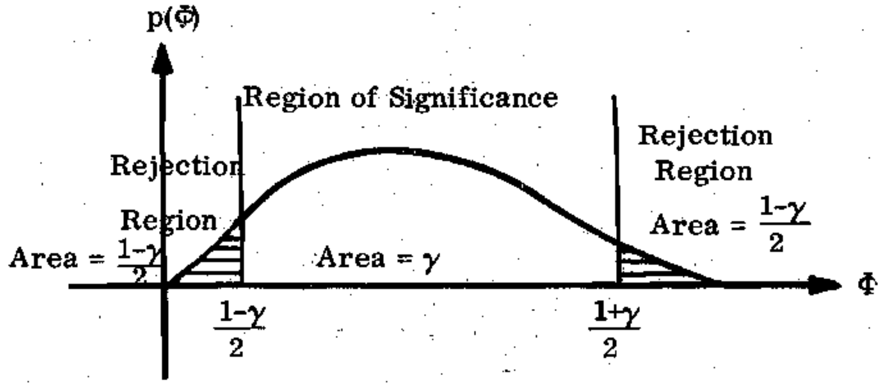

Figure 8. Probability Density of Random Variable Comparing Calculated Statistic and Theoretical Statistic.

random variable with a probability density of $p(\Phi)$. A region of significance (called a level of significance, γ) is chosen for the density curve. When the sequence possesses the specified theoretical statistic, the random variable Φ will be within the region of significance with probability of γ and outside of the region with probability of $1 - \gamma$. By choosing γ large, Φ will fall outside of the region of significance only a small portion of the time. When Φ is outside the region of significance, an assumption can be made that the sequence being tested does not possess the specified theoretical statistic. The smaller that γ is made, the more stringent the test becomes. Typical values for γ are .80 (or 80%), .90 (or 90%), and .95 (or 95%).

The tests then possess an inherent error. When the tested sequence possesses the specified theoretical statistic, the test will fail $(1 - \gamma)\%$ of the time. When the tested sequence does not possess the specified theoretical statistic, the test will pass only a small portion of the time. On the whole, however, the tests give a useful measure of the statistics being tested.

Mean Square Value Tests

The tests of this section apply to sequences $\{u_n\}$ that are normal with mean m_u and variance σ^2.

Mean. Consider a random sequence having N values. The mean $\bar{u}$ is given by (32)

$$\bar{u} = \frac{1}{N} \sum_{i=1}^{N} u_i \tag{5.4}$$

where u_i is the i^{th} value of the sequence. The mean of the sequence has a mean of

$$E[\bar{u}] = m_u , \tag{5.5}$$

and a variance of

$$\mathrm{Var}[\bar{u}] = \frac{\sigma^2}{N} . \tag{5.6}$$

A statistic showing the degree of disagreement between the calculated mean and the theoretical mean is given by the relationship

$$t_n = \frac{(\bar{u} - m_u)\sqrt{N}}{s} \tag{5.7}$$

where s is the calculated standard deviation given by

$$s = \sqrt{\frac{1}{N-1} \sum_{i=1}^{N} (u_i - \bar{u})^2} \tag{5.8}$$

The statistic t_n has a Student-t distribution with $n = N-1$ degrees of freedom. (The loss of one degree of freedom results from the lack of knowledge of the true standard deviation.) The theoretical mean can be tested by comparing the t_n statistic with a Student-t distribution having N-1 degrees of freedom where (32)

$$\text{Prob}\left[t_{n;\frac{1-\gamma}{2}} < \frac{(\bar{u}-m_u)\sqrt{N}}{s} \leq t_{n;\frac{1+\gamma}{2}}\right] = \gamma. \tag{5.9}$$

If t_n falls outside of these limits, it can be concluded that the true mean of the sequence is not m_u.

The variance of the mean $\bar{u}$ is a measure of the precision of the calculation of $\bar{u}$. Hansen, et al (36) suggests that the variation of the calculation not exceed 4%, or

$$\text{Var}[\bar{u}] = \frac{\sigma^2}{N} \leq .04. \tag{5.10}$$

Then, for a N(0, 1) process, $N > 25$.

Variance. Consider a random sequence having N values. The variance s^2 is given by (32)

$$s^2 = \frac{1}{N-1} \sum_{i=1}^{N} (u_i - \bar{u})^2 \tag{5.11}$$

where u_i is the $i^{\underline{th}}$ value of the sequence and $\bar{u}$ is the mean given by equation (5.4). The variance of the sequence has a mean of

$$E[s^2] = \sigma^2, \tag{5.12}$$

and variance of

$$\mathrm{Var}[s^2] = \frac{1}{N-1}\left\{E\left(u_n^4\right) - E^2\left(u_n^2\right) + 4\left[E\left(u_n^2\right)E^2(u_n) - E(u_n)E\left(u_n^3\right)\right]\right\}. \tag{5.13}$$

For a $N(0, \sigma^2)$ process $E(u_n) = E(u_n^3) = 0$, $E(u_n^2) = \sigma^2$, and $E(u_n^4) = 3\sigma^4$.

Thus,

$$\mathrm{Var}[s^2] = \frac{2}{N-1}\,\sigma^4. \tag{5.14}$$

The variance of the variance s^2 is a measure of the precision of the calculation of s^2. Using the suggested 4% maximum variation,

$$\mathrm{Var}[s^2] = \frac{2}{N-1}\,\sigma^4 \leq .04\,. \tag{5.15}$$

For a $N(0,1)$ process, $N \geq 51$.

A statistic indicating the degree of disagreement between the calculated variance and the theoretical variance is given by

$$\chi^2 = \frac{(N-1)s^2}{\sigma^2} \tag{5.16}$$

where the statistic χ^2 has a chi-square distribution with $n=N-1$ degrees of freedom. (The loss of one degree of freedom results from the lack of knowledge of the true mean.) The theoretical variance can be tested by comparing the χ^2 statistic with a chi-square distribution having $N-1$ degrees of freedom where (32)

$$\mathrm{Prob}\left[\chi^2_{n;\frac{1-\gamma}{2}} < \frac{(N-1)}{\sigma^2}{}^{2} \leq \chi^2_{n;\frac{1-\gamma}{2}}\right] = \gamma\,. \tag{5.17}$$

If χ^2 falls outside of these limits, it can be concluded that the true variance is not σ^2.

Probability Density Tests

Pearson Chi-square Goodness-of-Fit Test. This test compares the theoretical probability density function with the probability density function of a sequence composed of N values $u_1, u_2, \ldots, u_N$. A statistic X^2, is generated based upon the discrepancy between the two curves. To determine the value of X^2, the N values are first ordered from minimum to maximum to form a sequence $u_1', u_2', \ldots, u_N'$. The range of values that u_i can take on is divided into k intervals. Two numbers are determined for each interval, namely the actual number of observed values l_i and the number of values L_i that should appear for the theoretical density function. A normalized discrepancy figure is calculated in the form of

$$\frac{(l_i - L_i)^2}{L_i} \tag{5.18}$$

for each interval to give

$$X^2 = \sum_{i=1}^{k} \frac{(l_i - L_i)^2}{L_i} \quad . \tag{5.19}$$

The statistic X^2 will be approximately chi-square with k-3 degrees of freedom for Gaussian distributions and k-4 degrees of freedom for non-Gaussian distributions. (The loss of degrees of freedom result from (1) X^2 being composed of only k-1 independent random variables $(l_i - L_i)^2/L_i$, (2) the mean of the underlying distribution being unknown, (3) the variance of the underlying distribution being unknown, and (4) the density being non-Gaussian.)

The X^2 statistic is calculated and compared with a chi-square distribution having n=k-4 (or n=k-3) degrees of freedom. For a γ region of significance and n degrees of freedom (32)

$$\text{Prob}\left[X^2 \leq \chi^2_{n;\gamma}\right] = \gamma \; . \tag{5.20}$$

If X^2 falls outside of this limit, it can be concluded that the sequence does not have the specified theoretical probability density.

Slakter (37) indicates that the Pearson Chi-square Goodness-of-Fit Test gives acceptable results for as low as N=10 and for k=5 intervals.

Kolmogorov Test. The Kolmogorov Test (33, 34) (sometimes referred to as the Kolmogorov-Smirnov Goodness-of-Fit Test) measures the quality of the cumulative distribution of a sequence having N values using a statistic which represents the maximum deviation between the theoretical cumulative distribution function $P(u_i')$ and the calculated cumulative distribution function $P_N(u_i')$. The statistic is given by

$$D_N = \underset{\text{for all } u_i}{\text{Supremum}} \left| P_N(u_i') - P(u_i') \right| \tag{5.21}$$

where

$$P_N(u_i') = \frac{i}{N} \tag{5.22}$$

where u_i' represents the $i^{\underline{th}}$ value of the sequence of $\{u_n\}$ ordered from minimum to maximum. Since the absolute difference of $P(u_i')$ and $P_N(u_i')$ is used, the statistic is a "two-sided" statistic.

Unlike the statistic X^2 of the Pearson Chi-square Test, the statictic D_N is not dependent on the underlying distribution. Thus, one density function exists for D_N for all possible cumulative distributions. The density of D_N is discussed by Lindgren (34). For a sequence of N values and a level of significance γ the maximum allowable value for D_N can be determined. If D_N is greater than this constant, it is concluded that the sequence does not possess the specified theoretical cumulative distribution.

The effect of the Kolmogorov Test is to place confidence bounds about the theoretical cumulative distribution as illustrated by Figure 9. The calculated cumulative distribution must stay between these bounds at all points for the specified theoretical cumulative distribution to pass the test. It should be noted that this test is best applied to distributions having a continuous range of possible outcomes.

Slakter (37) indicates that the Kolmogorov Test does not give results that are as acceptable as the Pearson Chi-square Goodness-of-Fit Test for $N \leq 50$. The Kolmogorov Test is best used for sequences having $N > 50$.

Tests on Second-order Statistics

For discrete sequences the autocorrelation function becomes a grouping of second-order statistics. For a sequence containing N+T values, the second order statistics are given by

$$\hat{R}_u(\tau) = \frac{1}{N} \sum_{j=1}^{N} u_j u_{j+\tau} \text{ for } \tau = 0, 1, \ldots, T \tag{5.23}$$

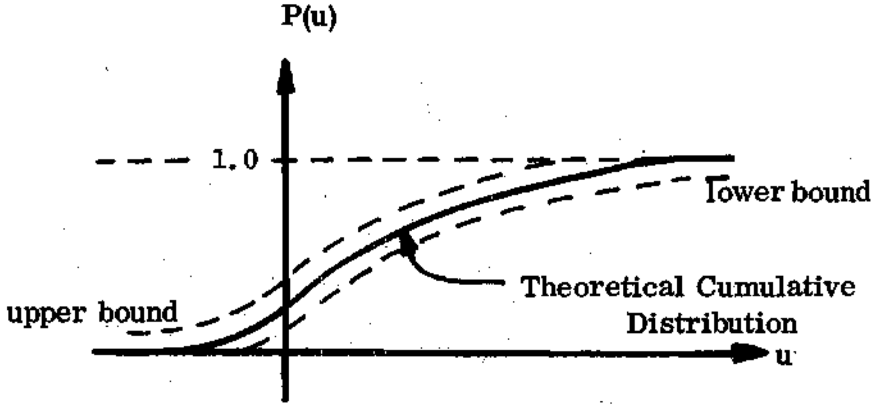

Figure 9. Confidence Bounds on Kolmogorov Test.

where τ is the separation between values of the sequence and T is the maximum separation considered. Normalized second-order statistics are given by

$$\hat{\rho}_u(\tau) = \frac{\hat{R}_u(\tau) - \bar{u}^2}{s^2} . \tag{5.24}$$

Two tests have been developed for comparing the calculated second-order statistics of the sequence with the specified theoretical second-order statistics. One test requires no knowledge of the variance of the sequence. The other test is for Gaussian sequences and uses a knowledge of the variance. This latter test can be extended for use with the output sequence of the Predistorted Transformed Gaussian Method by means of the input/output autocorrelation relationships developed for the nonlinear zero-memory filters of Chapter III.

Sequences Having Unknown Variance (35).* Most attempts to place confidence bounds about second-order statistics have been based upon the assumption that the distribution of the autocorrelation $R_u(\tau)$ (for a particular τ) is Gaussian. For cases in which N is large, a Gaussian approximation is valid. For a sequence of size N having a variance of σ^2 the calculated autocorrelation $\hat{R}_u(\tau)$ is related to the theoretical autocorrelation $R_u(\tau)$ for a 95% level of significance by (35)

$$\text{Prob}\left[\ -1.96 \le \frac{\left[\hat{R}_u(\tau)-R_u(\tau)\right]\sqrt{N}}{\sigma} \le 1.96\right] = \gamma = .95\ . \qquad (5.25)$$

This results in

$$\text{Prob}\left[\hat{R}_u(\tau) - 1.96\frac{\sigma}{\sqrt{N}} \le R(\tau) \le \hat{R}(\tau) + 1.96\frac{\sigma}{\sqrt{N}}\right] = .95\ . \qquad (5.26)$$

The Fisher's z statistic (37) (a statistic that is asymptotically Gaussian) offers a method for characterizing the calculated distribution $\hat{R}_u(\tau)$ based upon the theoretical distribution $R_u(\tau)$. A statistic z_τ can be formed, based upon a knowledge of the normalized autocorrelation $\rho_u(\tau)$, using the relationship

$$z_\tau = \frac{1}{2}\log_e \frac{1+\rho_u(\tau)}{1-\rho_u(\tau)} \text{ for } \tau \ne 0. \qquad (5.27)$$

---

*The ideas described in this section were developed by Dady T. Patel of the School of Mechanical Engineering at the Georgia Institute of Technology and by Dr. J. J. Goode of the School of Mathematics at the Georgia Institute of Technology.

Then for the calculated normalized autocorrelation $\hat{\rho}_u(\tau)$, a new statistic $\hat{z}_\tau$ is given by

$$\hat{z}_\tau = \frac{1}{2} \log_e \frac{1+\hat{\rho}_u(\tau)}{1-\hat{\rho}_u(\tau)} \text{ for } \tau \neq 0. \tag{5.28}$$

Now the statistic $\sqrt{N}\ (z_\tau - \hat{z}_\tau)$ is asymptotically Gaussian with mean zero and variance of unity if $N > 24$. This results in the probability statement

$$\Pr\left\{-1.96 < \sqrt{N-2}\left[z_\tau - \hat{z}_\tau\right] < 1.96\right\} = .95. \tag{5.29}$$

The confidence region for $N > 24$ then becomes

$$\tanh(z_\tau - 1.96/\sqrt{N-2}) > \hat{\rho}(\tau) > \tanh(z_\tau + 1.96/\sqrt{N-2}) . \tag{5.30}$$

This confidence region statement implies that the specified theoretical autocorrelation will be rejected if $\hat{\rho}_u(\tau)$ exceeds the bounds of the confidence region.

It should be noted that in the development of equation (5.30) a knowledge of the variance was not required.

Gaussian Sequences Having Known Variance. The method for calculating the second-order statistics of the calculated autocorrelation was given by equation (5.23). The mean of the autocorrelation is

$$E\left[\hat{R}_u(\tau)\right] = \frac{1}{N} \sum_{i=1}^{N} E\left[u_i u_{i+\tau}\right] = \sigma^2 \rho_u(\tau) - m_u = R_u(\tau) . \tag{5.31}$$

The variance of the calculated autocorrelation is

$$\mathrm{Var}\left[\hat{R}_u(\tau)\right] = E\left\{\left[\hat{R}_u(\tau) - R_u(\tau)\right]^2\right\}$$

$$= E\left[\hat{R}_u^2(\tau)\right] - R_u^2(\tau) \qquad (5.32)$$

where

$$E\left[\hat{R}_u^2(\tau)\right] = \left\{\left[\frac{1}{N}\sum_{i=1}^{N} u_i u_{i+\tau}\right]^2\right\}. \qquad (5.33)$$

Expanding equation (5.33) and collecting terms yields

$$E\left[\hat{R}_u^2(\tau)\right] = \frac{1}{N^2}\left\{\sum_{i=1}^{N} E\left[u_i^2 u_{i+\tau}^2\right]\right.$$

$$\left. + 2\sum_{i-1}^{N-1}\sum_{j=i+1}^{N} E\left[u_i u_{i+\tau} u_j u_{j+\tau}\right]\right\}. \qquad (5.34)$$

Consider now the case that occurs when the underlying distribution is normal with a mean of zero and a variance of σ^2. In this case (39)

$$E[u_i u_{i+\tau}] = \sigma^2 \rho_u(\tau), \qquad (5.35)$$

$$E[u_i^2 u_{i+\tau}^2] = \sigma^4 + 2\sigma^2 \rho_u(\tau), \qquad (5.36)$$

and

$$E[u_i u_{i+\tau} u_j u_{j+\tau}] = \sigma^4 \rho_u(t) \quad . \qquad (5.37)$$

Substitution of the results of equations (5.35), (5.36), and (5.37) into equation (5.34) yields

$$E\left[\hat{R}_u^2(\tau)\right] = \frac{1}{N^2}\left\{\sum_{i=1}^{N}\left[\sigma^4 + 2\sigma^2\rho_u(\tau)\right]\right.$$

$$\left. + \ 2 \sum_{i=1}^{N-1}\sum_{j=i+1}^{N} \sigma^4 \rho_u^{\ 2}(\tau) \right\}, \tag{5.38}$$

which reduces to

$$E\left[\hat{R}_u^{\ 2}(\tau)\right] = \frac{1}{N^2}\left\{N\left[\sigma^4 + 2\sigma^2\rho_u(\tau)\right] + N(N-1)\sigma^4\rho_u^{\ 2}(\tau)\right\}$$

$$= \frac{1}{N}\left\{\sigma^4 + 2\sigma^2\rho_u(\tau) + (N-1)\sigma^4\rho_u^{\ 2}(\tau)\right\}, \tag{5.39}$$

or

$$\mathrm{Var}\left[\hat{R}_u(\tau)\right] = \frac{1}{N}\left\{\sigma^4 + 2\sigma^2\rho_u(\tau) - \sigma^4\rho_u^{\ 2}(\tau)\right\}$$

$$= \frac{1}{N}\left\{2\sigma^2\rho_u(\tau) + \sigma^4\left[1 - \rho_u^{\ 2}(\tau)\right]\right\}. \tag{5.40}$$

An acceptable region of significance is $\gamma = .95$. The limits set by this region of significance offer a convenient basis for rejecting or not rejecting the specified theoretical autocorrelation since

$$R_u(\tau) - 2\mathrm{s.d.}\left[\hat{R}_u(\tau)\right] \leq \hat{R}_u(\tau) \leq R_u(\tau) + 2\mathrm{s.d.}\left[\hat{R}_u(\tau)\right] \tag{5.41}$$

where $\mathrm{s.d.}\left[\hat{R}_u(\tau)\right]$ is the standard deviation of $\hat{R}_u(\tau)$ and is found by taking the square root of the variance of $\hat{R}_u(\tau)$.

An autocorrelation test for Gaussian sequences having a mean of zero and a variance of σ^2 may now be stated. A specified autocorrelation $R_n(\tau)$ (second-order statistics) for a Gaussian sequence having a mean of zero and a variance of σ^2 will be rejected if

$$\hat{R}_u(\tau) < R_u(\tau) - 2\left[\frac{1}{N}\left\{2\sigma^2\rho_u(\tau) + \sigma^4\left[1 - \rho_u^2(\tau)\right]\right\}\right]^{\frac{1}{2}}, \tag{5.42}$$

or if

$$\hat{R}_u(\tau) > R_u(\tau) + 2\left[\frac{1}{N}\left\{2\sigma^2\rho_u(\tau) + \sigma^4\left[1 - \rho_u^2(\tau)\right]\right\}\right]^{\frac{1}{2}}. \tag{5.43}$$

The Predistorted Transformed Gaussian Method for generating pseudo-random sequences having a specified probability density and a specified autocorrelation transforms a correlated Gaussian sequence into the desired output sequence by a nonlinear zero-memory filter. Since the zero-memory filter has a one-to-one autocorrelation relationship between input and output sequences, this autocorrelation test may be applied to the output sequence. Each output second-order statistic may be transformed to give the corresponding input second-order statistic. The limits for the autocorrelation test may be set for a chosen level of significance in terms of the Gaussian sequence. The output equivalent of these limits may be determined by transforming these Gaussian limits back through the zero-memory element.

In conclusion, this chapter has discussed the statistical tests used to evaluate the performance of the designs for several interesting cases. The next chapter will present an evaluation of these designs.

CHAPTER VI

EVALUATION STUDIES

The design procedure of Chapter IV was implemented and tested for ten specified probability densities and autocorrelations as follows:

1. Uniform Density--Exponential Autocorrelation
2. Uniform Density--Triangular Autocorrelation
3. Uniform Density--Sin(x)/x Autocorrelation
4. Random Telegraph Signal Density--Exponential Autocorrelation
5. Random Telegraph Signal Density--Triangular Autocorrelation
6. Random Telegraph Signal Density--Sin(x)/x Autocorrelation
7. Chi-square Density with one degree of freedom--Exponential Autocorrelation
8. Chi-square Density with one degree of freedom--Triangular Autocorrelation
9. Rayleigh Density*--Exponential Autocorrelation
10. Rayleigh Density--Triangular Autocorrelation

The purpose of this chapter is to report the evaluation studies on the sequences produced for these cases. These specified probability densities and autocorrelations are representative of the general class of random number sequences

---

*The Rayleigh Density cases were produced by applying Envelope Detection (40) concepts to the Predistorted Transformed Gaussian Method.

to which the method applies and illustrates the properties of the method.

Introduction

The implemented cases* make use of a portion of the design procedure discussed in Chapter IV. A non-orthogonal sequence $\{x_n\}$ was used as input without correction being introduced in the design of the linear memory filter. The output autocorrelations that were specified were "distorted" from the autocorrelations which would be specified when "correction" for $\{x_n\}$ was included in the linear memory filter design.

The raw data for the design of the specified output sequences is presented in the Appendices. The specified output autocorrelation and the autocorrelation required as input to the nonlinear zero-memory filter are given in Appendix I. The filter coefficients for the linear memory filter for each design are given in Appendix II.

The random number sequence used as input was produced by a generator developed by Brown and Rowland (39). The generator was selected for its ability to produce sequences having stationary mean and variance properties. A discussion of the tests performed on this generator and a second generator, the UNIVAC 1108 standard MATHPACK generator RANDN, is given in Appendix III.

---

*The implementation and testing of the preliminary design study was performed on the IBM 7094 computer located at the U. S. Army Missile Command Computation Center at the Redstone Arsenal, Alabama. The implementation and testing of the final design procedure was performed on the UNIVAC 1108 and the Burroughs 5500 computers located at the Rich Electronic Computer Center on the Georgia Institute of Technology campus in Atlanta, Georgia.

Statistical Tests on Output Sequences of the Linear Memory Filter

The correlated Gaussian sequences produced as output from the linear memory filter in each of the ten design cases were tested with respect to stationarity and the basic random properties of the sequences, namely the mean square property, the probability density property, and the autocorrelation property.

Mean Square Property

For each design case 100 Gaussian sequences, containing 80 values each, were generated and tested using the mean and variance tests discussed in Chapter V. The results of these tests are summarized in Table 1. Each sequence was tested against a theoretical mean of $m_y = 0$ and variance of $\sigma^2 = 1$ at a 95% level of significance. For 100 sequences having the theoretical mean and theoretical variance, there is a probability that five sequences will fail the test on mean and five sequences will fail the test on variance. No design case had more than five sequences for which the theoretical value of the mean or the theoretical value of the variance was rejected. No significant variations in the calculated values of the mean or variance were observed to indicate that the process was nonstationary.

Probability Density Property

For each design case a sequence containing 10,000 values was generated and tested using the Pearson Chi-square Goodness-of-Fit Test and the Kolmogorov Test. For the test results shown in Tables 2 and 3, all tests passed at a 95% level of significance with the exception of the Pearson Chi-square Test on the sequence

Table 1. Mean Square Property of the Output Sequences

from the Linear Memory Filter Output

Specified Output Autocorrelation for $\{z_n\}$	Specified Output Density for $\{z_n\}$	For 100 Tests at $\gamma = .95$	
		Number of Tests which fail for Theoretical Mean	Number of Tests which fail for Theoretical Variance
Exponential	Uniform	5	3
Exponential	Random Telegraph Signal	5	4
Exponential	Chi-square	2	4
Exponential	Rayleigh	2	3
Triangular	Uniform	2	4
Triangular	Random Telegraph Signal	5	3
Triangular	Chi-square	0	2
Triangular	Rayleigh	0	4
Sin(x)/x	Uniform	0	4
Sin(x)/x	Random Telegraph Signal	0	0

required for the design case specified to have an output sequence having a Rayleigh density and exponential autocorrelation. Since the sequence passes the Kolmogorov Test, the sequence is assumed to have the required theoretical density. Failure of the Pearson Chi-square Test indicates that the sequence values do not appear in sufficient numbers to satisfy the theoretical requirements

Table 2. Results of Kolmogorov Test on Output Sequences from the Linear Memory Filter

Specified Output Autocorrelation for $\{z_n\}$	Specified Output Density for $\{z_n\}$	Number of Sequence Values Violating Limits for Region of Significance of γ*				
		80%	85%	90%	95%	99%
Exponential	Uniform	0	0	0	0	0
Exponential	Random Telegraph Signal	0	0	0	0	0
Exponential	Chi-square	17	0	0	0	0
Exponential	Rayleigh	123	15	0	0	0
Triangular	Uniform	0	0	0	0	0
Triangular	Random Telegraph Signal	0	0	0	0	0
Triangular	Chi-square	0	0	0	0	0
Triangular	Rayleigh	0	0	0	0	0
Sin(x)/x	Uniform	0	0	0	0	0
Sin(x)/x	Random Telegraph Signal	0	0	0	0	0

*As the region of significance increases, the limits of the test increase making the test less rigid.

Table 3. Results of Pearson Chi-square Test on Output Sequences from the Linear Memory Filter

Specified Output Autocorrelation for $\{z_n\}$	Specified Output Density for $\{z_n\}$	For 397 Degrees of Freedom: Value of γ which Corresponds to Calculated χ^2 *
Exponential	Uniform	.31
Exponential	Random Telegraph Signal	.18
Exponential	Chi-square	.64
Exponential	Rayleigh	.98
Triangular	Uniform	.35
Triangular	Random Telegraph Signal	.35
Triangular	Chi-square	.39
Triangular	Rayleigh	.76
Sin(x)/x	Uniform	.73
Sin(x)/x	Random Telegraph Signal	.79

*The test will pass for any region of significance larger than this value.

for strategic regions of the density curve.

For each design case, the sequence of 10,000 values was subdivided into sequences of 500 values to study the stationarity of the probability density function. No significant variations caused by nonstationarity were noted in the results.

Autocorrelation Property

For each design case a sequence containing 10,000 values was generated and tested for the autocorrelation property using the tests of Chapter V for sequences having unknown variance and for Gaussian sequences having known variance. The autocorrelation required for each sequence is shown graphically in Figures 10 through 19.* (These figures also include the autocorrelation specified for the output sequence for each design case.) Of the ten design cases four design cases, namely the cases with output sequences specified to have exponential autocorrelations, had linear memory filters designed directly. The remaining six design cases possessed autocorrelation requirements which required design of the linear memory filter by the Modified Pakov Method using optimization techniques. In all ten design cases, the second-order statistics passed both tests.

It should be noted that the nonlinear zero-memory filter tends to decorrelate sequences. The input normalized autocorrelation is consistently

---

*The difference between the autocorrelations specified for these design cases and the autocorrelation that would be specified if the non-orthogonality of the input sequence had been considered in the design of the linear memory filter is so small that the curves would not be able to distinguish between the two.

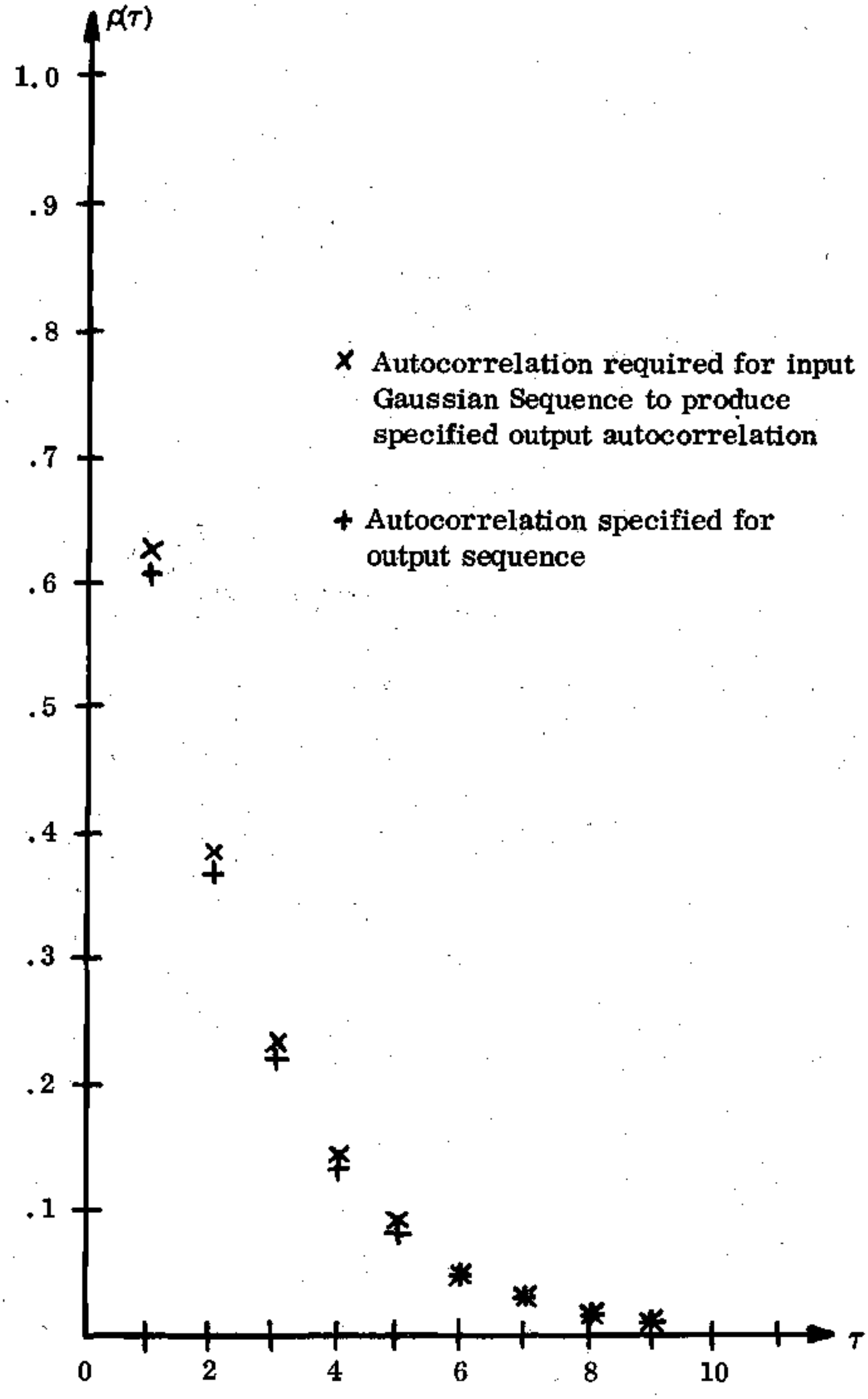

Figure 10. Input/Output Autocorrelation for Zero-memory Filter for Sequence Specified to have Uniform Density and Exponential Autocorrelation.

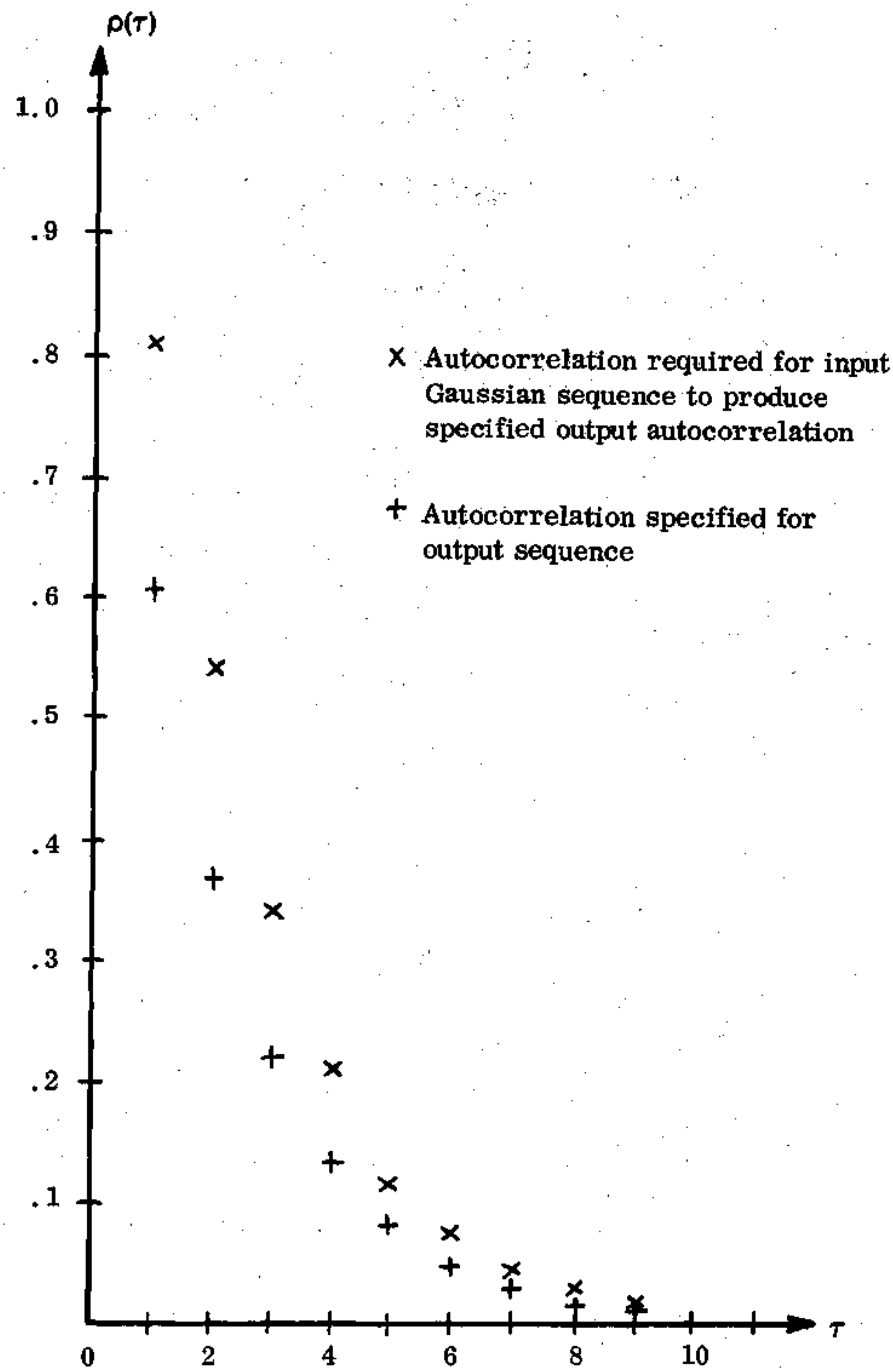

Figure 11. Input/Output Autocorrelation for Zero-memory Filter for Sequence Specified to have Random Telegraph Signal Density and Exponential Autocorrelation

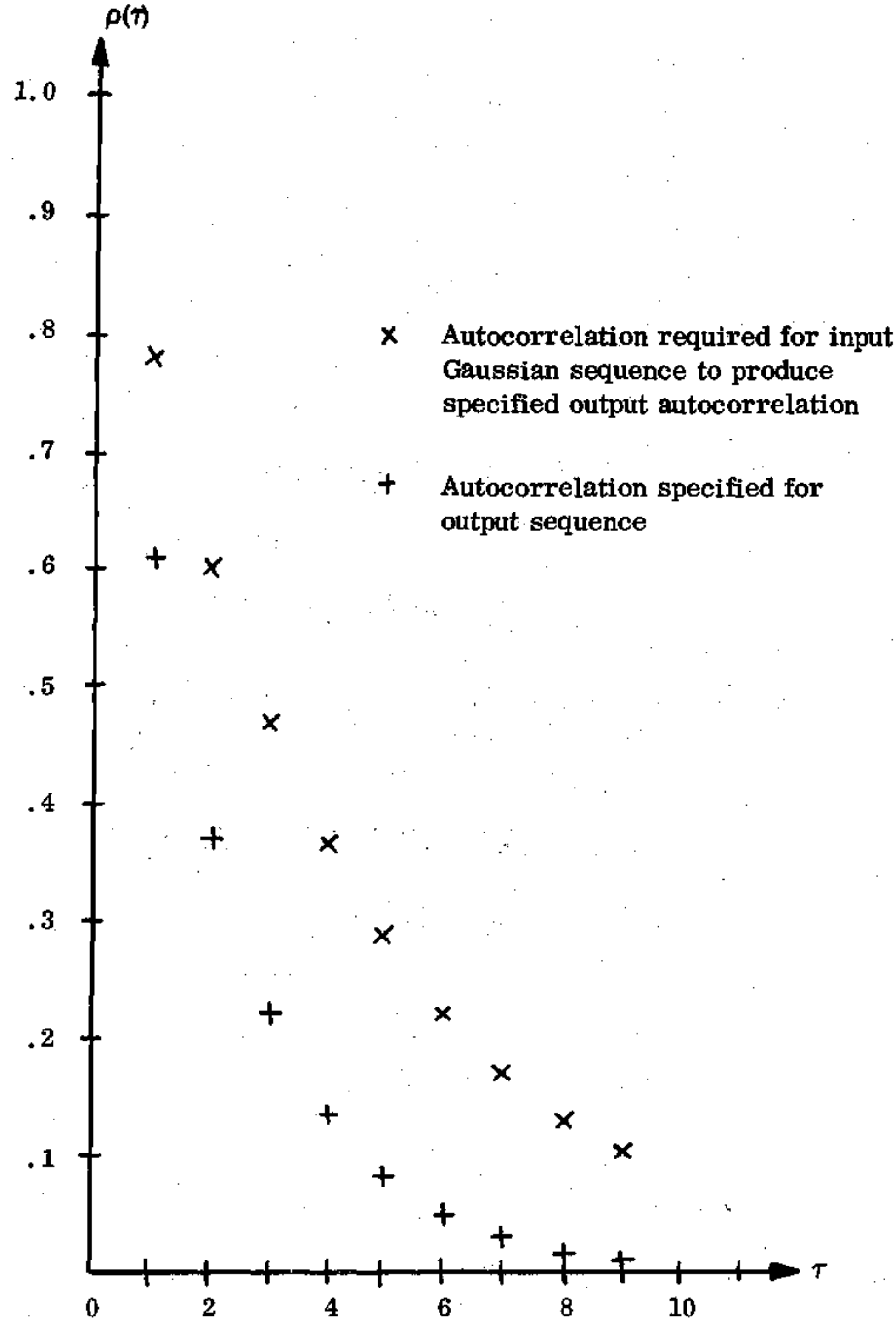

Figure 12. Input/Output Autocorrelation for Zero-memory Filter for Chi-square Density with one Degree of Freedom and Exponential Autocorrelation.

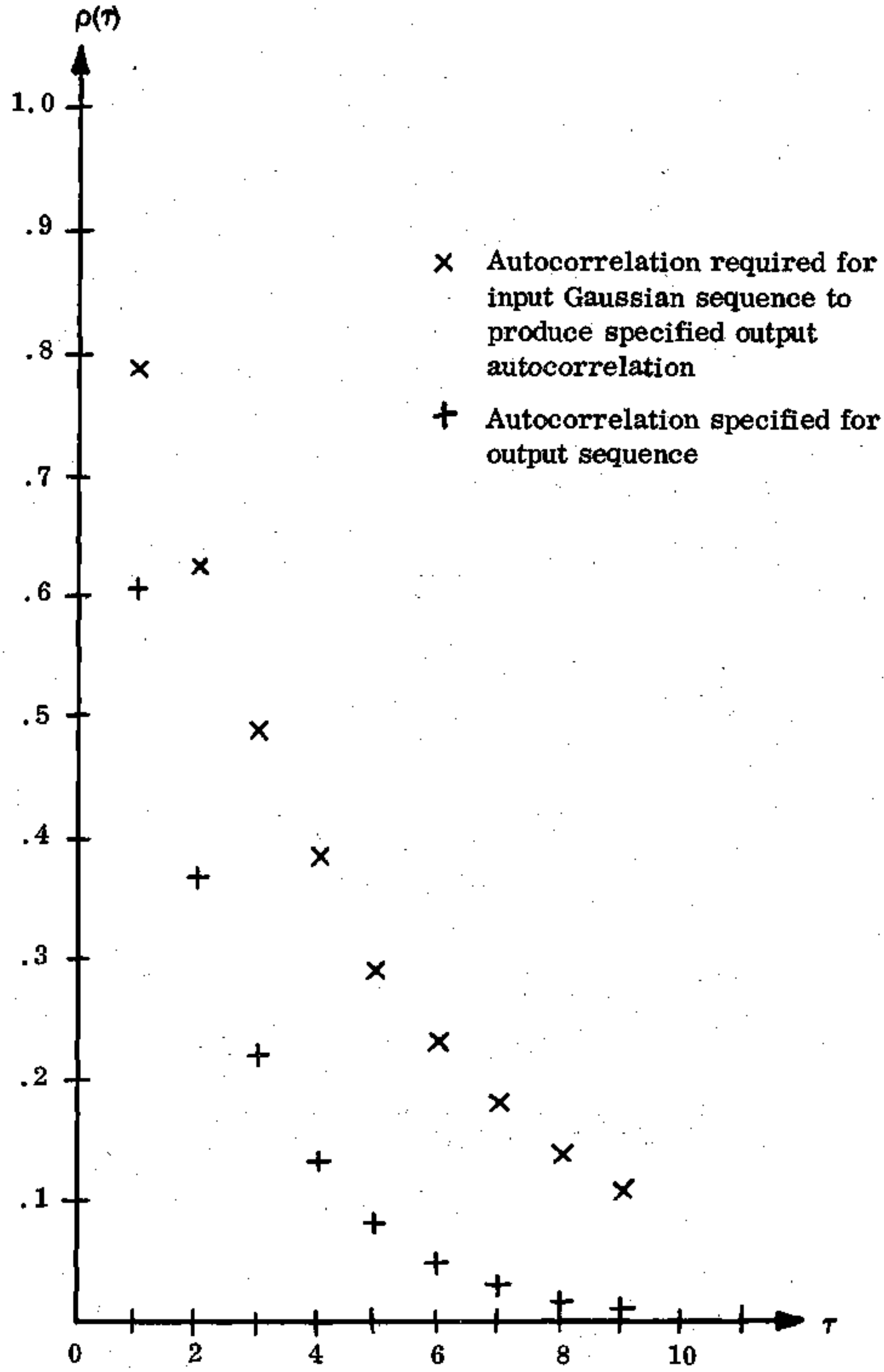

Figure 13. Input/Output Autocorrelation for Zero-memory Filter for Sequence Specified to have Rayleigh Density and Exponential Autocorrelation.

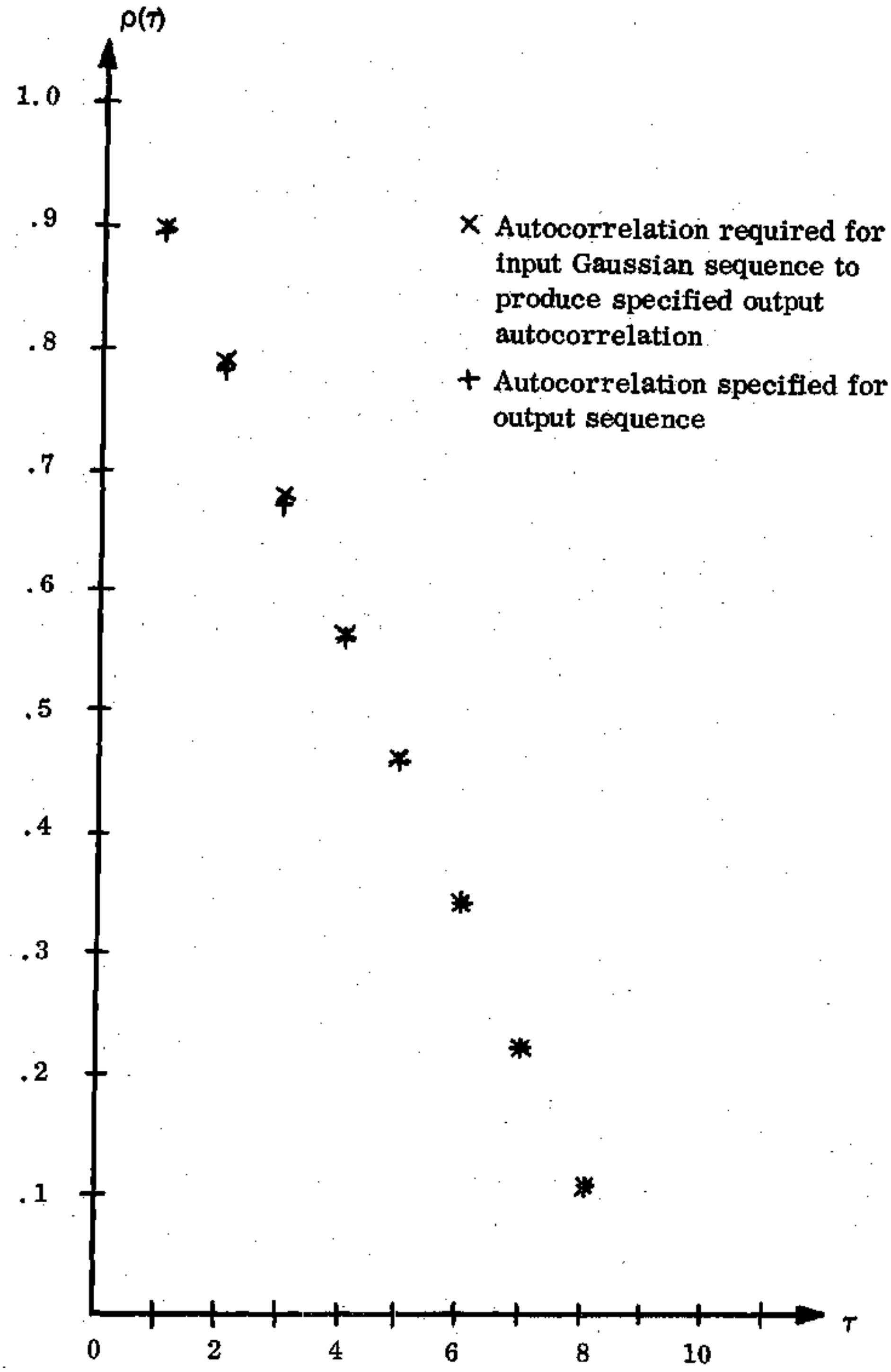

Figure 14. Input/Output Autocorrelation for Zero-memory Filter for Sequence Specified to have Uniform Density and Triangular Autocorrelation.

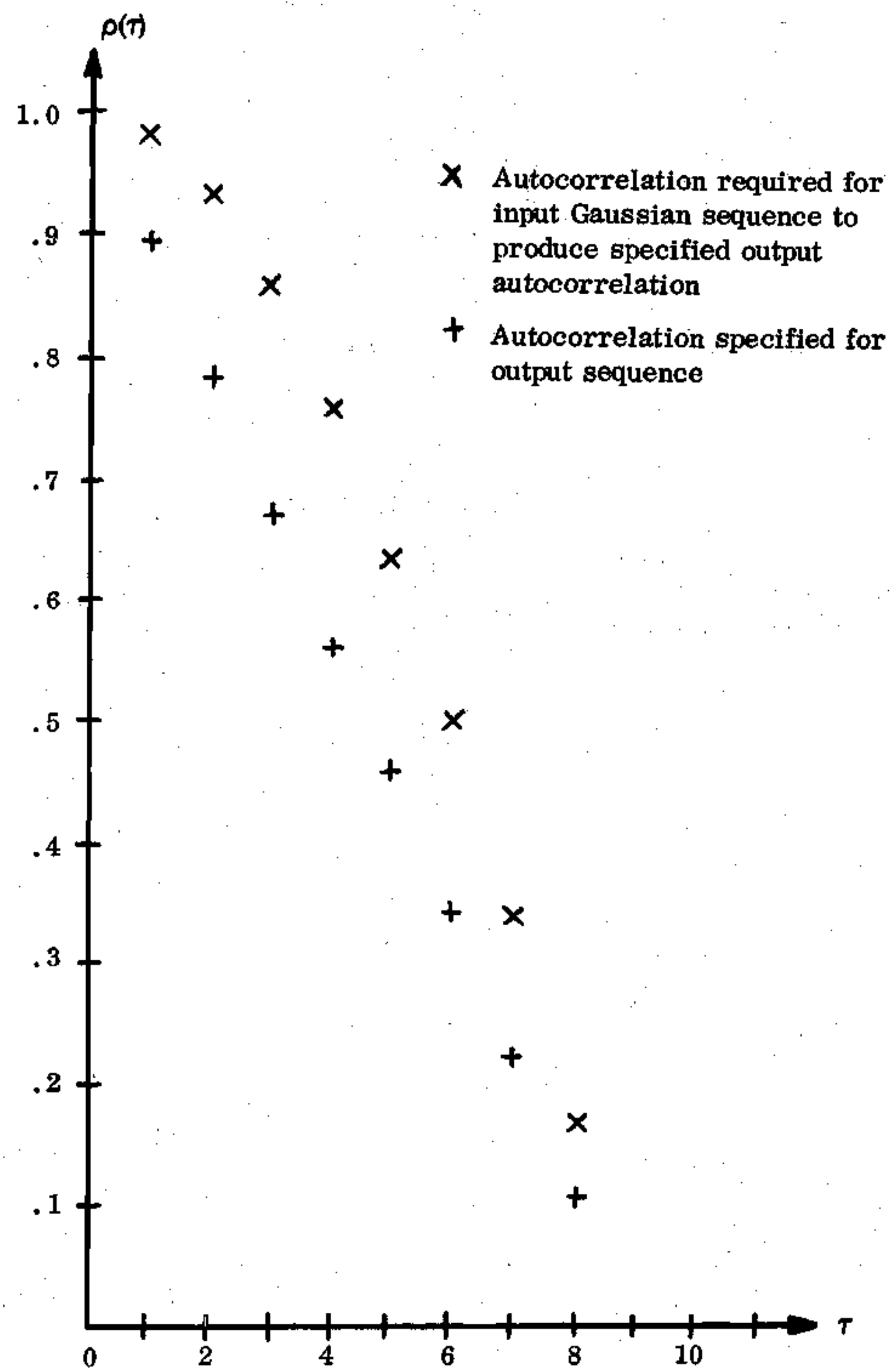

Figure 15. Input/Output Autocorrelation for Zero-memory Filter for Sequence Specified to have Random Telegraph Signal Density and Triangular Autocorrelation.

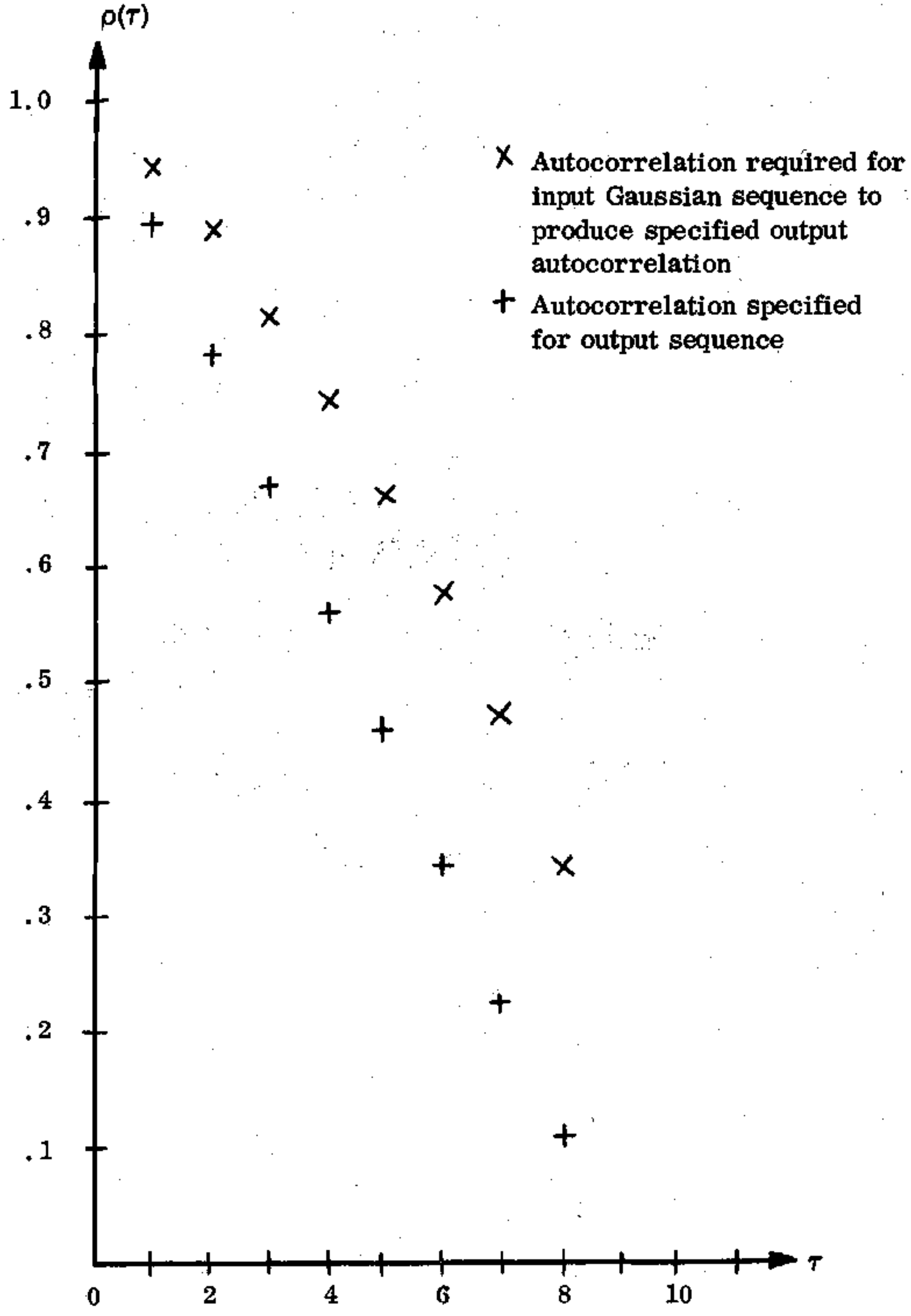

Figure 16. Input/Output Autocorrelation for Zero-memory Filter for Sequence Specified to have Chi-square Density with One Degree of Freedom and Triangular Autocorrelation.

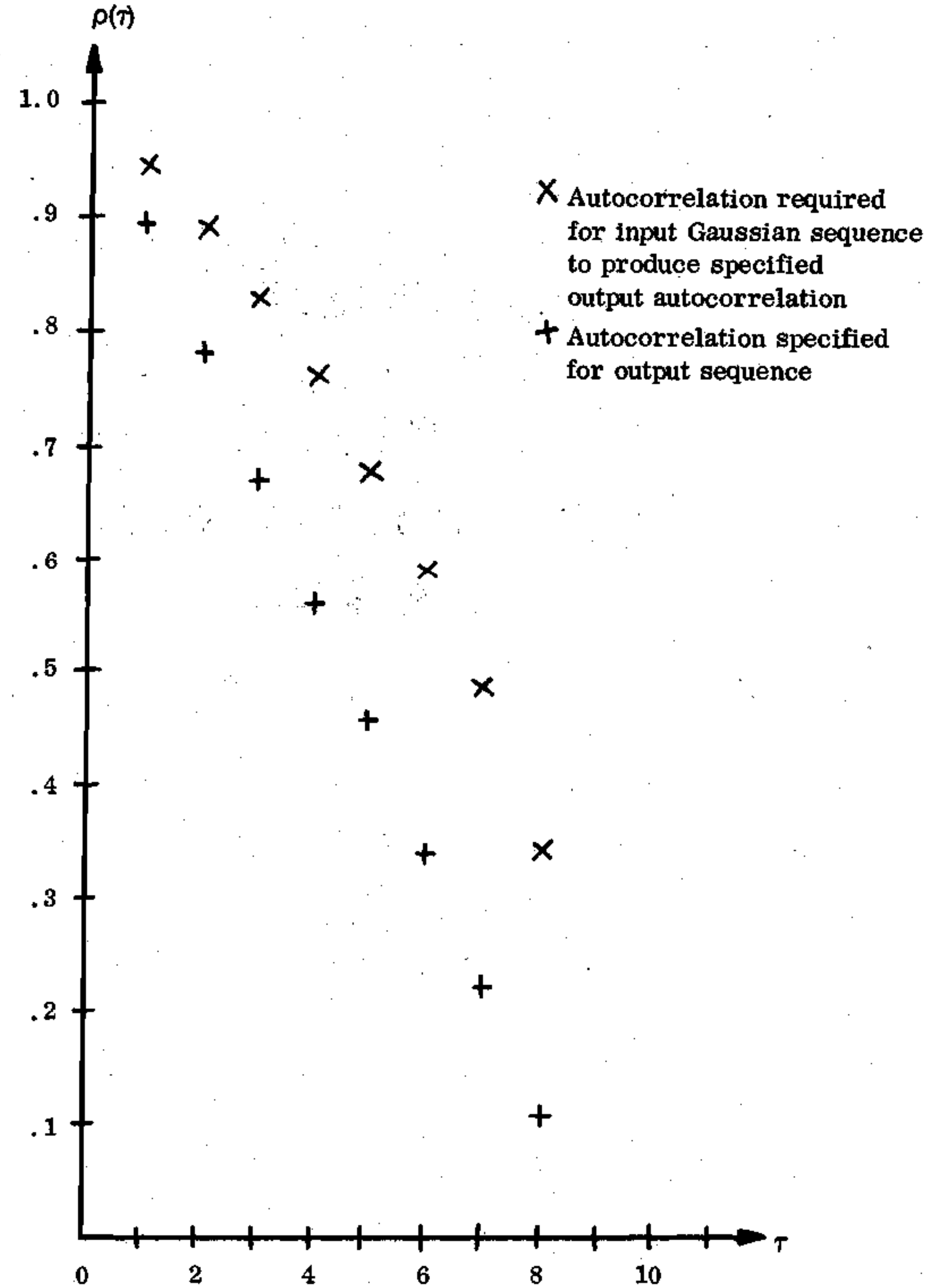

Figure 17. Input/Output Autocorrelation for Zero-memory Filter for Sequence Specified to have Rayleigh Density and Triangular Autocorrelation.

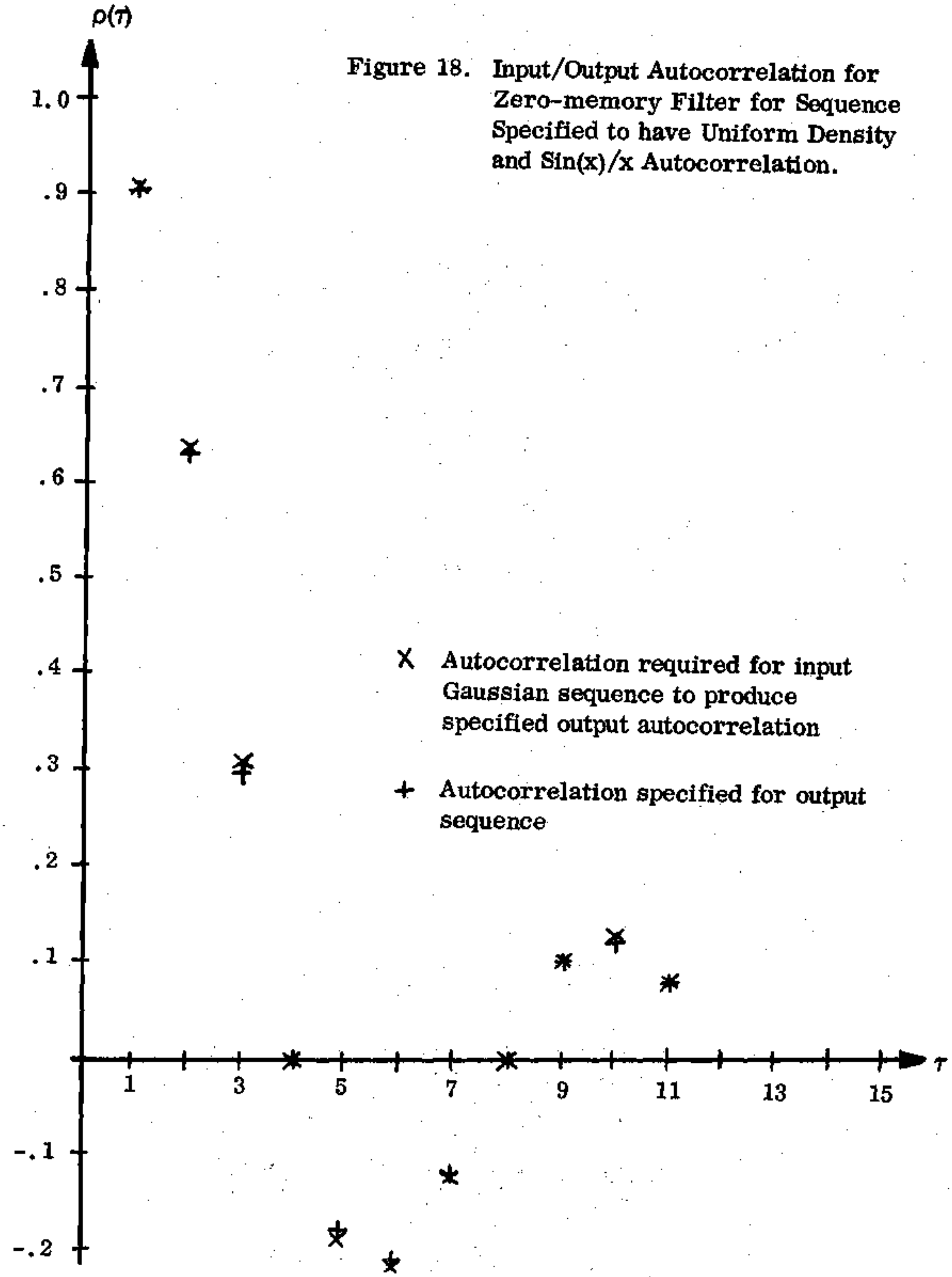

Figure 18. Input/Output Autocorrelation for Zero-memory Filter for Sequence Specified to have Uniform Density and Sin(x)/x Autocorrelation.

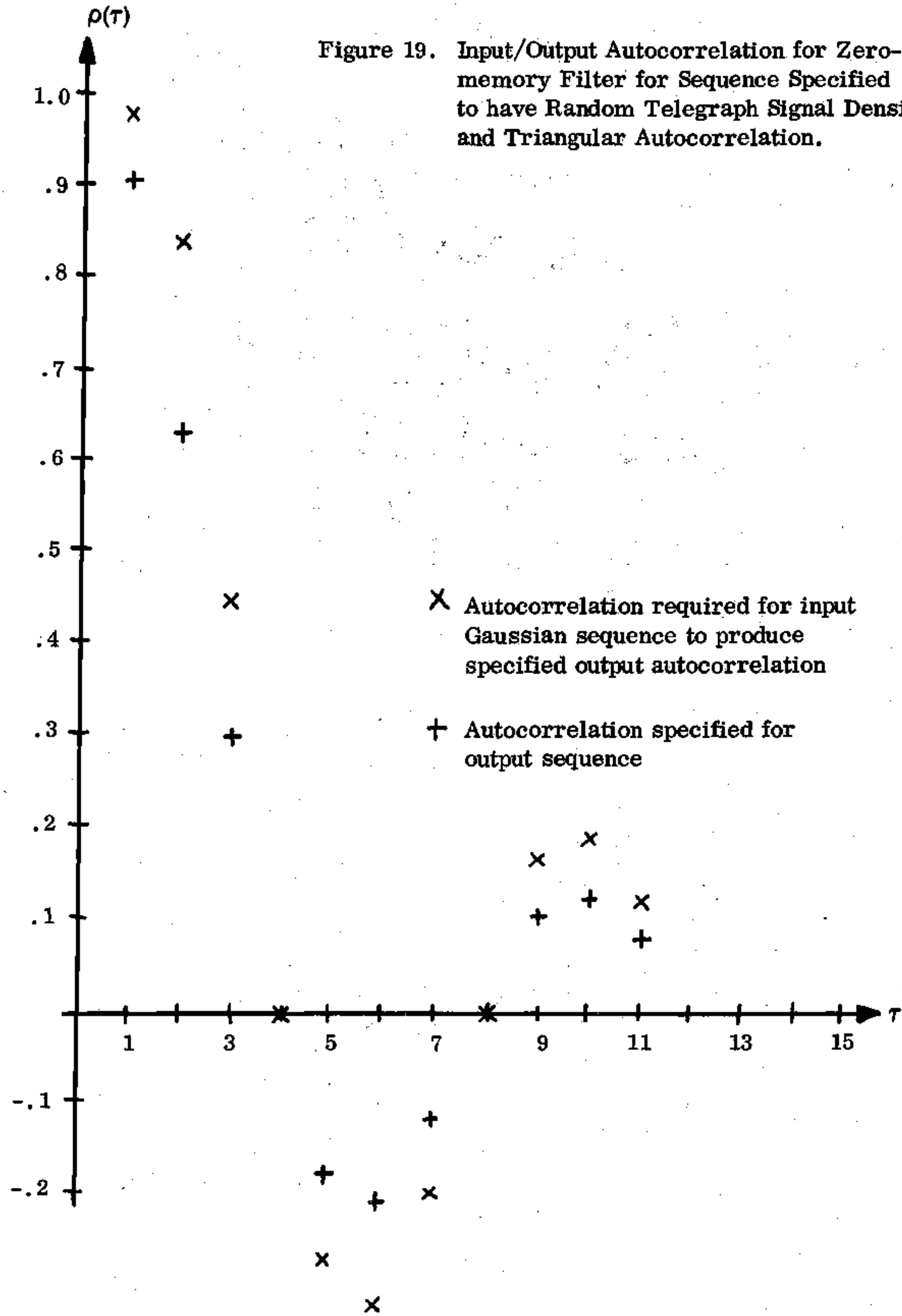

Figure 19. Input/Output Autocorrelation for Zero-memory Filter for Sequence Specified to have Random Telegraph Signal Density and Triangular Autocorrelation.

higher than the output normalized autocorrelation. This point contributes significantly to the design problems of the linear memory filter. The additional required input autocorrelation often results in the requirement for a linear memory filter than cannot be designed directly.

Statistical Tests on Output Sequences

Output sequences for the ten design cases were tested with respect to stationarity and two random properties, namely the probability density property and the autocorrelation property.

Probability Density Property

For each design case a sequence containing 10,000 values was generated and tested using the Pearson Chi-square Goodness-of-Fit Test and the Kolmogorov Test. For the results presented in Tables 4 and 5, eight of the sequences passed both tests at a 95% level of significance. Each of the two sequences specified to have a chi-square density with one degree of freedom failed at least one of the tests. A sequence having a chi-square density with one degree of freedom must contain a large number of values near zero. The tests on these two sequences emphasize a difficulty in generating large numbers of values in a sequence near zero while maintaining the correct emphasis in all strategic areas of the density curve. The failure of these tests indicates the need to make a prudent choice of an input sequence to insure that the values of the input sequence are distributed properly to give the specified output sequence.

The output sequence for each of the design cases was subdivided into

Table 4. Results of Kolmogorov Test on Output Sequences

Specified Output Autocorrelation	Specified Output Density	Number of Sequence Values Violating Limits for Region of Significance of γ *				
		80%	85%	90%	95%	99%
Exponential	Uniform	0	0	0	0	0
Exponential	Random Telegraph Signal	--Not applied to this case--				
Exponential	Chi-square	2958	2440	2091	1727	26
Exponential	Rayleigh	227	38	0	0	0
Triangular	Uniform	0	0	0	0	0
Triangular	Random Telegraph Signal	--Not applied to this case--				
Triangular	Chi-square	635	273	35	0	0
Triangular	Rayleigh	0	0	0	0	0
Sin(x)/x	Uniform	0	0	0	0	0
Sin(x)/x	Random Telegraph Signal	--Not applied to this case--				

*As the region of significance increases, the limits of the test increase making the test less rigid.

Table 5. Results of Pearson Chi-square Test on Output Sequences

Specified Output Autocorrelation	Specified Output Density	Value of γ which Corresponds to Calculated χ^2 *	Number of Degrees of Freedom
Exponential	Uniform	.06	396
Exponential	Random Telegraph Signal	.19	1
Exponential	Chi-square	Off Scale	396
Exponential	Rayleigh	.44	396
Triangular	Uniform	.62	396
Triangular	Random Telegraph Signal	.95	1
Triangular	Chi-square	Off Scale	396
Triangular	Rayleigh	.54	396
Sin(x)/x	Uniform	.89	396
Sin(x)/x	Random Telegraph Signal	.21	1

*The test will pass for any region of significance larger than this value.

sequences of 500 values to study the stationarity of the probability density function. No significant variations due to non-stationarity were noticed in the results.

Autocorrelation Property

For each design case a sequence containing 10,000 values was generated and tested for the autocorrelation property using the test for sequences having unknown variance and the modified test for Gaussian sequences having known variance discussed in Chapter V. The autocorrelation required for each sequence is shown graphically in Figures 10 through 19. In all ten design cases the output sequence possessed values having the specified autocorrelation functions within the mathematical limits as prescribed by the tests for a 95% region of significance. The results were equally satisfactory for linear memory filters designed directly and designed by the Modified Pakov Method employing optimization techniques.

Conclusions

The evaluation studies of the ten design cases indicate that the design procedure discussed in Chapter IV will produce random sequences having a specified probability density and specified autocorrelation. The output sequences generated by the design cases possessed the required output autocorrelation when tested at a 95% level of significance. The output sequences required to have Uniform densities, Random Telegraph Signal densities, and Rayleigh densities possessed the required probability density when tested at a 95% level of significance. As indicated in the probability density tests on the sequences having a chi-square density with one degree of freedom, a sequence having a high concentration of values in a localized region of the probability density curve will require special attention to be given to the probability density property of the input sequence in order to be able to pass the probability density tests.

The test results were equally satisfactory for the design cases having a linear memory filter designed directly and for the design cases having a linear memory filter designed using the Modified Pakov Method employing optimization techniques.

CHAPTER VII

CONCLUSIONS

Design Method

It is concluded from the evaluation studies of Chapter VI that a large class of sequences of pseudo-random numbers $\{z_n\}$ having a specified probability density function p(z) and a specified autocorrelation $R_z(\tau)$ can be generated using the Predistorted Transformed Gaussian Method, illustrated in Figure 4, from non-orthogonal input sequences $\{x_n\}$. The design procedure for implementing this technique includes the following steps.

(1) Design the nonlinear zero-memory filter to transform a Gaussian sequence $\{y_n\}$ having a mean of zero and a variance of unity into a sequence $\{z_n\}$ having the desired probability density. The relation which must be implemented is

$$z_i = g(y_i) = P_z^{-1}\left[\mathrm{Erf}(y_i)\right], \tag{4.1}$$

and this step can be carried out for any cumulative distribution $P_z(z)$ for which an inverse P_z^{-1} exists.

(2) Calculate the normalized autocorrelation $\rho_y(\tau)$ required for the input sequence to the zero-memory filter to produce the desired output autocorrelation $R_z(\tau)$. The autocorrelation can be determined in general from the relation

$$R_z(\tau) = \frac{1}{2\pi\sqrt{1-\rho_y^2(\tau)}} \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(y_1)g(y_2) \exp\left[-\frac{y_1^2-2\rho_y(\tau)y_1y_2-y_2^2}{2(1-\rho_y^2(\tau))}\right] dy_1dy_2. \quad (4.2)$$

In equation (4.2) it should be noted that g(y) is determined by step (1) and $R_z(\tau)$ is a specified parameter. Thus, $\rho_y(\tau)$ is the only unknown in equation (4.2), and it can always be determined numerically by the brute force technique of fixing τ and tabulating the double integral as a function of $\rho(\tau)$. Two examples of the brute force approach, namely the Rayleigh Density design cases, were worked out using series to approximate the integral.

In many cases of interest, g(y) has a tractable form such that the double integral of equation (4.2) can be evaluated analytically. Eight design cases of this type are tabulated in the thesis.

When implementing the design step to calculate the normalized autocorrelation $\rho_y(\tau)$, the solution of equation (4.2) could prove to be an untractable mathematical problem. This problem can be circumvented by using either the Characteristic Function Method or the Series Method.

(3) Design the linear memory filter using the Modified Pakov Method. Three major steps are involved in this design.

Step 1: Measure the second-order statistics of the input sequence.

Step 2: Set up design equations by determining the second-order statistics that must be introduced by the linear memory filter. The design equations are given by

$$
\left.
\begin{aligned}
f_1 &= \hat{\alpha}_1^2 + \hat{\alpha}_2^2 + \ldots + \alpha_N^2 - a_0 = 0 \\
f_2 &= \hat{\alpha}_1 \alpha_2 + \hat{\alpha}_2 \hat{\alpha}_3 + \ldots + \hat{\alpha}_{N-1} \hat{\alpha}_N - a_1 = 0 \\
&\vdots \\
f_N &= \hat{\alpha}_1 \hat{\alpha}_N - a_{N-1} = 0
\end{aligned}
\right\} \qquad (4.9)
$$

where the $\hat{\alpha}_i$ are the filter coefficients and the a_i are the second-order statistics to be introduced by the linear memory filter.

<u>Step 3:</u> Solve equation (4.9) for the $\hat{\alpha}_i$ by minimizing the functional

$$
F = \sum_{i=1}^{N} f_i^2 \qquad (4.10)
$$

using conventional optimization methods.

The accuracy is dependent upon the ability to minimize the functional produced in equation (4.10). An assessment of the error of the design method can be based upon the value of F. The design procedure is considered to be exact for functional minimums less than 10^{-6}. As the functional minimum increases above 10^{-6}, the design has a degree of error introduced by the design of the linear memory filter. For the sequences specified for evaluating the design method the largest functional minimum, F = .156, occurred for the sequence specified to have a Rayleigh Density and triangular autocorrelation with nine filter coefficients. The output sequence generated in this case possessed the specified probability density and autocorrelation within acceptable mathematical limits for a region of significance of $\gamma = .95$ for both parameters.

The design was implemented to produce ten sequences having specified probability densities and autocorrelations that are representative of the general class of random number sequences to which the method applies. The design cases included (1) both continuous and discrete probability densities, (2) autocorrelations having both positive and negative effects, (3) autocorrelations for which the double integral of equation (4.2) was evaluated both analytically and numerically, and (4) linear memory filters having both an exact design and no exact design.

Statistical tests were performed on the results of the ten design cases to establish (1) the stationarity and the basic random properties of the sequences, namely, (2) the mean square value property, (3) the probability density property, and (4) the autocorrelation property. The sequences proved to be stationary and produce the specified autocorrelation for a region of significance of $\gamma = .95$. The sequences produced the required probability density when tested at a $\gamma = .95$ region of significance, in all design cases except those requiring a high concentration of sequence values in a localized region of the probability density curve. These design cases require that special attention be given to the probability density properties of the input sequence to insure that the values of the sequence have good Gaussian characteristics. In all of the ten cases examined the procedure could be carried through with acceptable error.

Recommendations for Future Work

This research indicates the need for further work in several areas. The equations of (4.9) were solved directly for only four of the ten design studies. Clearly, the restriction for direct solution developed by Nakamura (30) is not sufficient. Additional investigation is needed upon the boundaries placed upon a_i for direct solution of equation (4.9).

Additional study is needed for the number of filter coefficients required to optimize the linear memory filter design. The empirical study cited in Chapter III showed that an improvement in filter design could be achieved for output sequences specified to have a triangular autocorrelation when the number of filter coefficients were increased to the order of 30 coefficients. On the other hand, no improvement was achieved when the number of coefficients were increased for the output sequences having sin(x)/x autocorrelation.

In summary, this thesis has presented a technique for generating pseudo-random sequences having specified probability density and specified autocorrelation and illustrated the technique for ten design cases. The statistical tests performed on these cases have shown that the technique gives results that are mathematically acceptable.

APPENDICES

APPENDIX I

INPUT/OUTPUT AUTOCORRELATION DATA FOR ZERO-MEMORY FILTER

The atuocorrelations (second-order statistics) specified for the ten implemented output sequences are given in Tables 6 through 15. The corresponding autocorrelations (second-order statistics) required as input to the nonlinear zero-memory filters for each case are also given in Tables 6 through 15.

Table 6. Input/Output Autocorrelation for Zero-memory Filter for Sequence Specified to have Uniform Density and Exponential Autocorrelation

Lag Coefficient	Input Autocorrelation	Normalized Output Autocorrelation
0	1.0000000	1.0000000
1	.6245346	.6065307
2	.3828646	.3678794
3	.2331302	.2231302
4	.1416042	.1353353
5	.0859328	.0820850
6	.0521310	.0497871
7	.0316213	.0301974
8	.0191798	.0183156
9	.0116332	.0111090

Table 7. Input/Output Autocorrelation for Zero-memory Filter for Sequence Specified to have Random Telegraph Signal Density and Exponential Autocorrelation

Lag Coefficient	Input Autocorrelation	Output Autocorrelation
0	1.0000000	1.0000000
1	.8150040	.6065307
2	.5462357	.3678794
3	.3433600	.2231302
4	.2109866	.1353353
5	.1285818	.0820850
6	.0781256	.0497871
7	.0474162	.0301974
8	.0287662	.0183156
9	.0174491	.0111090

Table 8. Input/Output Autocorrelation for Zero-memory Filter for Sequence Specified to have Chi-square Density with One Degree of Freedom and Exponential Autocorrelation

Lag Coefficient	Input Autocorrelation	Output Autocorrelation
0	1.0000000	3.0000000
1	.7788008	2.2130613
2	.6065307	1.7357589
3	.4723665	1.4462603
4	.3678794	1.2706706
5	.2865048	1.1641700
6	.2231302	1.0995741
7	.1737739	1.0603948
8	.1353353	1.0366313
9	.1053992	1.0222180
10	.0000000	1.0000000
11	.0000000	1.0000000

Table 9. Input/Output Autocorrelation for Zero-memory Filter for Sequence Specified to have Rayleigh Density and Exponential Autocorrelation

Lag Coefficient	Input Autocorrelation	Normalized Output Autocorrelation
0	1.0000	1.0000000
1	.7947	.6065307
2	.6251	.3678794
3	.4895	.2231302
4	.3825	.1353353
5	.2984	.0820850
6	.2327	.0497871
7	.1813	.0301974
8	.1413	.0183156
9	.1101	.0111090

Table 10. Input/Output Autocorrelation for Zero-memory Filter for Sequence Specified to have Uniform Density and Triangular Autocorrelation

Lag Coefficient	Input Autocorrelation	Normalized Output Autocorrelation
0	1.0000000	1.0000000
1	.8975984	.8888889
2	.7921596	.7777778
3	.6840403	.6666667
4	.5736065	.5555556
5	.4612318	.4444444
6	.3472964	.3333333
7	.2321858	.2222222
8	.1162897	.1111111

Table 11. Input/Output Autocorrelation for Zero-memory Filter for Sequence Specified to have Random Telegraph Signal Density and Triangular Autocorrelation

Lag Coefficient	Input Autocorrelation	Output Autocorrelation
0	1.0000000	1.0000000
1	.9848078	.8888889
2	.9396926	.7777778
3	.8660254	.6666667
4	.7660444	.5555556
5	.6427876	.4444444
6	.5000000	.3333333
7	.3420201	.2222222
8	.1736482	.1111111

Table 12. Input/Output Autocorrelation for Zero-memory Filter for Sequence Specified to have Chi-square Density with One Degree of Freedom and Triangular Autocorrelation

Lag Coefficient	Input Autocorrelation	Output Autocorrelation
0	1.0000000	3.0000000
1	.9428090	2.7777778
2	.8819171	1.5555556
3	.8164966	2.3333333
4	.7453560	2.1111111
5	.6666667	1.8888889
6	.5773503	1.6666667
7	.4714045	1.4444444
8	.3333333	1.2222222
9	.0000000	1.0000000
10	.0000000	1.0000000

Table 13. Input/Output Autocorrelation for Zero-memory Filter for Sequence Specified to have Rayleigh Density and Triangular Autocorrelation

Lag Coefficient	Input Autocorrelation	Normalized Output Autocorrelation
0	1.0000	1.0000000
1	.9488	.8888889
2	.8926	.7777778
3	.8308	.6666667
4	.7622	.5555556
5	.6850	.4444444
6	.5958	.3333333
7	.4885	.2222222
8	.3468	.1111111

Table 14. Input/Output Autocorrelation for Zero-memory Filter for Sequence Specified to have Uniform Density and Sin(x)/x Autocorrelation

Lag Coefficient	Input Autocorrelation	Normalized Output Autocorrelation
0	1.0000000	1.0000000
1	.9082762	.9003163
2	.6543894	.6366198
3	.3129780	.3001054
4	.0000000	.0000000
5	-.1882856	-.1800643
6	-.2217653	-.2122066
7	-.1345852	-.1286166
8	.0000000	.0000000
9	.1047087	.1000352
10	.1332346	.1273240
11	.0856837	.0818469

Table 15. Input/Output Autocorrelation for Zero-memory Filter for Sequence Specified to have Random Telegraph Signal Density and Sin(x)/x Autocorrelation

Lag Coefficient	Input Autocorrelation	Output Autocorrelation
0	1.0000000	1.0000000
1	.9877659	.9003163
2	.8414710	.6366198
3	.4541380	.3001054
4	.0000000	.0000000
5	-.2790865	-.1800643
6	-.3271947	-.2122066
7	-.2006589	-.1286166
8	.0000000	.0000000
9	.1564890	.1000352
10	.1986693	.1273240
11	.1282110	.0818469

APPENDIX II

FILTER COEFFICIENTS FOR LINEAR MEMORY FILTER

To calculate the filter coefficients for the linear memory filter, the equations of (4.9) were solved using two techniques. For the first technique, used in four design cases, the filter coefficients were obtained by linearizing the equations and using Newton's Method of Successive Approximations. For the second technique, used in all ten design cases, the filter coefficients were obtained using the Fletcher-Powell Method of optimization to minimize the equations expressed in the form of equation (4.10). These results are given in Tables 16 through 25. A tabulation of the functional minimums for the latter technique is given in Table 26 for all ten design cases.

Tables 27 and 28 present a tabulation of the results of an empirical study performed to determine the effect that the number of filter weights has on the value of the functional minimum. A discussion of the conclusions drawn from this study is given in Chapter III.

Table 16. Linear Memory Filter Coefficients for Sequence Specified to have Uniform Density and Exponential Autocorrelation

Position	Coefficients Using Linearized Equations	Coefficients Using Minimization Scheme
1	.780866	.780866
2	.493465	.493465
3	.303772	.303772
4	.185250	.185250
5	.112583	.112583
6	.068335	.068335
7	.041455	.041454
8	.025127	.025127
9	.015148	.015148
10	.014898	.014898

Table 17. Linear Memory Filter Coefficients for Sequence Specified to have Random Telegraph Signal Density and Exponential Autocorrelation

Position	Coefficients Using Linearized Equations	Coefficients Using Minimization Scheme
1	.537757	.537760
2	.615072	.615069
3	.445693	.445694
4	.287620	.287619
5	.178231	.178232
6	.109199	.109199
7	.065687	.065687
8	.042546	.042546
9	.016380	.016383
10	.032448	.032448

Table 18. Linear Memory Filter Coefficients for Sequence Specified to have Chi-square Density with One Degree of Freedom and Exponential Autocorrelation

Position	Coefficients Using Linearized Equations	Coefficients Using Minimization Scheme
1	.601227	.601243
2	.494790	.494362
3	.384052	.384839
4	.299225	.298533
5	.233021	.233655
6	.181487	.180938
7	.141283	.141825
8	.110530	.110028
9	.080828	.081016
10	.175306	.175219

Table 19. Linear Memory Filter Coefficients for Sequence Specified to have Rayleigh Density and Exponential Autocorrelation

Position	Coefficients Using Linearized Equations	Coefficients Using Minimization Scheme
1	.570300	.570294
2	.500896	.500901
3	.393494	.393490
4	.309552	.309554
5	.242587	.242580
6	.189167	.189174
7	.147390	.147385
8	.116013	.116017
9	.078203	.078198
10	.193056	.193061

Table 20. Linear Memory Filter Coefficients for Sequence Specified to have Uniform Density and Triangular Autocorrelation

Position	Coefficients Using Minimization Scheme
1	.342851
2	.333225
3	.334323
4	.335042
5	.335208
6	.334895
7	.334021
8	.333554
9.	.342611

Table 21. Linear Memory Filter Coefficients for Sequence Specified to have Random Telegraph Signal Density and Triangular Autocorrelation

Position	Coefficients Using Minimization Scheme
1	.404002
2	.354174
3	.350228
4	.352499
5	.353693
6	.352499
7	.350228
8	.354174
9	.404002

Table 22. Linear Memory Filter Coefficients for Sequence Specified to have Chi-square Density with One Degree of Freedom and Triangular Autocorrelation

Position	Coefficients Using Minimization Scheme
1	.477675
2	.364979
3	.329714
4	.315889
5	.312037
6	.315889
7	.329714
8	.364979
9	.477675

Table 23. Linear Memory Filter Coefficients for Sequence Specified to have Rayleigh Density and Triangular Autocorrelation

Position	Coefficients Using Minimization Scheme
1	.482834
2	.367610
3	.331250
4	.316992
5	.312977
6	.316995
7	.331250
8	.367610
9	.482834

Table 24. Linear Memory Filter Coefficients for Sequence Specified to have Uniform Density and Sin(x)/x Autocorrelation

Position	Coefficients Using Minimization Scheme
1	.364992
2	.468521
3	.517433
4	.467438
5	.325323
6	.137229
7	-.037811
8	-.148436
9	-.168114
10	-.105895
11	.013116
12	.195998

Table 25. Linear Memory Filter Coefficients for Sequence Specified to have Random Telegraph Signal Density and Sin(x)/x Autocorrelation

Position	Coefficients Using Minimization Scheme
1	.395707
2	.473413
3	.538830
4	.502416
5	.349128
6	.148005
7	-.041153
8	-.161509
9	-.175676
10	-.122402
11	-.023127
12	.203795

Table 26. Optimization Functional Minimums for Linear Memory Filter Design

Autocorrelation	Density	Value of Minimized Function
Exponential	Uniform	1.31×10^{-13}
Exponential	Random Telegraph Signal	8.13×10^{-13}
Exponential	Chi-square	1.39×10^{-7}
Exponential	Rayleigh	5.66×10^{-12}
Triangular	Uniform	5.88×10^{-4}
Triangular	Random Telegraph Signal	8.42×10^{-2}
Triangular	Chi-square	1.36×10^{-1}
Triangular	Rayleigh	1.56×10^{-1}
Sin(x)/x	Uniform	1.31×10^{-2}
Sin(x)/x	Random Telegraph Signal	1.26×10^{-1}

Table 27. Optimization Functional Minimum vs. the Number of Filter Weights for Triangular Autocorrelation

	n = 10		n = 20		n = 30		n = 40	
Density	$F \times 10^{-2}$	$A \times 10^{-2}$	$F \times 10^{-2}$	$A \times 10^{-2}$	$F \times 10^{-2}$	$A \times 10^{-2}$	$F \times 10^{-2}$	$A \times 10^{-2}$
Uniform	2.57	2.57	1.02	.591	.661	.379	.530*	.299*
Random Telegraph Signal	19.3	19.3	21.0	13.4	25.3	12.6	30.1	12.4

Notations:

F = Value of functional minimum for all n filter weights

A = Value of error introduced by ten specified equations

* = Convergence was not obtained

Table 28. Optimization Functional Minimum vs. the Number of Filter Weights for Sin(x)/x Autocorrelation

	n = 9		n = 18		n = 27		n = 36	
Density	$Fx10^{-2}$	$Ax10^{-2}$	$Fx10^{-2}$	$Ax10^{-2}$	$Fx10^{-2}$	$Ax10^{-2}$	$Fx10^{-2}$	$Ax10^{-2}$
Uniform	.997	.997	.340	.0201	.532	.213	.965	.416
Random Telegraph Signal	3.20	3.20	8.24	4.97	14.7	7.07	21.5	8.45

Notations:

F = Value of functional minimum for all n filter weights

A = Value of error introduced by nine specified equations

APPENDIX III

RESULTS OF STATISTICAL TESTS ON INPUT RANDOM NUMBER SEQUENCE

It is the purpose of this appendix to discuss the tests performed on two different pseudo-random sequences considered for use as input to the Predistorted Transformed Gaussian Method. The sequences were Gaussian with zero mean and unity variance. The sequences were produced by the RANDN generator of the UNIVAC 1108 standard software package called MATHPACK and the Brown-Rowland Generator (41) where multiplicative uniform values were generated using the relationship

$$X_{i+1} = 19971X_i \text{ (molulo } 2^{20}) \qquad \text{(A3.1)}$$

with twelve uniform values being summed to create each Gaussian sequence value. Each sequence was tested for its mean square property and probability density property as well as for stationarity.

Mean Square Property

The mean square property was tested using the tests on the mean and variance of Chapter V. Twenty sequences of 500 values were tested. As summarized in Table 29, both generators produced sequences having well behaved means. The sequences from the UNIVAC RANDN generator, however, proved to be unstationary with the tests for the theoretical variance failing three times. The Brown-Rowland generator gave no indication of being unstationary.

Table 29. Independent Gaussian Generator Mean Square Property Tests

Gaussian Generators	Rejections of Theoretical Mean for $\gamma = .95$	Rejections of Theoretical Variance for $\gamma = .95$
UNIVAC RANDN	0	3
Brown-Rowland	0	1

Probability Density Property

The probability density property was tested for both generators for sequences of 10,000 values and 20 subgroupings of 500 values. Both generators gave acceptable results for both the Pearson Chi-square and Kolmogorov Goodness-of-Fit Tests on the total sequence and the 20 subgroupings. The sequence of 10,000 values from the Brown-Rowland generator passed the Kolmogorov Test at better than an 80 percent level of significance while passing the Pearson Chi-square Test at a 57 percent level of significance. The sequence of 10,000 values from the UNIVAC RANDN generator passed the Kolmogorov Test at better than an 80 percent level of significance while passing the Pearson Chi-square Test at a 22 percent level of significance. Neither set of tests gave indications of the sequences being unstationary.

In conclusion, of the tests performed on the two Gaussian pseudo-random number generators, the Brown-Rowland generator is preferred because of the stationarity of the variance of the sequences.

BIBLIOGRAPHY

1. R. P. Chambers, "Random-number generation on digital computers," IEEE Spectrum, Vol. 4, No. 2, February 1967, p. 48-56.

2. T. E. Hull and A. R. Dobell, "Mixed congruential random number generators for binary machines," Journal of the Association for Computing Machinery, Vol. 11, No. 1, January 1964, p. 31-40.

3. M. D. MacLaren and G. Marsaglia, "Uniform random number generators," Journal of the Association for Computing Machinery, Vol. 12, No. 1, January 1965, p. 83-89.

4. R. P. Coveyou, "Serial correlation in the generation of pseudo-random numbers," Journal of the Association for Computing Machinery, Vol. 7, 1960, p. 72-74.

5. B. Jansson, "Autocorrelations between pseudo-random numbers," BIT, Vol. 4, 1963, p. 6-27.

6. B. Jansson, Random Number Generators, Victor Pettersons Bokindustri Aktiebolag, Stockholm, 1966.

7. M. E. Muller, "A comparison of methods for generating normal deviates on digital computers," Journal of the Association for Computing Machinery, Vol. 6, 1959, p. 376-383.

8. S. Stein and J. E. Storer, "Generating a Gaussian sample," IRE Trans. on Information Theory, Vol. IT-2, June 1956, p. 87-90.

9. M. J. Levin, "Generation of a sampled Gaussian time series having a specified correlation function," IRE Trans. on Information Theory, Vol. IT-6, December 1960, p. 545-548.

10. S. Gery, "The generation of correlated multivariate samples for Monte Carlo simulation," IEEE Spectrum, Vol. 5, No. 2, February 1968, p. 5.

11. G. H. Pakov, "Generation of a random correlated quantity on a highspeed electronic computer," Joint Publications Research Service Document No. 5784, November 8, 1960, 30 pages.

12. B. P. Lathi, An Introduction to Random Signals and Communication Theory, International Textbook Company, Scranton, Pennsylvania, 1968.

13. G. Marsaglia, "Generating exponential random variables," Annuals of Mathematical Statistics, Vol. 32, 1961, p. 891-900.

14. G. Marsaglia, "Generating discrete random variables in a computer," Communications of the ACM, Vol. 6, No. 1, January 1963, p. 37-38.

15. M. D. MacLaren, G. Marsaglia, and T. A. Bray, "A fast procedure for generating exponential random variables," Communications of the ACM, Vol. 7, No. 5, May 1964, p. 298-300.

16. G. Bankovi, "A decomposition-rejection technique for generating exponential random variables," Publ. Math. Inst. Hung. Acad. Sir., Vol. 9, 1964, p. 573-581.

17. G. B. Curtis, "Simulation of fluxional radar cross sections and other continuous random variables," Hughes Aircraft Co., Technical Internal Correspondence ID 62-12-46, 1 August 1962, 15 pages.

18. V. G. Gujar and R. J. Kavanagh, "Generation of random signals with specified probability density functions and power density spectra," IEEE Trans. on Automatic Control, Vol. AC-13, No. 6, December 1968, p. 716-719.

19. N. A. Broste, "Digital Generation of Random Sequences," IEEE Trans. on Automatic Control, Vol. AC-16, No. 2, April 1971, p. 213-214.

20. A. H. Nuttal, "Shaping correlation functions with nonlinear no-memory networks," Quarterly Progress Report, MIT Res. Lab. of Electronics, July 15, 1957, p. 68-72.

21. T. S. Edrington, "The amplitude statistics of aircraft radar echos," IEEE Trans. on Military Electronics, Vol. MIL-9, No. 1, January 1965, p. 10-16.

22. R. R. Boothe, "The Weibull distribution applied to the ground clutter backscatter coefficient," U. S. Army Missile Command Report No. RE-TR-69-15, June 1965, 25 pages.

23. E. M. Holliday, "Transformation of a set of pseudo-random numbers into a set representing any desired probability and correlation," U. S. Army Missile Command Report No. RE-TR-69-25, December 1969, 148 pages.

24. J. R. Rowland, "Improved digital simulation of continuous correlated stochastic process," Submitted to the Proc. of the IEEE for publication in the January 1972 special issue on Computers in Design.

25. R. Deutsch, Nonlinear Transformations of Random Processes, Prentice-Hall, Englewood Cliffs, N.J., 1962.

26. J. B. Thomas, An Introduction to Statistical Communication Theory, John Wiley and Sons, Inc., New York, 1969.

27. W. E. Thomson, "The response of a nonlinear system to random noise," Proc. of the IEE, Vol. 102, Pt. C, 1955, p. 46-48.

28. G. Bonet, "Nonlinear, zero-memory transformations of random signals," Ann. Telecom. Vol. 19, 1964, p. 203-220.

29. F. Baum, "The correlation function of Gaussian noise passed through non-linear devices," IEEE Trans. on Information Theory, Vol. IT-15, No. 4, July 1969, p. 448-456.

30. M. Nakamura, "Obtaining a normal sequence with prescribed covariances from an independent uniform distribution of random variables," IEEE Trans. on Systems Science and Cybernetics, Vol. SSC-4, No. 2, July 1968, p. 191.

31. D. J. Wilde and C. S. Beightler, Foundations of Optimization, Prentice-Hall, Inc., Englewood Cliffs, N.J., 1967.

32. J. S. Bendat and A. G. Piersol, Measurement and Analysis of Random Data, John Wiley and Sons, Inc., New York, 1958.

33. P. G. Hoel, Introduction to Mathematical Statistics, John Wiley and Sons, New York, Third Edition, 1962.

34. B. W. Lindgren, Statistical Theory, The MacMillan Company, New York, Second Edition, 1968.

35. D. J. Patel, Two-dimensional Estimation from a Restricted Sampling Lattice, Ph.D. Thesis, Georgia Institute of Technology, August 1971.

36. M. H. Hansen, W. N. Hurwitz, and W. G. Madow, Sample Survey Methods and Theory, Volume I--Methods and Applications, John Wiley and Sons, Inc., New York, 1953.

37. M. J. Slakter, "A comparison of the Pearson Chi-square and Kolmogorov goodness-of-fit tests with respect to validity," Journal of the American Statistical Association, Vol. 60, No. 311, September 1965, p. 854-858.

38. T. W. Anderson, An Introduction to Multivariate Statistics, John Wiley and Sons, Inc., New York, 1958.

39. A. Papoulis, Probability, Random Variables, and Stochastic Processes, McGraw-Hill Book Co., New York, 1965.

40. J. S. Bendat, Principles and Applications of Random Noise Theory, John Wiley and Sons, Inc., New York, 1958.

41. R. J. Brown Jr. and J. R. Rowland, "Autocorrelation significance in digital pseudo-random number generation," Georgia Institute of Technology, School of Electrical Engineering Technical Note, March 2, 1970.

VITA

David Allen Conner, son of William Allen Conner and Louise (Sargent) Conner, was born in Greenville, Alabama, on August 4, 1939. He was educated in the public school system of Pensacola, Florida, and graduated from Pensacola High School in 1957. He entered Auburn University in September, 1957, and received the degree of Bachelor of Science in Electrical Engineering (Communications Option) in March, 1961. In March, 1961, he entered the Graduate School of Auburn University as a candidate for the degree of Master of Science in the Department of Electrical Engineering. While in graduate school he was employed successively as a graduate teaching assistant and a full-time Instructor on the Electrical Engineering faculty of Auburn University. He received the degree of Master of Science in August, 1963. In June, 1963, he was employed by the International Business Machines Corporation--Federal Systems Division--to work on the Apollo/Saturn Program in Huntsville, Alabama. Concurrently with this employment he served as lecturer in the Department of Electrical Engineering at the University of Alabama-Huntsville campus. In January, 1966, he was employed by the Georgia Institute of Technology as a full-time instructor in the School of Electrical Engineering. Concurrently with this employment he served as an ARO-D consultant to the U. S. Army Missile Command at Redstone Arsenal, Alabama, a consultant to the American Telephone and Telegraph Company, a consultant to the legal firm of Beasley, Kahn, Rosser, and Kiser, and a consultant to the Prentice-Hall Book Company. He is a registered Professional Engineer in the State of Georgia.

Mr. Conner married Jerry Ann Macks, daughter of Ollie and Sadie (Faulk) Macks on September 4, 1959. They have three sons, Wesley Macks, eights years old; Jeffrey Allen, six years old; and Mark David, two years old.