

NAVAL POSTGRADUATE SCHOOL

Monterey, California



THESIS

THE UTILITY OF HIGHER-ORDER STATISTICS IN GAUSSIAN NOISE SUPPRESSION

by

Donald R. Green

March 2003

Thesis Advisor:
Co-Advisor:

Charles W. Therrien
Charles W. Granderson

Approved for public release; distribution is unlimited.

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.				
1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE March 2003	3. REPORT TYPE AND DATES COVERED Master's Thesis	
4. TITLE AND SUBTITLE: The Utility of Higher-Order Statistics in Gaussian Noise Suppression			5. FUNDING NUMBERS	
6. AUTHOR(S) Green, Donald R.				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000			8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A			10. SPONSORING / MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.				
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.			12b. DISTRIBUTION CODE	
13. ABSTRACT (maximum 200 words) The properties of higher-order statistics are becoming more and more thoroughly studied in the field of signals processing. One property of great interest is the fact that the cumulants of Gaussian signals disappear entirely at higher orders. Because many noise and interference signals have Gaussian distributions, this property offers the possibility that higher-order statistics may be useful in signal recovery or interference mitigation, which would be of great advantage in military communications, intelligence, or surveillance systems. This thesis examines some of the theory behind higher-order statistics, and discusses the estimation of third-order cumulant values for several random variable distributions. After a minimum sample size has been determined, the study progresses to the frequency domain for an examination of the bispectra of the distributions. The thesis then examines the effects on the bispectrum of combining Gaussian and non-Gaussian signals, and concludes with recommendations for implementing signal processing systems which utilize higher-order statistics.				
14. SUBJECT TERMS Higher-order statistics, statistical signal processing			15. NUMBER OF PAGES 139	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL	

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

**THE UTILITY OF HIGHER-ORDER STATISTICS IN GAUSSIAN
NOISE SUPPRESSION**

Donald R. Green
United States Department of Defense
B.E.E., Georgia Institute of Technology

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

**NAVAL POSTGRADUATE SCHOOL
March 2003**

Author: Donald Richard Green

Approved by: Charles W. Therrien
Thesis Advisor

Charles W. Granderson
Co-Advisor

John Powers
Chairman
Department of Electrical and Computer Engineering

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

The properties of higher-order statistics are becoming more and more thoroughly studied in the field of signal processing. One property of great interest is the fact that the cumulants of Gaussian signals disappear entirely at higher orders. Because many noise and interference signals have Gaussian distributions, this property offers the possibility that higher-order statistics may be useful in signal recovery or interference mitigation, which would be of great advantage in military communications, intelligence, or surveillance systems. This thesis examines some of the theory behind higher-order statistics, and discusses the estimation of third-order cumulant values for several random variable distributions. After a minimum sample size has been determined, the study progresses to the frequency domain for an examination of the bispectra of the distributions. The thesis then explores the bispectra of non-Gaussian signals in the presence of Gaussian noise, and concludes with recommendations for implementing signal processing systems which utilize higher-order statistics.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

I. INTRODUCTION.....	1
A. THE NATURE AND UTILITY OF HIGHER-ORDER STATISTICS.....	1
1. Introduction.....	1
2. A Brief Introduction to the Theory of Higher-Order Statistics.....	2
a. Moments.....	2
b. Cumulants.....	3
c. Spectra and Polyspectra.....	4
3. Experiment Objectives.....	5
II. NECESSARY SIGNAL SAMPLE LENGTH WHEN ESTIMATING THIRD-ORDER CUMULANTS	7
A. THEORY.....	7
1. Introduction.....	7
2. Theory.....	8
a. Normal (Gaussian) distribution.....	9
b. Uniform distribution.....	9
c. Mean-shifted exponential distribution.....	10
d. Mean-shifted Rayleigh distribution.....	10
3. Results.....	10
B. EXPERIMENTAL RESULTS.....	11
1. Procedure.....	11
2. Results.....	12
3. Conclusions.....	13
III. BISPECTRA OF DIFFERENT SIGNAL TYPES.....	15
A. THEORY.....	15
1. Introduction.....	15
2. MATLAB Bispectrum Estimation Functions.....	15
3. Theoretical Predictions.....	16
B. PROCEDURE.....	20
C. EXPERIMENTAL RESULTS AND CONCLUSIONS.....	23
IV. NONGAUSSIAN SIGNALS IN THE PRESENCE OF GAUSSIAN NOISE.....	29
A. THEORY.....	29

1. Introduction.....	29
2. Theoretical Predictions.....	29
B. EXPERIMENTAL RESULTS.....	29
1. Procedure.....	29
2. Data.....	31
3. Conclusions.....	32
V. CONCLUSION.....	35
A. OVERALL CONCLUSIONS.....	35
B. STEPS FOR FURTHER STUDY.....	36
APPENDIX A. MATLAB RESULTS FROM CHAPTER II.....	39
A. Statistic: Mean.....	39
B. Statistic: Variance.....	42
C. Statistic: Third-order cumulant estimate.....	45
APPENDIX B. PLOTS FOR CHAPTER III.....	49
A. Lag argument = 32, DFT length = 128, white signal.....	49
B. Lag argument = 64, DFT length = 256, white signal.....	52
C. Lag argument = 128, DFT length = 512, white signal.....	55
D. Lag argument = 256, DFT length = 1024, white signal.....	58
E. Lag argument = 512, DFT length = 2048, white signal.....	61
F. Lag argument = 256, DFT length = 2048, white signal.....	63
G. Lag argument = 32, DFT length = 128, colored signal ($\beta = 0.25$).....	67
H. Lag argument = 64, DFT length = 256, colored signal ($\beta = 0.25$).....	70
I. Lag argument = 128, DFT length = 512, colored signal ($\beta = 0.25$).....	73
J. Lag argument = 256, DFT length = 1024, colored signal ($\beta = 0.25$).....	76
K. Lag argument = 512, DFT length = 2048, colored signal ($\beta = 0.25$).....	79
L. Lag argument = 1024, DFT length = 4096, colored signal ($\beta = 0.25$).....	82
M. Lag argument = 128, DFT length = 1024, colored signal ($\beta = 0.25$).....	85
N. Lag argument = 128, DFT length = 2048, colored signal ($\beta = 0.25$).....	88
APPENDIX C. PLOTS FOR CHAPTER IV.....	91
A. Mean-shifted exponential distribution, lag argument = 128, colored signal.....	91
B. Mean-shifted Rayleigh distribution, lag argument = 128, colored signal.....	95
C. Mean-shifted exponential distribution, lag argument = 256, colored signal.....	98

D. Mean-shifted Rayleigh distribution, lag argument = 256, colored signal.....	102
APPENDIX D. GENERAL EXPRESSIONS FOR THIRD-ORDER CUMULANTS...	107
APPENDIX E. THEORETICAL THIRD-ORDER CUMULANT EXPRESSIONS FOR DISTRIBUTIONS USED IN THIS THESIS.....	109
A. Normal (Gaussian) distribution.....	109
B. Uniform distribution.....	109
C. Mean-shifted exponential distribution.....	110
D. Mean-shifted Rayleigh distribution.....	110
APPENDIX F. FLOW DIAGRAM OF MATLAB SIMULATIONS.....	113
APPENDIX G. MATLAB CODE.....	115
A. MATLAB script from Chapter II.....	115
B. MATLAB FUNCTION rpiid_var.....	116
C. MATLAB script from Chapter III.....	120
LIST OF REFERENCES.....	123
INITIAL DISTRIBUTION LIST.....	125

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENT

The author would like to gratefully acknowledge the guidance and patience of his two advisors, Charles Granderson at the Department of Defense and Professor Charles Therrien at the U.S. Naval Postgraduate School.

He would also like to acknowledge the feedback and direction provided by Professor Ralph Hippenstiel, late of the U.S. Naval Postgraduate School, currently at the University of Texas, Tyler.

Finally, the author would like to thank his wife Tina, for her unfailing encouragement, patience, and support.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

The use of higher-order statistics provides insight into signals which is not always available at lower orders. Additionally, Gaussian-distributed signals have the interesting characteristic of disappearing at higher orders. Because so much of the noise and interference environment is Gaussian-distributed, higher-order statistics thus offer the promise of an additional method of noise reduction and interference mitigation. As communications signals become more and more complex, any additional ability to reduce the effects of noise and interference will have a profound impact on communications, surveillance, and intelligence systems.

A. THE NATURE AND UTILITY OF HIGHER-ORDER STATISTICS

1. Introduction

Traditionally, most of the procedures performed in statistical signal processing applications have been first- and second-order operations, i.e., the calculation of a signal's first- and second-order moments. Whereas the first-order moment of a signal is simply its mean, its second-order moment is the correlation of the signal with a time-delayed ("lagged") copy of itself. This moment is also referred to as the signal's *autocorrelation*. The Fourier transform of the autocorrelation, in turn, produces the power spectral density (PSD) so familiar to signal engineers. By its nature the PSD is a second-order quantity.

Recently, however, significant interest has been paid to *higher-order* statistical operations. Instead of merely correlating a signal with one time-delayed copy of itself, it is sometimes useful to correlate it with two or more such copies, at varying lags. These multiple-lag processes may also be operated on by Fourier transforms to produce, for example, three- or four-dimensional spectral displays (known as *bispectra* and *trispectra*, respectively, and more generally as *polyspectra*) in a manner similar to that of the traditional PSD.

Such higher-order techniques have typically been used less frequently than their second-order cousins in the signals world, partly because of the significantly greater amount of processing power required to compute them. However, recently such techniques have been the subject of increasing study and have found applications in such

disciplines as image processing, biological monitoring, and the modeling of wave phenomena [Ref.1].

The reason for such interest in higher-order statistics is that they provide information not always available through second-order techniques. Some such information includes the detection of nonlinearity; the preservation of phase relationships, and the measurement of a signal's deviation from Gaussianity. This last property is the result of the fact that the third- and fourth-order cumulants of Gaussian signals are zero. This, in turn, suggests that the third- or fourth-order cumulant of a signal can be used to quantify just how "non-Gaussian" a signal is and further suggests that higher-order statistical properties could be used to generate a filtering algorithm to remove unwanted Gaussian elements (noise or interference) from a signal. The feasibility of such a filter is the impetus behind this thesis.

This thesis initially discusses some of the theory behind higher-order statistics, particularly as it applies to filtering away Gaussian signal components. It then details the steps taken towards constructing such a filter. These steps are to determine the signal sample length necessary to provide accurate cumulant estimates for various signal types (Gaussian and non-Gaussian), to examine the performance of bispectrum estimation for Gaussian and non-Gaussian signals, and to examine the ability to distinguish a non-Gaussian signal from additive Gaussian noise.

2. A Brief Introduction to the Theory of Higher-Order Statistics

a. Moments

The n^{th} -order moment of a real, stationary sequence $X(t)$ with PDF $f_x(x)$ is given by $E\{X^n\} = \int_{-\infty}^{\infty} x^n f_x(x) dx$ for $n = 1, 2, 3, \dots$, where $E\{\cdot\}$ denotes expectation and where x , as per probability convention, denotes a specific realization of the random variable X . For $n = 1$ (i.e., first-order), the moment is simply the expected value of X and is commonly known as the *mean*, denoted by $m_x = E\{X\}$. For $n = 2$ (i.e., second-order), the moment is the autocorrelation, which indicates how well the sequence correlates with a time-lagged version of itself.

Moments of arbitrary order may be calculated via a distribution's *moment-*

generating function $M_x(\omega) = E\{e^{x\omega}\} = \int_{-\infty}^{\infty} e^{x\omega} f_x(x) dx$. The n^{th} -order moment of a signal is generated by differentiating $M_x(\omega)$ n times, and setting $\omega = 0$. Thus the mean of a signal may be calculated as $m_x = E\{X\} = M_x'(0)$, where the single prime mark indicates the first differential of $M_x(\omega)$. Moments of higher order may be generated similarly [Ref. 2].

Because the concept of signal lag is frequently of interest in the signals world, however, it is frequently used to define second- and higher-order moments. Here, lags will be denoted by τ_n and the n^{th} -order moment written as

$$m_n^x(\tau_1, \tau_2, \dots, \tau_{n-1}) = E\{X(t)X(t+\tau_1)\dots X(t+\tau_{n-1})\}. \quad (1)$$

Such notation makes it more obvious, for example, that the second-order moment is the autocorrelation when it is written as $m_2^x(\tau_1) = E\{X(t)X(t+\tau_1)\}$.

b. Cumulants

Although moments of order greater than two are indeed higher-order statistics, frequently *cumulants*, rather than moments, are used. Cumulants may be calculated in a manner analogous to that of moments; the *cumulant-generating function* is produced by taking the natural logarithm of the moment-generating function: $C_x(\omega) = \ln(M_x(\omega))$. Differentiating $C_x(\omega)$ n times and setting $\omega = 0$, as with the moment-generating function, produces a signal's n^{th} -order cumulant.

However, if the study is limited to order four and below (as it is here), an equally valid but more intuitive representation (for purposes of this thesis) is

$$c_n^x(\tau_1, \tau_2, \dots, \tau_{n-1}) = m_n^x(\tau_1, \tau_2, \dots, \tau_{n-1}) - m_n^G(\tau_1, \tau_2, \dots, \tau_{n-1}) \quad (2)$$

where $c_n^x(\tau_1, \tau_2, \dots, \tau_{n-1})$ is the n^{th} -order cumulant of the random variable x ; $m_n^x(\tau_1, \tau_2, \dots, \tau_{n-1})$ is the n^{th} -order moment of the random variable x ; and $m_n^G(\tau_1, \tau_2, \dots, \tau_{n-1})$ is the n^{th} -order moment of a Gaussian random variable that has the same first- and second-order moments as x . This notation makes it clear that cumulants may be used to measure the Gaussianity, or lack thereof, of a random variable. It is also a convenient way of illustrating a significant property of cumulants; namely that the cumulant of a Gaussian-distributed signal is identically zero. (Note that even though the alternative representa-

tion above only applies to third- and fourth-order statistics, the n^{th} -order cumulant of a Gaussian-distributed signal is zero for *any* value of n greater than two.) This is one of the major justifications for using cumulants rather than moments: if it is desired to measure deviation from Gaussianity, then third- and fourth-order cumulants are ideal. Another justification is that the higher-order cumulants of white noise are multidimensional impulse functions, which is not necessarily the case with higher-order moments. However, it should be pointed out that for zero-mean processes, such as those used in this study, the third-order moment and third-order cumulant are equal, so this last justification becomes somewhat moot [Ref. 3, 4].

Before concluding this introduction of moments and cumulants, two additional terms should be mentioned. As discussed above, the first-order moment is the mean of a random variable and the second-order moment is its variance. We may also use higher-order statistics for the random variables, known as *skewness* and *kurtosis*. Skewness is the third-order cumulant with $\tau_1 = \tau_2 = 0$, while kurtosis is the fourth-order cumulant with $\tau_1 = \tau_2 = \tau_3 = 0$. To use a physical analogy, let the PDF represent a mass distribution function. In this case, the mean is simply the location of the center of gravity of the mass; the variance is a measure of how the mass is dispersed from its center of gravity; the skewness is a measure of how symmetric the mass is about its mean value (i.e., whether it is "skewed" to the right or the left of center—if it is symmetric about the mean then its skewness is zero); and the kurtosis is a measure of the flatness or excess about the mean. These terms are particularly significant because some higher-order signal processing applications require zero skewness, or zero kurtosis, in order to be effective. In the case of this thesis, however, the only requirement is a mean of zero.

c. Spectra and Polyspectra

A few introductory comments need to be made about polyspectra, the higher-order analogs of a signal's power spectral density. As with second-order statistics, the calculation of a signal's higher-order spectrum frequently allows insight into the nature of the signal that may not be possible in the time domain. If nothing else, it can provide a more intuitive, visual signal representation.

When the Fourier transform is taken of a second-order cumulant (the

covariance), the result is the power spectrum. Likewise, when the Fourier transform of the third- and fourth-order cumulants are generated, the results are the bispectrum and trispectrum, respectively. The bispectrum is a function plotted versus two frequency axes and is frequently written as $C^x_3(\omega_1, \omega_2)$. (The presence of two dependent axes results in the somewhat confusing term "bispectrum" for $n=3$.) In a similar manner, the trispectrum is a function of three variables denoted as $C^x_4(\omega_1, \omega_2, \omega_3)$ and usually plotted as a three-dimensional contour with one axis variable held constant. Both the bispectrum and trispectrum are complex-valued functions, so magnitude and phase must be plotted separately.

3. Experiment Objectives

With the above theory in mind, the purpose of this thesis was to study third-order cumulants and bispectra. Specifically, the intent was to examine the effects of various signal parameters, as well as of the various parameters involved in bispectral estimation. It was further desired to study the effects of filtering (to change a signal from white to colored), and of adding various levels of noise to the environment.

These different aspects of cumulant and bispectral estimation, would be very useful in any attempt to develop a practical signal processing application using higher-order statistics. One example might be that of an adaptive filter; because most noise, and many interferers, seen in the signal world are Gaussian, the ability to suppress Gaussian signal elements with a limited effect on a non-Gaussian information-bearing signal would be of obvious benefit. To construct such a filter, however, would also require an appreciation not only of the parameters required to generate higher-order statistics, but also of the effects of examining the higher-order statistics of various non-Gaussian signal types. This potential application is one motivation for the study performed here.

The different factors discussed here were investigated using MATLAB and its Higher-Order Spectral Analysis (HOSA) toolbox [Ref. 6]. MATLAB was used to generate sample data sets, to calculate estimates of the data's third-order cumulants and bispectra, and to simulate operations on the data.

Chapter II describes the generation of the initial datasets, and the estimates of their third-order cumulants. The results of the estimations were compared to their theo-

retical results, and the difference between the two was used to determine the signal lengths necessary to produce consistent third-order cumulant estimates.

Chapter III describes the next step in the experiment. The study of bispectra, rather than simply of third-order cumulants, introduces additional considerations, such as segmentation and averaging of the data. This Chapter records the effects of varying different parameters, such as the number of lags computed and the length of the direct Fourier transform (DFT) used, on the bispectra.

Chapter IV explores the bispectral estimation in a slightly more realistic environment, that of non-Gaussian signals in additive white Gaussian noise.

Chapter V offers a summary and conclusions, and projects future steps towards developing signal processing applications using higher-order statistics.

Appendix A is a catalog of the plots produced by the procedure of Chapter II, as the mean, variance, and third-order cumulants were examined while the experimental ensemble length was increased.

Appendix B is a catalog of the plots produced by the procedure of Chapter III, as the bispectra of white and colored signals were generated. The plots in this Chapter show the effects of varying the lag argument and DFT length in the MATLAB *bispeci* function.

Appendix C is a catalog of the plots produced by the procedure of Chapter IV, as the bispectra of colored signals plus additive white Gaussian noise were generated.

Appendix D is a derivation of a general expression for the third-order cumulant of a zero-mean, independent and identically-distributed dataset.

Appendix E contains the derivations of the theoretical third-order cumulants for the four distributions examined here.

Appendix F is a flow diagram for the MATLAB code used in the procedure of Chapter II.

Appendix G contains the actual MATLAB code used here. It includes simple scripts used to generate the data for Chapters II and III, as well as a function called *rpiid_var* which was used to create random data with various distributions and variances.

II. NECESSARY SIGNAL SAMPLE LENGTH WHEN ESTIMATING THIRD-ORDER CUMULANTS

As mentioned in the introduction, a recurring limitation on the use of higher-order statistics in signal processing has been the significant computational burden involved. Therefore, an early step in this study was to determine the size of signal samples that would be required. Specifically, the purpose of this first step was to determine the minimum sample length that would produce an accurate characterization of multiple types of signals in a noise-free environment. The signal types (as determined by their probability distributions) examined were Gaussian (the normal distribution), uniform, exponential, and Rayleigh. The reasoning behind these choices is given below.

The theoretical mean, variance, and third-order cumulant were calculated for each of the distributions. For the distributions examined here, these results were straightforward—either a constant value, or a peak at one specific value. After the theoretical values were calculated, MATLAB was used to simulate the sequences and to calculate their bispectra with the HOSA toolbox. These results were then compared to the theoretical predictions, and the process was repeated for various sample lengths. As might be expected, the calculated mean, variance, and third-order cumulants more closely approximated their theoretical values as the sample lengths were increased. The result of this step was a minimum signal sample length which provides a quantifiably accurate estimate for each of the distributions under study.

A. THEORY

1. Introduction

As mentioned above, the four signal distribution types examined in this experiment were the Gaussian, uniform, exponential, and Rayleigh distributions. The Gaussian distribution was examined as a baseline; its prevalence in the world of signals (especially in light of the Central Limit Theorem) makes its inclusion more or less mandatory. In addition, the Gaussian distribution is particularly interesting in the field of higher-order statistics due to the fact, discussed above, that its third- and higher-order cumulants are identically zero. This characteristic, of course, is the impetus behind this thesis; if most

noise is Gaussian, and Gaussian signals disappear at higher orders, then perhaps higher-order statistics are a good candidate for the development of noise-reduction techniques.

The second distribution examined was the zero-centered uniform distribution. As with the Gaussian sequence, the uniform sequence was studied here more as a control or baseline rather than in the hope that it would provide some unique noise resistance in higher orders. This is due to the fact that, as with the Gaussian distribution, the zero-mean uniform distribution has a third-order cumulant of zero. These third-order cumulant values of zero arise from the symmetry of the Gaussian and uniform distributions, which is discussed in more detail below.

The third distribution examined was the exponential distribution. As an asymmetric distribution, the exponential distribution has a nonzero third-order cumulant and could potentially prove to be valuable as an information-bearing signal in the presence of Gaussian noise, if such noise can be reduced via higher-order statistics. The exponential distribution examined here, however, is not the pure exponential distribution as described in textbooks but is rather a "mean-shifted" version. Essentially it is the pure exponential distribution from which the mean has been subtracted. This has the effect, of course, of shifting the distribution to center it around zero.

The fourth distribution examined, the Rayleigh distribution, was also a "mean-shifted" version. As with the exponential distribution, the Rayleigh distribution is asymmetric and hence in this study is less of a control signal and more of a potentially useful information-bearing signal. As with the exponential signal, the "mean-shifted" version was generated simply by subtracting the mean on a sample-by-sample basis so that the resulting distribution was centered about zero.

2. Theory

The third-order cumulant of a sequence of samples of a zero-mean process is equivalent to its third-order moment [Ref. 1]. Additionally, if the samples are independent and identically distributed ("i.i.d."), the third-order cumulant is equal to the (scalar) value of the third-order moment at only one point, that at which both lags are zero (see Appendix D). Two additional useful properties are that the third-order cumulant of a symmetric distribution is zero, and that the cumulant of two statistically independent

random processes is equal to the sum of the cumulants of the individual processes [Ref. 4].

The Gaussian and uniform distributions are indeed symmetric, so their third-order cumulants are zero. The exponential and Rayleigh distributions are asymmetric, and therefore have nonzero third-order cumulants. However, the pure exponential and Rayleigh distributions have nonzero mean and, hence, must be "mean shifted" in order to take advantage of the properties listed above. Once the mean has been removed, these new sequences contain no more signal energy than the Gaussian- and uniform-distributed sequences, but still yield nonzero third-order cumulants.

The probability density functions (PDFs) and theoretical third-order cumulant values are given here for each of the distributions. Most of the functions listed here may be found in any of a number of standard statistics texts; e.g., see [Ref. 5]. Some of the functions have been derived from others; this work is catalogued in Appendix E (as well as in the MATLAB code of Appendix G).

a. Normal (Gaussian) distribution

The PDF of a Gaussian distribution is given by

$$f_x(x) = \frac{1}{\sigma \sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad \text{for } -\infty < x < \infty \quad (3)$$

where μ is the mean and σ is the standard deviation of the random variable. The third-order cumulant of such a distribution is zero, because the distribution is symmetric.

b. Uniform distribution

The PDF of a uniform distribution is given by

$$f_{(x)}(x) = \begin{cases} \frac{1}{b-a} & \text{for } a < x < b \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

where a and b are the lower and upper range limits, respectively. The third-order cumulant of this distribution is zero as well.

c. Mean-shifted exponential distribution

The PDF of the "mean-shifted" exponential distribution is given by

$$f_{(x)}(x) = \begin{cases} \lambda e^{-\lambda(x+1)} & \text{for } x \geq \frac{-1}{\lambda} \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

where λ is the reciprocal of the distribution's scale parameter [Ref. 5: p. 59]. For this distribution, the third-order cumulant is $2/\lambda^3$. Setting the variance of a variable with such a distribution to four produces a theoretical third-order cumulant of 16.

d. Mean-shifted Rayleigh distribution

The PDF of the "mean-shifted" Rayleigh distribution is given by

$$f_{(x)}(x) = \begin{cases} \frac{\left(x+b\sqrt{\frac{\pi}{2}}\right)}{b^2} e^{-\frac{\left(x+b\sqrt{\frac{\pi}{2}}\right)^2}{2b^2}} & \text{for } x \geq -b\sqrt{\frac{\pi}{2}} \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

where b is the distribution scale parameter [Ref. 5: p. 134]. The third-order cumulant of this distribution is $b^3(\pi-3)\sqrt{(\pi/2)}$, and setting the variance to four forces the third-order cumulant to evaluate to a (truncated) value of 5.04739.

The Gaussian and uniform distributions are the standard ones, taken in this case from [Ref. 5]. The mean-shifted Rayleigh and mean-shifted exponential distributions are adapted from the standard Rayleigh and exponential distributions, from the same reference. The calculations for the third-order cumulant values are given in Appendix E.

3. Results

The values to be predicted in this case were the sequence mean, variance, and third-order cumulant, for each of the four distributions. Since the sequences consist of independent, identically-distributed random variables, the second- and third-order cumulants have the form $c_2^x(\tau) = \sigma^2 \delta(\tau)$ and $c_3^x(\tau_1, \tau_2) = \alpha \delta(\tau_1) \delta(\tau_2)$, respectively (i.e., they are zero except for $\tau_i = 0$). The theoretical values are summarized in Table 1, and will be compared to the experimental results in the following Chapters.

<i>Statistic</i>	<i>Gaussian</i>	<i>Uniform</i>	<i>Mean-shifted exponential</i>	<i>Mean-shifted Rayleigh</i>
<i>Mean (μ)</i>	0	0	0	0
<i>Variance (σ^2)</i>	4	4	4	4
<i>Third-order cumulant (α)</i>	0	0	16	5.04739...

Table 1. Theoretical Statistical Values for Various Distribution Types

B. EXPERIMENTAL RESULTS

1. Procedure

For each of the four distributions under study, a MATLAB script (see Appendix G) was executed with the sequence (realization) length set to 512. This length was selected because prior experimentation had suggested that 512 points were adequate to consistently produce third-order cumulant estimates with a variance of unity or less. The script calls a MATLAB function *rpiid_var* (included in Appendix G), which is an adaptation of the HOSA *rpiid* program, to generate the random variables. The *rpiid_var* function has the added functionality of allowing a user-specified variance and of providing additional sequence types (notably the mean-shifted exponential and mean-shifted Rayleigh distributions).

Each of the four ensembles produced consisted of 8192 realizations of the 512-point sequences. The mean, variance, and third-order cumulant were first estimated for one realization (one 512-point sequence) out of each ensemble, and the three values estimated were each plotted as a data point. The procedure was then repeated for two realizations, for four realizations, and so on, as the number of sequences examined were doubled with each iteration. [The flow of these simulations is expanded for clarity in Appendix F.] Thus, the estimates of the mean, variance, and third-order cumulant for each distribution could be plotted as a function of ensemble length.

Each of these plots is reproduced in Appendix A. The plots are grouped together by relevant statistic: for example, Figures 9 through 12 represent the mean calculations for ensembles of Gaussian, uniform, mean-shifted exponential, and mean-shifted Rayleigh distributions, respectively. Similarly, Figures 13 through 17 represent the calculated variances, and Figures 17 through 20 represent the calculated third-order cumulant

estimates. The horizontal line in each plot is the theoretical value of the statistic in question, for comparative purposes.

Observations and intermediate conclusions follow each set of four plots. Overall conclusions are in the section following the data; general results are discussed below.

2. Results

The plots suggest, unsurprisingly, that the estimates of ensemble mean, variance, and third-order cumulant become more accurate as the ensemble length is increased. One typical plot is reproduced below as Figure 1; it and the others are contained in Appendix A. Figure 1 shows that as the ensemble length is increased from 1 to 1024 sequences, the estimate of the ensemble third-order cumulant approaches its theoretical value as a decaying oscillation. (The theoretical value, zero, is represented in the plot below as a line of dots, spaced logarithmically.)

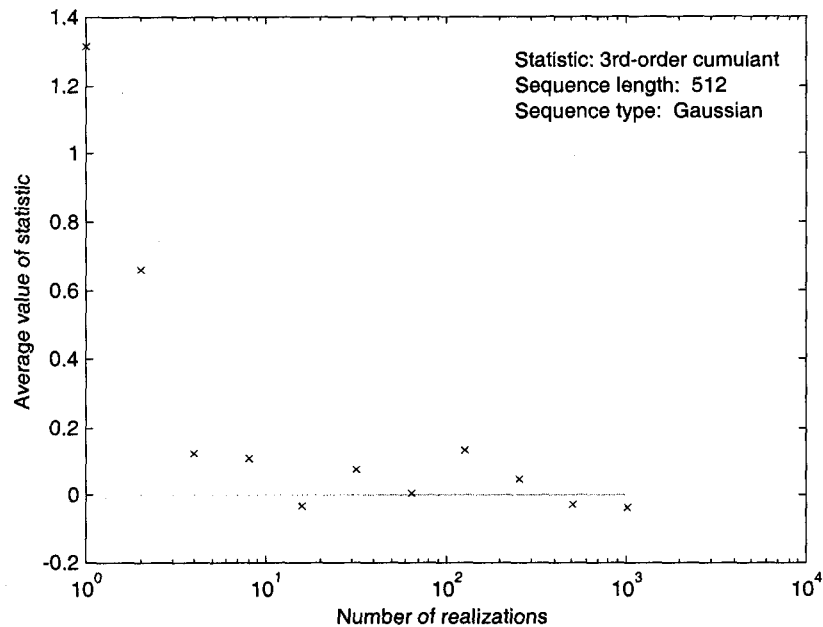


Figure 1. Third-order cumulant estimates. Estimates were calculated for Gaussian ensembles of varying lengths

Table 2 shows the minimum ensemble length, by distribution type, to attain five percent or less experimental error for each of the three statistics. Note that the mean of all four sequences, and the third-order cumulant estimate of the Gaussian and uniform

sequences, is zero; in these cases the number given is the minimum number of realizations required to draw to within an absolute error of 0.05.

<i>Statistic</i>	<i>Gaussian</i>	<i>Uniform</i>	<i>Mean-shifted exponential</i>	<i>Mean-shifted Rayleigh</i>
<i>Mean (μ)</i>	8	2	1	1
<i>Variance (σ^2)</i>	4	2	16	4
<i>Third-order cumulant (α)</i>	256	128	64	128

Table 2 Minimum ensemble sizes required to produce five percent or less error, or to within an absolute error of 0.05

Table 3 shows the experimental error for each statistic (for each distribution) when the ensemble used comprises 256 realizations of 512 data points each. As in Table 2, the error values where the expected value is zero are absolute magnitude errors, and not percent errors.

<i>Statistic</i>	<i>Gaussian</i>	<i>Uniform</i>	<i>Mean-shifted exponential</i>	<i>Mean-shifted Rayleigh</i>
<i>Mean (μ)</i>	0.006	0.008	0.033×10^{-15}	1.0×10^{-16}
<i>Variance (σ^2)</i>	0.625 %	0.615 %	1.25 %	0.625 %
<i>Third-order cumulant (α)</i>	0.05	0.05	0.7813 %	1.98 %

Table 3 Error in statistic estimates for ensemble size of 256 realizations, 512 points each (shaded cells are absolute error values, nonshaded cells are percent error values)

3. Conclusions

The error in the ensemble mean calculations remained small regardless of the ensemble size. The ensemble variance estimates showed a good deal more error at shorter ensemble sizes, and seemed to suggest that 256 realizations, at a minimum, are necessary to obtain estimates which are consistently within an experimental error of one percent.

The ensemble third-order cumulant estimates exhibited similar results, although the error values involved were slightly larger. While one percent error was never realized for all three statistics of the mean-shifted Rayleigh distribution, the errors did reach less than five percent at 128 realizations.

Overall, the data suggest that a minimum ensemble size of 256 realizations, of

512 points each should be sufficient for cumulant estimation. This ensemble size corresponds to a total data set length of 131072 points.

In the second-order domain, it is frequently the case that a signal's power spectral density is of more interest than its correlation function. Similarly, it may be more useful to examine a signal's polyspectra than to restrict examination to its moments and cumulants. For this reason, Chapter III examines the effects of various parameters on the generation of the bispectra for these distributions of interest.

III. BISPECTRA OF DIFFERENT SIGNAL TYPES

A. THEORY

1. Introduction

Because of the insight it provides, it is often desirable to examine a signal in the frequency domain. As discussed in Chapter I, an estimate of a signal's autocorrelation function (a second-order statistic) may be used to generate a power spectral density display. Likewise, the estimate of a signal's third-order cumulant may be used to generate its bispectrum. Unfortunately, generation of such an estimate requires more careful consideration than simply generating a signal's third-order moment or cumulant.

MATLAB, through its HOSA toolbox, provides a number of convenient tools with which to estimate the bispectrum of a signal. These functions require the user to specify a number of parameters, whose relationships to the final result are not always completely predictable or intuitive. This Chapter, then, details experimentation with one of the functions and compares the results produced with the results predicted by theory.

2. MATLAB Bispectrum Estimation Functions

The MATLAB HOSA toolbox provides several functions to generate bispectral estimates [Ref. 6]. The two conventional (non-parametric) functions it provides are *bispecd* and *bispeci*. The former uses a "direct" method, performing discrete Fourier transforms (DFTs) before smoothing the result in the frequency domain. The latter, used here, operates via an "indirect" method and smooths the third-order cumulant estimates in the time domain with a lag window, before performing the DFT. [The HOSA toolbox also provides a function, *bispect*, to compute the bispectrum using a parametric method, as well as additional functions to produce cross-bispectral estimates. This parametric function was not used here.]

The *bispeci* function accepts several parameters. The user provides the number of lags to be computed, the size of each data segment, the overlap between segments, the length of the DFT to be used, the type of lag window to be applied to the data, and whether a biased or an unbiased estimate is to be computed.

If the dataset provided is a matrix, *bispeci* assumes that each column is an inde-

pendent realization of data. In this case, the data segment size is forced to be the row dimension, and the overlap between segments is set to zero. The window used by *bispeci* defaults to a Parzen window [Refs. 6, 7] but can be overridden to use a unit hexagonal window.

3. Theoretical Predictions

The bispectra of the Gaussian- and uniform-distributed signal ensembles are theoretically flat in both dimensions and have a magnitude of zero. The bispectra of the mean-shifted exponential and mean-shifted Rayleigh distributions are also theoretically flat, but have nonzero magnitudes; the magnitude in each case is the scalar value of the signal's third order cumulant α . Thus, given a variance of four in the underlying random process, the bispectrum of the mean-shifted exponential ensemble should be a flat plane with a magnitude of 16. Likewise, the bispectrum of the mean-shifted Rayleigh ensemble should be a flat plane with a magnitude of approximately 5.04. (See Table 1 in Chapter II for the theoretical values.)

If the signal is processed by a linear filter before its bispectrum is calculated, the filter will serve to shape the bispectrum. The "floor" value of such a signal should still be its third-order cumulant value: zero for the Gaussian and uniform distributions, 16 for the exponential distribution, and approximately 5.04 for the Rayleigh distribution. However, for the exponential and Rayleigh distributions, the shaping will cause the bispectrum to no longer be flat; i.e., it will go from being a white to a colored signal. The theoretical bispectrum of this shaped signal may be expressed as a function of the original sequence and of the transfer function of the shaping filter.

When a signal is transformed in some fashion, its polyspectra are obviously transformed as well. If the signal is stationary and has zero mean, and the transformation is linear and time-invariant (as is the case here), then the bispectrum of the output is related to the bispectrum of the input by the expression

$$B_y(\omega_1, \omega_2) = H^*(e^{j(\omega_1 + \omega_2)}) H(e^{j\omega_1}) H(e^{j\omega_2}) B_x(\omega_1, \omega_2), \quad (7)$$

where $B_x(\omega_1, \omega_2)$ is the bispectrum of the input sequence, $B_y(\omega_1, \omega_2)$ is the bispectrum of the output (filtered) sequence, and $H(e^{j\omega})$ is the filter transfer function [Ref. 3, p. 265]. From this expression, the bispectrum of a filtered signal can be predicted, if the filter

transfer function and the bispectrum of the original signal are known.

The filter used in this study was a simple one-pole, lowpass filter (LPF), described by the difference equation $y[n] = \beta y[n-1] + x[n]$, where $x[n]$ is the input sequence, $y[n]$ is the output sequence, and β is a feedback parameter ($0 < \beta < 1$). The frequency response of this filter is given by [Ref. 8]

$$H(e^{j\omega}) = \frac{1}{1 - \beta e^{-j\omega}} \quad (8)$$

Therefore, substituting this expression into Equation (7) yields

$$B_y(\omega_1, \omega_2) = \frac{1}{1 - \beta e^{j(\omega_1 + \omega_2)}} \cdot \frac{1}{1 - \beta e^{-j\omega_1}} \cdot \frac{1}{1 - \beta e^{-j\omega_2}} \cdot B_x(\omega_1, \omega_2) \quad (9)$$

In this case, the bispectrum of the input sequence $B_x(\omega_1, \omega_2)$ is a real scalar value α : zero for the Gaussian and uniform cases, 16 for the mean-shifted exponential case (given an initial signal variance of four), and approximately 5.04 for the mean-shifted Rayleigh case (again, given an initial signal variance of four). Thus, producing the theoretical bispectrum of the output sequence $B_y(\omega_1, \omega_2)$ is simply a matter of having MATLAB calculate the appropriate result for each value of ω_1 and ω_2 , and multiplying by the appropriate scalar value.

For purposes of this study, the filter parameter β was initially arbitrarily chosen to be 0.25, although other values were examined as well. This value was provided to a short MATLAB script, along with the scalar value $B_x(\omega_1, \omega_2)$. The script then produced the theoretical bispectra of white noise signals that had been shaped by the LPF.

For example, Figures 2 through 5 show the magnitude and phase of the theoretical bispectrum of a mean-shifted exponentially-distributed signal, which has been processed by a filter described in Equation (8) with a parameter β of 0.25. The plots show that the bispectrum is indeed shaped by the filter and is no longer flat in magnitude or phase response.

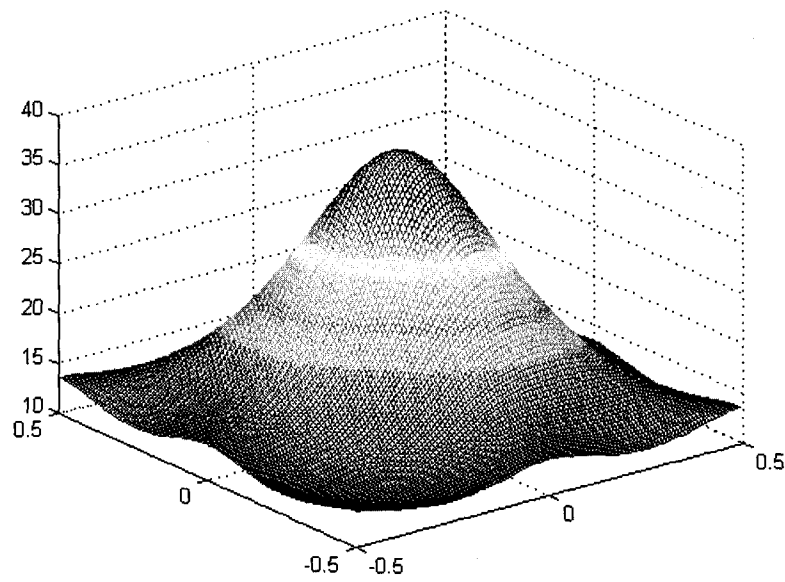


Figure 2 Theoretical magnitude, mesh plot. Signal has a mean-shifted exponential distribution with variance = 4 and filter $\beta = 0.25$.

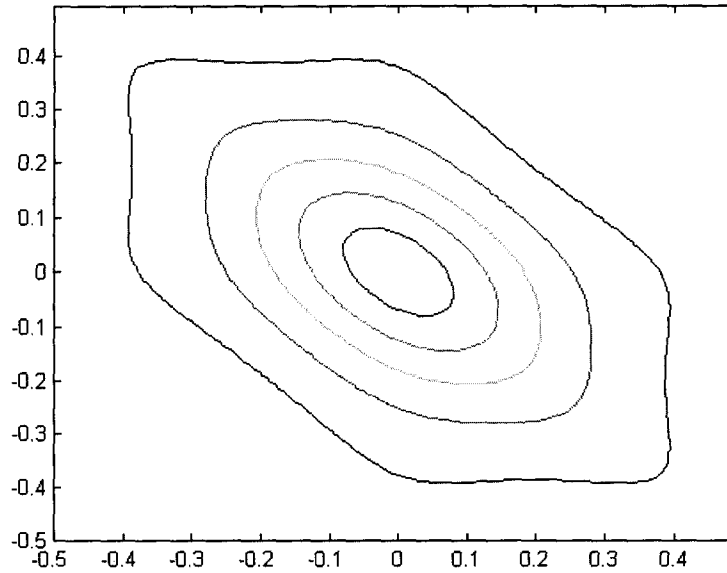


Figure 3 Theoretical magnitude, contour plot. Signal has a mean-shifted exponential distribution with variance = 4 and filter $\beta = 0.25$.

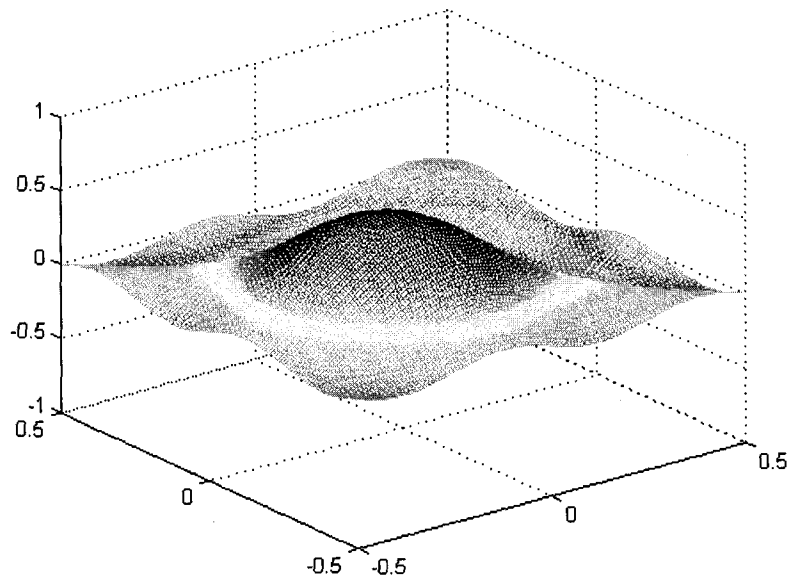


Figure 4 Theoretical phase, mesh plot. Signal has a mean-shifted exponential distribution with variance = 4 and filter $\beta = 0.25$.

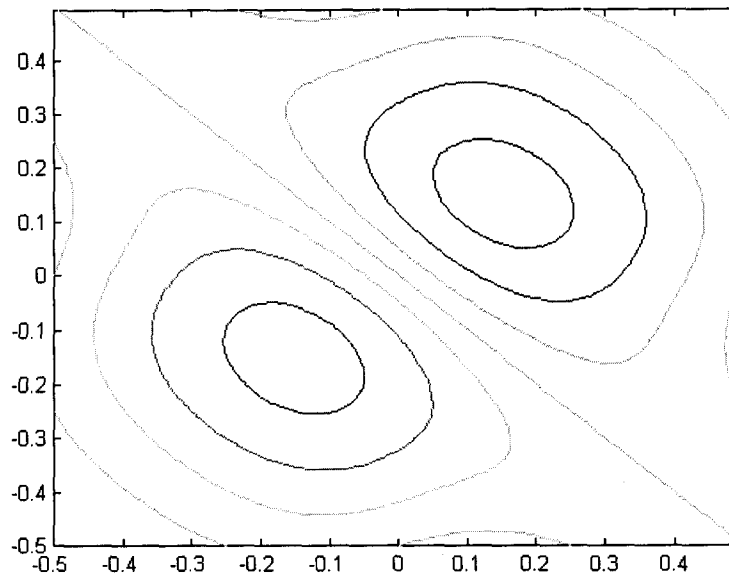


Figure 5 Theoretical phase, contour plot. Signal has a mean-shifted exponential distribution with variance = 4 and filter $\beta = 0.25$.

By comparing the results produced by *bispeci* with these predicted results, the effects of changing the various arguments to *bispeci* were studied. The theoretical results were also used to calculate mean-squared error (MSE) values for each estimated bispectrum. The MSE equation is given by [Ref. 3: p. 312]

$$\xi = E \{ |B - \hat{B}|^2 \}, \quad (10)$$

where B is the theoretical value being examined and $\hat{B}$ is the experimentally obtained value. The mean-square error values calculated are given in the "Experimental Results and Conclusions" section, following.

B. PROCEDURE

The procedure followed was to generate a data ensemble for each of the four distributions, and then to use *bispeci* to generate bispectral estimates of each ensemble. As the parameters to *bispeci* were varied, the effects on the bispectral estimates were examined. After the original distributions were examined, the MATLAB *filter* function was used to produce a shaped version of each signal ensemble. The *bispeci* function was then used to generate bispectral estimates of these shaped signals, which were examined as well.

The size of the datasets was set according to the conclusions of the previous Chapter; matrices comprising 256 realizations of 512-point sequences were used. This dataset size was the minimum necessary to provide consistent third-order cumulant estimates. As before, the variance of each data sequence was set to four, so that the third-order cumulant estimates from Chapter II would still be valid.

Initially, the datasets were provided to *bispeci* and the number of lags to be computed was varied from 32 to 512 while the DFT size was left at its default. If not specified by the user, the DFT length in *bispeci* defaults to the next power of two greater than twice the lag argument (e.g., the MATLAB command `[bspec,waxis] = bispeci(y,32)` has a lag argument of 32; this command sets the DFT length to its default of 128). Table 4 shows the default DFT length for each lag argument used here.

Next, the datasets were provided to *bispeci* with a lag argument of 256, but the DFT length was increased rather than allowed to remain as its default.

<i>Lag argument</i>	<i>Default DFT length</i>
32	128
64	256
128	512
256	1024
512	2048

Table 4 Default DFT values for bispeci lag arguments

Because the datasets used were matrices, the data segment size and overlap parameters could not be changed manually. However, this was not a problem, as the *bispeci* function properly interpreted each matrix as a series of realizations of the same random sequence. Biased estimates were used throughout the procedure. The choice of windows is limited in *bispeci* to either a Parzen or a unit hexagonal window. Neither of these choices, however, seemed to greatly affect the bispectral plots and hence the default Parzen window was used throughout.

As the various bispectral estimates were calculated, it was observed that they were not purely real, as theory states they should be, but had imaginary components. Using the estimate magnitudes for three-dimensional *mesh* plots and mean-square error calculations produced reasonably accurate results; however, the presence of the imaginary components did serve as a source of error.

For example, Figure 6 shows the phase of one of the bispectral estimates, that of the mean-shifted exponential distribution, when calculated using a lag argument of 128 and a DFT length of 512. The phase in both frequency dimensions should be flat and zero-valued, and clearly is not.

It was determined that the *bispeci* function was adding a bilinear phase term as it generated the bispectral estimates. (It was not clear from examination of the source code how this unwanted phase was produced, but it is apparently due to a temporal shift of the data.) Repeated attempts to eliminate the imaginary components while maintaining an accurate bispectral estimate were also unsuccessful. Therefore, the magnitude of the bispectral estimate was used whenever a plot was generated or a mean-square error value was calculated.

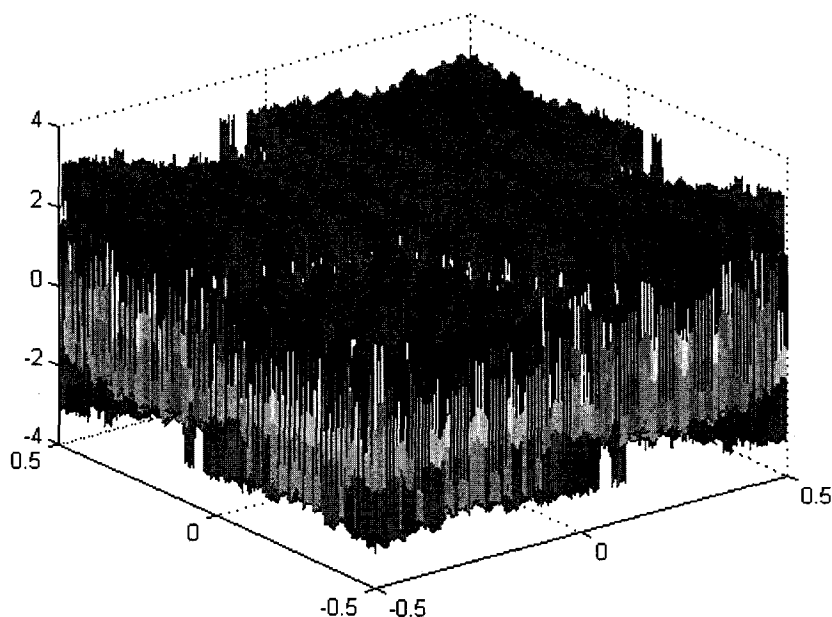


Figure 6 Phase (in radians) of bispectral estimate. Signal has mean-shifted exponential distribution and has been colored ($\beta = 0.25$). Lag argument used in bispectral estimate is 128.

Plots (generated using the MATLAB *mesh* function) of the bispectral estimate magnitudes are shown in Appendix B, along with some observations.

After the four original datasets were examined, the MATLAB *filter* function was used to produce filtered versions of the two asymmetrically-distributed signals (mean-shifted exponential and mean-shifted Rayleigh). The filter used was the basic, one-pole LPF described in the previous section, and the filter parameter β was initially set to 0.25.

Bispectra for these shaped signals were then generated, using *bispeci*. This procedure was then repeated with β set to 0.1, and again with β set to 0.05. Each set of bispectral estimates was compared to its theoretical value to calculate a mean-square error result.

As in the case of the flat signals, bispectral estimates for the shaped signals were first produced as the lag argument to *bispeci* was increased from 32 to 512. Once these six bispectral estimates were calculated, the lag argument was set to 128 and the DFT length was increased. The bispectral plots produced are in Appendix B as well, along

with some interim observations.

C. EXPERIMENTAL RESULTS AND CONCLUSIONS

The bispectra of the unshaped signals were indeed flat, although some variance (in the form of jaggedness in the bispectral plots) was noticeable, which grew as the number of lags calculated increased. The mean of the bispectrum of each of the 24 white signals (six bispectral estimates each of the four signal ensembles, corresponding to Figures 21 through 44 in Appendix B) is given below in Table 5. In every case, with the exceptions of the lag and DFT size arguments, the parameters provided to *bispeci* were left as their default values.

Table 6 contains mean-squared error values generated from the same bispectral estimates. The theoretical values used here were those given at the top of Table 5; the theoretical bispectra of these white signals are flat planes of a particular value, as discussed above.

<i>bispeci</i> parameters	<i>Gaussian</i>	<i>Uniform</i>	<i>Mean-shifted exponential</i>	<i>Mean-shifted Rayleigh</i>
Theoretical value	0.0000	0.0000	16.0000	5.04739....
nlag = 32, nDFT = 128	0.4441	0.4027	15.7044	5.0097
nlag = 64, nDFT = 256	0.8838	1.2886	15.7655	5.0431
nlag = 128, nDFT = 512	1.7145	2.208	15.8604	5.1738
nlag = 256, nDFT = 1024	3.2327	3.5828	16.0357	5.6939
nlag = 512, nDFT = 2048	3.3218	3.4421	15.9785	5.751
nlag = 256, nDFT = 2048	3.2333	3.5831	16.0358	5.6944

Table 5 Mean values of bispectral estimates of white signals

<i>bispeci parameters</i>	<i>Gaussian</i>	<i>Uniform</i>	<i>Mean-shifted exponential</i>	<i>Mean-shifted Rayleigh</i>
nlag = 32, nDFT = 128	0.2596	0.2076	0.3185	0.1700
nlag = 64, nDFT = 256	1.0056	2.0902	2.1513	0.5834
nlag = 128, nDFT = 512	3.7679	6.1982	5.1110	1.8885
nlag = 256, nDFT = 1024	13.3734	16.3846	10.7852	5.8966
nlag = 512, nDFT = 2048	14.0492	15.0831	8.6563	6.1501
nlag = 256, nDFT = 2048	13.3734	16.3846	10.7844	5.8908

Table 6 Mean-squared error for bispectral estimates of white signals

An initial observation is that the mean values and the mean-square error of the bispectral estimate, rises as the lag argument and DFT length is increased. This is due to the increased variance of the bispectral estimate as longer lags are used. However, comparing the fourth and last rows of Tables 5 and 6 also suggests that the DFT length is not a factor, as holding the lag argument while increasing the DFT length does not significantly affect the calculated values of either the mean or the mean-square error.

It is also interesting to note that the mean-squared error values calculated for the mean-shifted Rayleigh signal are significantly lower than those calculated for the mean-shifted exponential signal.

Overall, the mean values seem to correspond to those predicted by theory and also to fit with the plots contained in Appendix B.

After the white datasets were examined, the signals which had been shaped by the LPF were used to generate bispectral estimates. Figure 7, reproduced from Appendix B [Figure 46], shows the result for one of the shaped signal ensembles. In this case, a mean-shifted, exponentially-distributed ensemble was generated with variance $\sigma^2 = 4$, and filtered by the LPF described above. The bispectrum of the resultant signal was then estimated using a lag argument of 32 and DFT length of 128 and is plotted here.

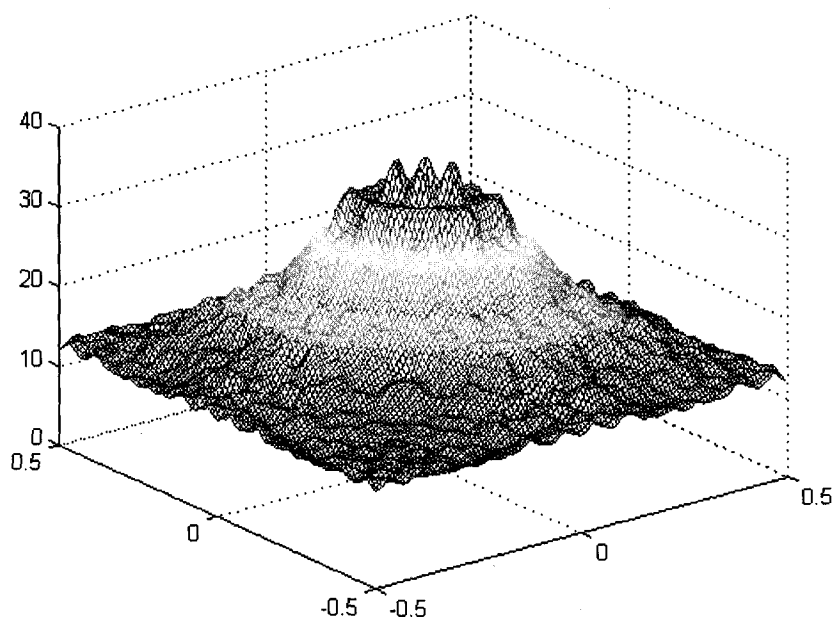


Figure 7 Estimated bispectral magnitude. Signal has mean-shifted exponential distribution and has been colored (filter $\beta = 0.25$). Bispectral estimate was generated with 32 lags and DFT length of 128.

Comparing Figure 7 with Figure 2 shows that, although the signal has indeed been shaped by the LPF, the difference between the bispectrum estimate and its theoretical value is significant. Once the bispectral estimates were complete, their theoretical values were used to calculate mean-square error figures. These values are given in Table 7, below.

<i>bispeci</i> parameters	<i>Gaussian</i>	<i>Uniform</i>	<i>Mean-shifted exponential</i>	<i>Mean-shifted Rayleigh</i>
nlag = 32, nDFT = 128	0.3623	0.2549	0.4654	0.2083
nlag = 64, nDFT = 256	1.3239	2.6350	2.6927	0.7035
nlag = 128, nDFT = 512	4.8376	7.8563	6.4024	2.3588
nlag = 256, nDFT = 1024	16.9476	20.6921	13.6114	7.2742

Table 7 Mean-squared error for bispectral estimates of colored signals ($\beta = 0.25$)

The filter parameter β was then reduced, from 0.25 to 0.1, and the bispectral estimates and their predicted values were again calculated. This new set of values were used to generate a second set of mean-square error results, which are given in Table 8.

<i>bispeci parameters</i>	<i>Gaussian</i>	<i>Uniform</i>	<i>Mean-shifted exponential</i>	<i>Mean-shifted Rayleigh</i>
nlag = 32, nDFT = 128	0.3212	0.2463	1.1248	0.2624
nlag = 64, nDFT = 256	1.1028	2.2425	3.1781	0.6969
nlag = 128, nDFT = 512	3.8865	6.2737	6.5623	2.1195
nlag = 256, nDFT = 1024	13.6627	16.5790	12.9732	6.3619

Table 8 Mean-squared error for bispectral estimates of colored signals ($\beta = 0.1$)

Finally, the filter parameter β was reduced once more to 0.05. The experimental and theoretical bispectral estimates were again calculated, and again used to generate a set of mean-square error measurements. These values are given in Table 9.

<i>bispeci parameters</i>	<i>Gaussian</i>	<i>Uniform</i>	<i>Mean-shifted exponential</i>	<i>Mean-shifted Rayleigh</i>
nlag = 32, nDFT = 128	0.3104	0.2420	0.3822	0.2134
nlag = 64, nDFT = 256	1.0747	2.2135	2.3498	0.6405
nlag = 128, nDFT = 512	3.8109	6.1732	5.6016	2.0147
nlag = 256, nDFT = 1024	13.3975	16.2237	11.7695	6.0951

Table 9 Mean-squared error for bispectral estimates of colored signals ($\beta = 0.05$)

Recall that the filter parameter β can be interpreted as a feedback coefficient in the filter difference equation. This means that lowering its value effectively reduces the effect of the feedback path, and should be reflected in lower mean-square error values. An examination of Tables 7 through 9 shows that this is indeed generally the case.

Overall, the most significant conclusion here is that greatly increasing the length of DFTs used, in and of itself, does not significantly alter the mean-square error of the bispectral estimate (see Table 5). The mean-square error does go up when both the lag argument and the DFT length are increased, but when the lag argument is held constant and the DFT length only is increased, the effect on the mean-square error is minimal.

This is significant because DFT length would almost certainly be a critical issue if one were to design a practical signal processing application using this method. While DFT length is not completely insignificant, it would still be helpful for designers to know that scarce hardware resources or processing cycles might be more usefully spent elsewhere in a signal processing application.

Another significant discovery is the difference between the mean square error of the bispectral estimates of mean-shifted exponential signals and those produced by the bispectral estimates of mean-shifted Rayleigh signals when the signals have been shaped by a filter (see Tables 7 through 9). The lower error values for the mean-shifted Rayleigh distribution suggest that it might be a more appropriate distribution for an information-bearing signal, if a communication system were being developed to take advantage of a higher-order statistical function for signal recovery.

Of course, if this method were indeed to be developed into a real-world system, any received signals would not be simply shaped, shifted versions of the signals transmitted; they would also include additive noise. The receiver would be required to distinguish the signal of interest from this noise—which typically has a Gaussian distribution and is frequently referred to as additive white Gaussian noise (AWGN). The next Chapter, then, examines the effects of adding a shaped, asymmetric information-bearing signal to such noise.

THIS PAGE INTENTIONALLY LEFT BLANK

IV. NONGAUSSIAN SIGNALS IN THE PRESENCE OF GAUSSIAN NOISE

A. THEORY

1. Introduction

In practice, information-bearing signals are not seen by themselves; they occur in an environment containing interfering noise sources, both natural and manmade. This Chapter investigates the effects of combining colored non-Gaussian information signals with Gaussian noise signals, at differing signal-to-noise ratios (SNR).

2. Theoretical Predictions

As discussed in Chapter I, the bispectrum of a Gaussian-distributed signal is theoretically flat, with magnitude zero. Since the cumulants of a sum of independent signals add, the bispectrum of a non-Gaussian signal in additive Gaussian noise should be identical to the bispectrum of the non-Gaussian signal alone.

However, it seems likely that in practice the Gaussian noise will not be completely eliminated at higher orders. In this case, the Gaussian noise with a lower variance should be suppressed more effectively than that with a higher variance. This stands to reason in light of the fact that variance may be viewed as a measure of signal strength.

Based on the results of Chapter III, another reasonable prediction would seem to be that the mean-shifted exponential ensemble will produce larger bispectral components, but at a cost of greater variance.

B. EXPERIMENTAL RESULTS

1. Procedure

In MATLAB, four Gaussian noise signal ensembles were created, with variances $\sigma^2 = 0.5$, $\sigma^2 = 1$, $\sigma^2 = 4$, and $\sigma^2 = 16$. Additionally, mean-shifted exponential- and Rayleigh-distributed "information-bearing" signals were generated, both with a variance of four. These information signals were filtered, as in the previous Chapter, by a shaping filter with a filter parameter $\beta = 0.1$. This parameter was chosen based on the results of the previous Chapter; $\beta = 0.1$ was the middle value examined and appeared to produce reasonable mean-square error values.

The signal to noise ratio for the ensembles with a noise variance $\sigma_n^2 = 4$ was thus

$$SNR = 10 \log_{10} \frac{\sigma_s^2}{\sigma_n^2} = 10 \log_{10} \frac{4}{4} = 0 \text{ dB}. \quad (11)$$

Similarly, the SNR for the ensembles with a noise variance $\sigma^2 = 0.5$ is 9.03 dB; that for the ensembles with a noise variance $\sigma^2 = 1$ is 6.02 dB; and that for the ensembles with a noise variance $\sigma^2 = 16$ is -6.02 dB.

After the colored signal ensembles were added to the noise dataset, the resulting signals were examined using *cumest* (the HOSA third-order cumulant estimation function) and *bispeci*. Appendix C contains the plots generated by *bispeci*, as well as further discussion of each plot.

First, the exponential data ensemble from the previous Chapters was combined with each of the two Gaussian noise files. This process was then repeated with the Rayleigh data ensemble. Once the combined ensembles were ready, the third-order cumulant of each data and noise ensemble was estimated using the HOSA function *cumest*.

Next, *bispeci* was used to generate bispectral estimates of the four datasets. Based on the results of Chapter III, two bispectral estimates were calculated for each dataset. For the first, the number of lags specified was 128, and the rest of the parameters to *bispeci* were allowed to remain as their defaults (see Table 4 in Chapter III). This set the DFT length to 512 points, the window to "Parzen", and the estimate type to "biased." The lag argument (and hence the DFT length as well) were chosen as the minimum length to provide reasonable bispectral estimates; this kept the computing time and power required (a significant consideration here) to a minimum.

The second bispectral estimate used a larger lag argument of 256. Again, the rest of the parameters were allowed to remain at their defaults. These second estimates were generated to determine whether increasing the lag argument had a significant effect on the bispectral estimate of the signal and noise combination.

2. Data

Table 10, below, contains the third-order cumulant estimate results for each SNR value of the mean-shifted exponentially-distributed signal ensemble used, and the percent error that each represents.

<i>Theoretical SNR (dB)</i>	<i>Calculated third-order cumulant</i>	<i>Percent error</i>
9.03	15.7655	1.5 %
6.02	15.6215	2.4 %
0	15.8593	0.88 %
-6.02	15.6328	2.3 %

Table 10 Cumulant and percent error at zero lag, colored mean-shifted exponential distributed sequence in AWGN (signal expected value = 16)

Table 11 contains the third-order cumulant estimate results for each SNR value of the mean-shifted Rayleigh-distributed signal ensemble used, and the percent error each represents.

<i>Theoretical SNR (dB)</i>	<i>Calculated third-order cumulant</i>	<i>Percent error</i>
9.03	4.9946	1.05 %
6.02	5.0298	0.35 %
0	5.1447	1.93 %
-6.02	4.9311	2.3 %

Table 11 Cumulant and percent error at zero lag, colored mean-shifted Rayleigh distributed sequence in AWGN (signal expected value = 5.04739)

After the cumulants of each combined signal and noise ensemble were calculated, the bispectrum of the ensemble was estimated using *bispeci*. For each bispectral estimate produced, a mean-square error figure was produced. For the mean-square error calculation, the theoretical values used were those from the previous Chapter. The first set of bispectra produced used a lag argument of 128, and the resulting mean-square error values are given here in Table 12.

SNR (dB)	<i>Mean-shifted exponential data sequence</i>	<i>Mean-shifted Rayleigh data sequence</i>
9.03	6.5442	2.7481
6.02	7.5014	3.7146
0	18.4954	13.7302
-6.02	220.4491	320.4302

Table 12 Mean-square error for colored signals in Gaussian noise, 128 lags computed

The second set of bispectra produced used a lag argument of 256. The mean-square values calculated for these estimates are given in Table 13.

SNR (dB)	<i>Mean-shifted exponential data sequence</i>	<i>Mean-shifted Rayleigh data sequence</i>
9.03	14.8658	8.5169
6.02	18.1758	11.4167
0	54.3311	54.7393
-6.02	933.9822	1333.0

Table 13 Mean-square error for colored signals in Gaussian noise, 256 lags computed

3. Conclusions

One immediate observation is that, for the bispectral estimations, the mean-square error values increase as the signal-to-noise ratio drops—dramatically as the SNR goes to -6.02. This suggests that, at least with this bispectral estimator, Gaussian signals do not completely vanish at higher orders as theory predicts they should.

As in the previous Chapter, it is also clear that the bispectrum of a signal ensemble generated with the mean-shifted exponential distribution is significantly different from that produced by a signal ensemble whose underlying distribution is the mean-shifted Rayleigh distribution. The bispectra of the mean-shifted exponential distribution contains many more visible signal components than those of the mean-shifted Rayleigh distribution. This is due to the fact that, for a given variance, the third-order cumulant of a signal with a mean-shifted exponential distribution is different from that of one with a mean-shifted Rayleigh distribution. This difference remained visually

apparent even when the information-bearing signals were combined with Gaussian noise signals (see Appendix C).

It is interesting to note that the mean-square error for mean-shifted exponentially-distributed signal ensembles were typically greater than those for mean-shifted Rayleigh-distributed ensembles. This is not the case, however, as the SNR drops from zero to -6.02 , suggesting that while the mean-shifted Rayleigh distribution may be more useful at low signal-to-noise ratios, its relative utility drops significantly as the noise level gets stronger.

A difference between Tables 12 and 13 which is not evident here is the amount of time required to generate the bispectral estimates; doubling the lag argument more than doubled the amount of time required to generate the bispectra. This is a result of the DFT calculations involved, and would obviously be a significant consideration in any real-world signal processing application.

It is also evident, in comparing Tables 10 and 11 with Tables 12 and 13, that the error introduced goes up dramatically when bispectra are calculated, rather than just cumulant estimates. Also, of course, the time required to produce bispectral estimates is significantly greater as well.

Returning to Tables 10 and 11, it is interesting to note that the percent error of the cumulant estimate remains fairly low for both distribution types, even as the variance of the noise signal is doubled and doubled again. This suggests that the third-order cumulant estimate does indeed have some resistance to Gaussian noise, and could be appropriate for Gaussian noise mitigation. However, Tables 12 and 13 suggest that the bispectral estimation does not share this resistance, at least not to the same degree.

It is difficult to draw an overall conclusion as to whether a mean-shifted exponential distribution or a mean-shifted Rayleigh distribution would be preferable in an information-bearing signal (assuming that higher-order statistics were to be used in processing the signal). While the mean-square error is generally less for the mean-shifted Rayleigh signal, at least at high SNRs, the plots in Appendix C suggest that it would be difficult to distinguish the signal from the noise floor. The large mean-square error values, and the high variance of the plots produced by bispectral estimates, strongly suggest it would be preferable to work with cumulant estimates rather than bispectral estimates, if possible.

THIS PAGE INTENTIONALLY LEFT BLANK

V. CONCLUSION

As computing power continues to increase, signals become more complex and signal environments become more dense, it seems likely that interest in higher-order statistics will continue to increase. The efforts described here indicate some results which would be useful for anyone trying to implement a signal processing system (noise mitigation or other) using higher-order statistics. Additionally, the work has pointed out some other issues which will need to be addressed before such a system could be designed.

A. OVERALL CONCLUSIONS

One of the two most significant conclusions uncovered here was that the length of the DFT used does not seem to significantly affect the mean-square error in bispectral estimation of colored signals. This is somewhat surprising, as increasing DFT length would be an immediate, intuitive step to improve spectral estimation. It might be explained, however, as an artifact of the estimation method used here. The datasets used in Chapters III and IV, based on the conclusions of Chapter II, were 256 realizations of 512 points each. As this matrix is fed into the indirect estimator, the cumulants are estimated and the Fourier transform is performed—including zero-padding if the DFT length is greater than that of the data. This could explain the diminishing returns exhibited in Tables 5 and 6 as the DFT length is increased beyond 512 (the sample realization size).

Another useful (but perhaps intuitive) result is the amount of error introduced in performing bispectral estimation as opposed to the relatively simpler cumulant estimation. Chapter III details that the MATLAB *bispeci* estimator somehow introduces phase components to its estimates and the effort spent to remove these components. One of the often-touted advantages of higher-order statistics is that it is phase-blind [Ref. 7: p.3]; in light of this fact the introduction of phase components by an application is a significant issue.

Similarly, Chapter IV shows that even in the presence of noise, a cumulant estimate may be accurately calculated where a bispectral estimate will be greatly different from its theoretical value. These two observations strongly recommend that, if possible, any application be developed to operate solely on cumulant estimates, rather than bispec-

trum estimates. Naturally, the added computation—and time—involved in calculating bispectra would also recommend simply using cumulant estimates.

The results of Chapter III, and part of Chapter IV, also speak to the value of the mean-shifted Rayleigh distribution at higher orders, with respect to the mean-shifted exponential distribution. The mean-square error values calculated in both Chapters suggest that cumulant and bispectral estimates of data ensembles with a mean-shifted Rayleigh distribution are more accurate than those of data ensembles with a mean-shifted exponential distribution. The plots in Appendices B and C would seem to mitigate this conclusion somewhat, as the signal energy of the mean-shifted Rayleigh ensembles is much lower than that of the mean-shifted exponential ensembles. However, this is based simply on a visual examination of the plots, and any evaluation of the different distributions for a practical application should be performed keeping in mind the needs of the application itself (i.e., the appearance of the bispectrum is irrelevant if the processor in question is able to distinguish the signal properly).

Overall, the results seen here suggest that higher-order statistics continue to hold promise as a method for reducing the effects of Gaussian noise and interference. However, a number of additional questions will need to be answered before they can be used to construct an effective processing system for realistic signal environments.

B. STEPS FOR FURTHER STUDY

As discussed in Chapter I, the ultimate impetus behind this work was the possibility of a practical signal processing application, such as interference mitigation, using higher-order statistics. The conclusions arrived at here are useful pointers towards the development of such an application, but several intermediate steps would have to be performed before it could be realized.

An obvious first step would be to study additional distributions. Just as the "mean-shifted" versions of exponential and Rayleigh distributions were generated by subtracting the mean from standard distributions, other such distributions can be generated as well. As long as a distribution is asymmetric and zero-mean, its third-order cumulant can be theoretically predicted as in Chapter II. A list of such distributions would be useful to select one with the strongest third-order presence (either its cumulant or its

bispectral magnitude), to be used as an information-bearing signal in a Gaussian noise environment.

If a likely candidate signal was determined, it would serve as the basis for development of the application itself (whether it be an active filter, demodulator, or other signal processing application, in hardware or in software). The combination of information-bearing signal and expected application would drive one of the next steps of analysis, which is a determination of the SNR required given a signal and receiver type (i.e., a basic link budget). As seen in Chapter IV, background noise—even Gaussian noise—takes a toll on a signal, whether at higher orders or not, and its effect must be taken into account in system design.

If bispectral estimation (rather than simple cumulant estimation) were necessary, an additional study of value would be an examination of bispectral estimators which were not used here. In MATLAB, this would mean studying *bispecd*, an estimator using the "direct" method, and *bispect*, a parametric estimator. It would be of value both to gauge the accuracy of these methods with respect to each other, as well as to understand the processing requirements of each. (It may be that one is more suitable than the others for hardware acceleration, for example, and would make it a more desirable candidate.) If the *bispeci* function were to be utilized, as it was here, it would be necessary to understand exactly how the unwanted phase components are introduced, and how to remove them. This would be particularly necessary in any receiving or filtering application designed with modern communications systems in mind, as a large proportion of such systems use signal phase to carry information (e.g., the various types of phase-shift keying and quadrature-amplitude modulation schemes).

Regardless of whether any of these avenues are pursued further, it seems clear that higher-order statistics will be continuing their progress from the purely theoretical domain, further and further into the realm of practical and valuable signal processing techniques.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX A. MATLAB RESULTS FROM CHAPTER II

This Appendix contains detailed results of the procedure described in Chapter II. It is divided into three Sections, each of which is grouped by one of the three statistics being examined (ensemble mean, ensemble variance, and ensemble third-order cumulant). Section A, for example, contains a plot showing the ensemble mean for each of the four distribution types examined, as the ensemble size was lengthened. In this way, the effects of changing the ensemble size could be studied both with respect to the statistic of interest and with respect to the distributions under study.

A. Statistic: Mean

Figures 8 through 11 demonstrate the effect of increasing the ensemble size on the average of the ensemble means. In each case, the expected value of the ensemble mean is zero (each ensemble comprises a series of realizations of zero-mean sequences—please see Appendix F for a description how a series of realizations is generated to produce an ensemble and, in turn, the statistics of interest). Please note that in these plots, as in the following ones, the expected value is indicated by a line of dots, spaced logarithmically.

Figure 8 shows that the Gaussian distribution exhibits a small amount of "ringing" (overshoot of the expected value), but the absolute error is still never greater than 0.2. Regardless, the curve flattens out significantly, suggesting that for 512-point, Gaussian-distributed sequences, an ensemble of as few as 128 sequences is sufficient to produce a consistent mean estimate.

Figure 9 shows that the uniform-distributed ensembles are also very close to their expected values even at short ensemble sizes. Even an ensemble length of just one realization exhibits an absolute error of less than 0.1. Also, as with the Gaussian sequences, after some initial variation the mean estimates seem to approach their expected value asymptotically. While the values are "close enough" at just about any ensemble size, they are the most stable for ensemble lengths of 128 or more realizations.

Figure 10 shows that, for the mean-shifted exponential distribution, the data follows the same general trend as that of the previous two plots. However, note that the scale is even smaller, less than 2×10^{-15} at its largest. In this case, any ensemble size seems

to produce a sufficiently accurate mean estimate, and for ensemble sizes of 64 or more realizations the mean estimates seem to approach their theoretical value.

Figure 11, a plot of the ensemble mean estimates of the mean-shifted Rayleigh sequences, is at an even smaller scale, producing an absolute error of less than 8×10^{-16} at its largest. The graph appears to demonstrate more "ringing" than any of the three previous plots—although the mean estimate oscillates around its predicted value, it does not ever seem to approach it asymptotically. However, when the small scale of the plot is considered, it becomes evident that the estimate at any ensemble size is highly accurate, and the oscillation is probably as much a function of the precision of the simulation, as it is of the math itself. In any case, an ensemble size of 64 realizations should be sufficient to produce accurate mean estimates.

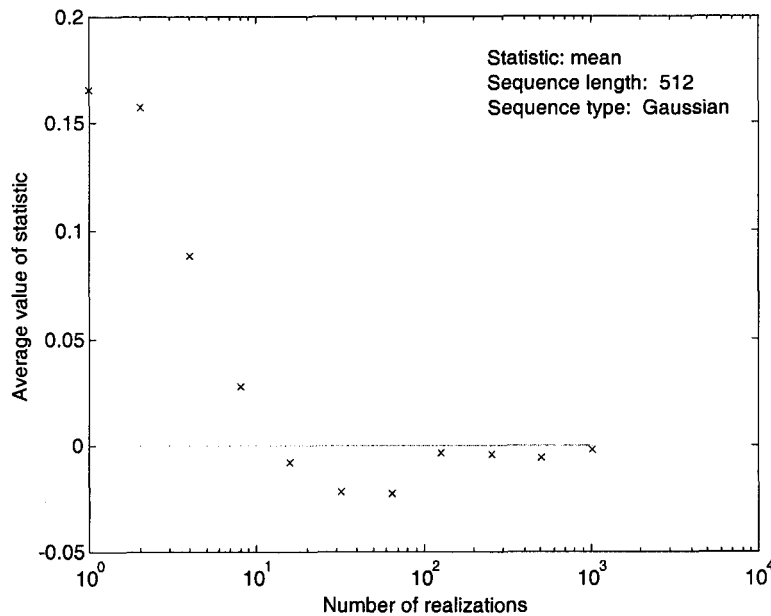


Figure 8. Gaussian distribution, statistic: mean.

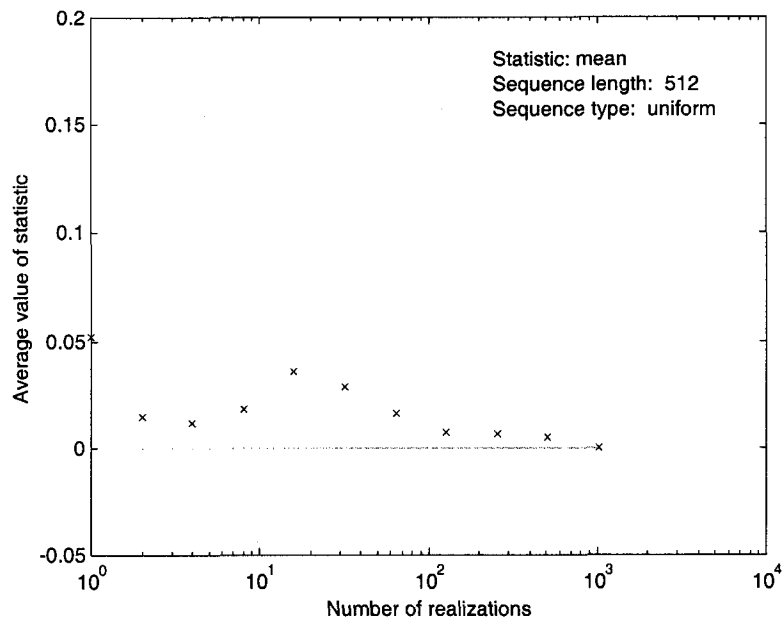


Figure 9. Uniform distribution, statistic: mean.

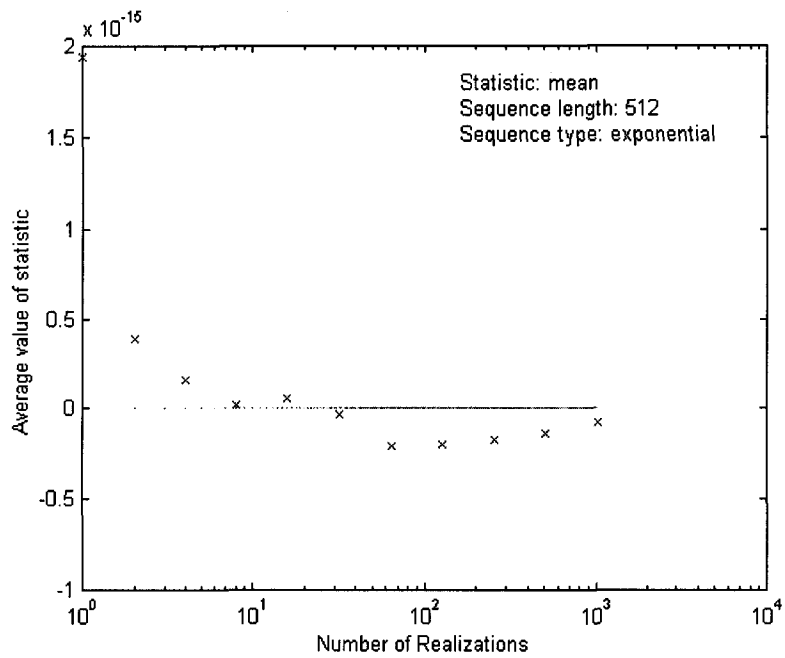


Figure 10. Mean-shifted exponential distribution, statistic: mean.

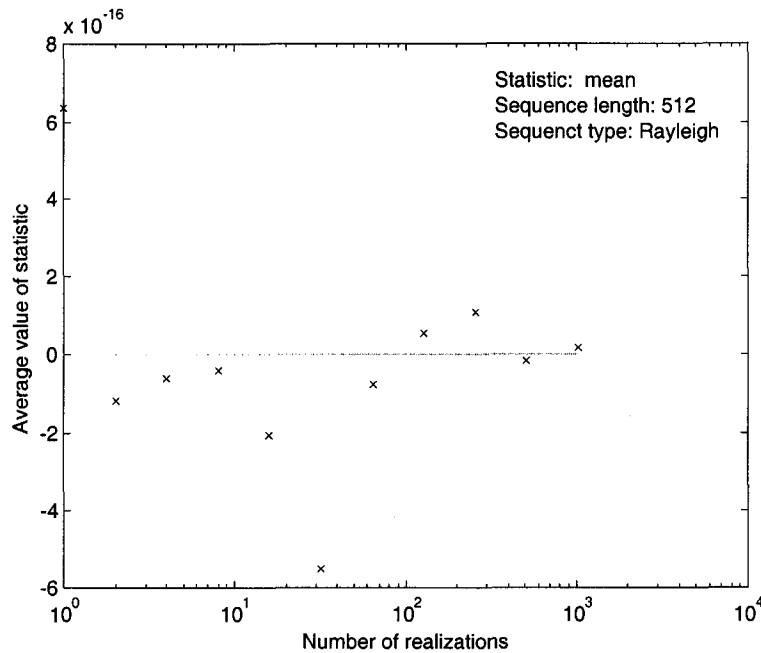


Figure 11. Mean-shifted Rayleigh distribution, statistic: mean.

B. Statistic: Variance

Figures 12 through 15 plot the average values of the ensemble variances as a function of ensemble length. In all four cases, the ensembles comprise realizations of sequences whose variances have been set to four; this is therefore the expected value in each case.

Figure 12 shows the average value of the ensemble variance for Gaussian-distributed sequences. It shows that for any ensemble length of a mere four realizations or more, the experimental error is less than one percent. As in the mean plots, the data points show some ringing but after a certain point converge on their expected value. In this case, an ensemble length of 128 points will certainly be more than adequate.

Figure 13 shows similar behavior for the uniform-distributed sequences, with an ensemble length of eight or more realizations sufficing to produce less than one percent error. Again, at an ensemble length of 128 realizations, the plot seems to converge on its theoretical value.

Figure 14, the variance estimates for the mean-shifted exponential sequences, shows an exponential decay of the variance towards its theoretical value (with the case of

an ensemble length of one as the exception). The error in this plot seems to remain larger for longer ensemble lengths than in any of the previous plots, remaining at greater than one percent until the ensemble length reaches 128 realizations. However, at this point, the error continues to shrink.

Figure 15 shows that, compared to the previous plot, the mean-shifted Rayleigh sequences seem to meander a bit in their variance estimates, but nevertheless move towards their theoretical value. The plot demonstrates that an ensemble length of 256 or more realizations is sufficient to put the experimental error at less than one percent.

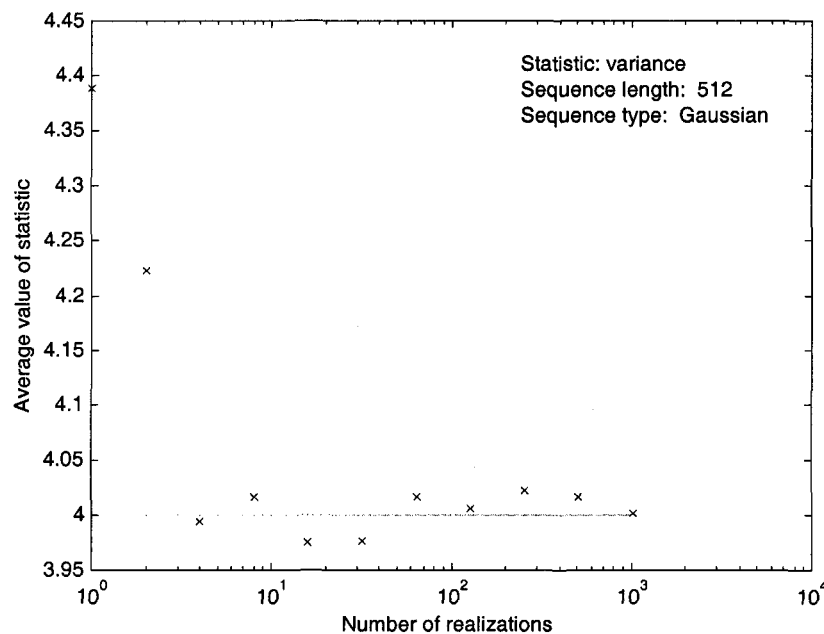


Figure 12. Gaussian distribution, statistic: variance.

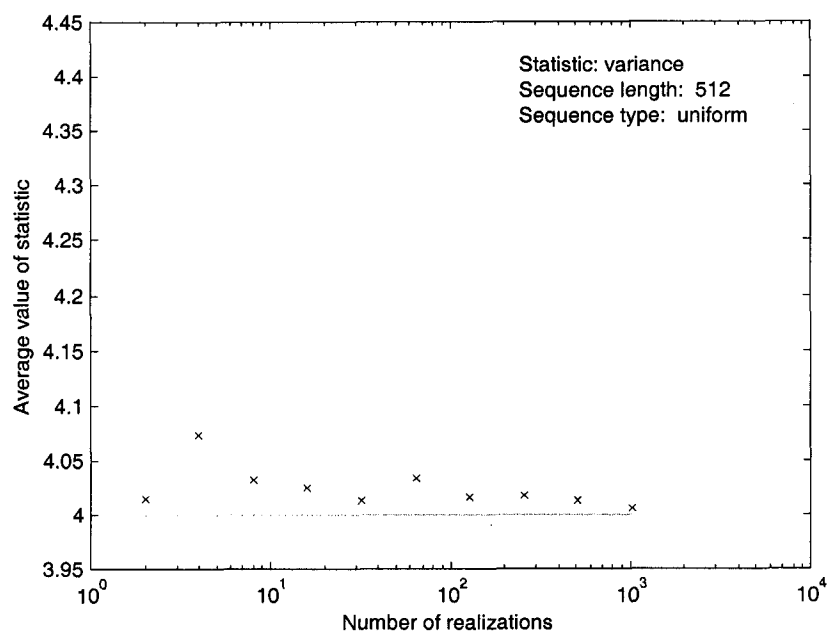


Figure 13. Uniform distribution, statistic: variance.

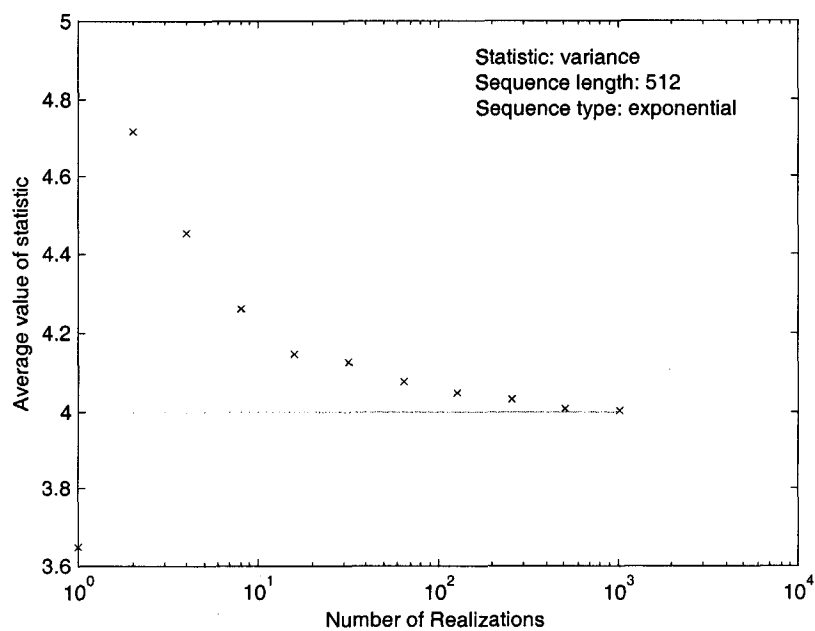


Figure 14. Mean-shifted exponential distribution, statistic: variance.

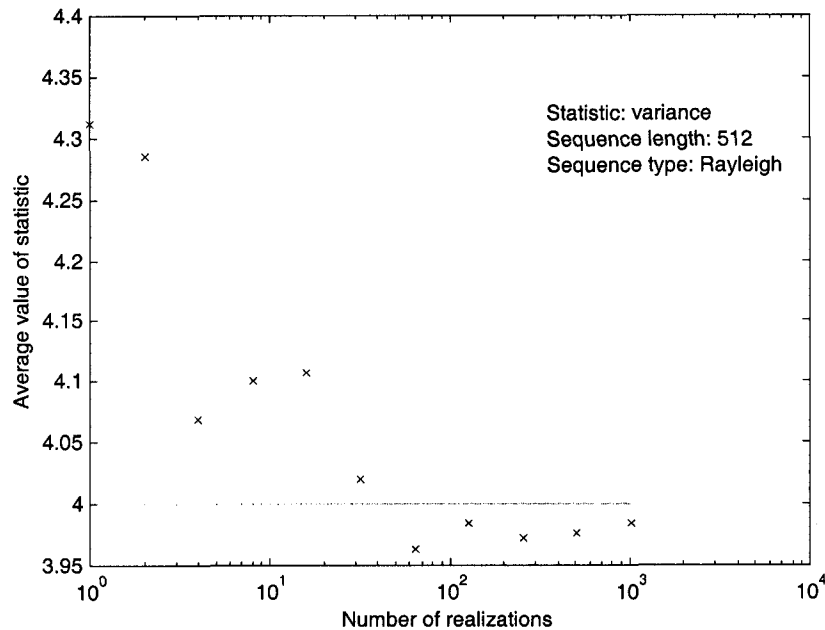


Figure 15. Mean-shifted Rayleigh distribution, statistic: variance.

C. Statistic: Third-order cumulant estimate

Figures 16 through 19 show the average values of the third-order cumulant estimates as a function of ensemble length. As noted above, for the symmetric Gaussian and uniform distributions (Figures 16 and 17, respectively), the expected value is zero. For the mean-shifted exponential distribution, the expected value is 16, and for the mean-shifted Rayleigh distribution, the expected value is approximately 5.043 (this is a truncation of the calculated expected value).

Figure 16 shows the third-order cumulant estimate for the series of Gaussian-distributed sequences. It shows some ringing as in the previous plots, with the mean calculated estimates closing to within an absolute error of less than 0.2 for any ensemble size of four or more realizations. To close to within an absolute error of 0.1 or less requires a minimum ensemble length of 256 realizations.

Figure 17 shows the third-order cumulant estimates for the uniform-distributed sequences. Its plot approaches the theoretical value, although certainly not asymptotically, and exhibits the interesting feature that all the error values (except for the last) are negative for some reason. At any rate, the absolute error remains at less than 0.1 for any

ensemble length of 64 or more realizations.

Figure 18 shows the third-order cumulant estimates for the mean-shifted exponential distribution. Similarly to that of the variance estimates, the plot of the third-order cumulant estimates approaches its theoretical value in an exponential decay (again, ignoring the very first data point). In this case, the experimental error drops to within five percent for ensemble sizes of 64 or more realizations, and drops to within one percent for ensemble sizes of 256 or more realizations.

Figure 19 shows the third-order cumulant estimates for the mean-shifted Rayleigh distribution. While this plot also closes in on its theoretical value, it does so in neither an oscillating nor an asymptotic fashion. It also never has one percent error or less, for any ensemble length examined (the maximum ensemble length here is 1024 realizations). However, once the ensemble length reaches 128 realizations, the experimental error is less than five percent and continues to shrink.

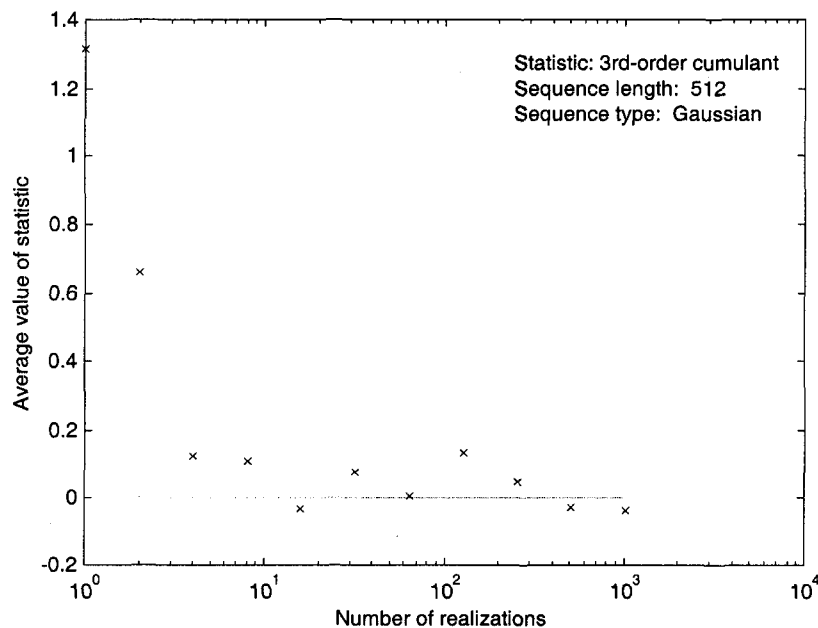


Figure 16. Gaussian distribution, statistic: third-order cumulant.

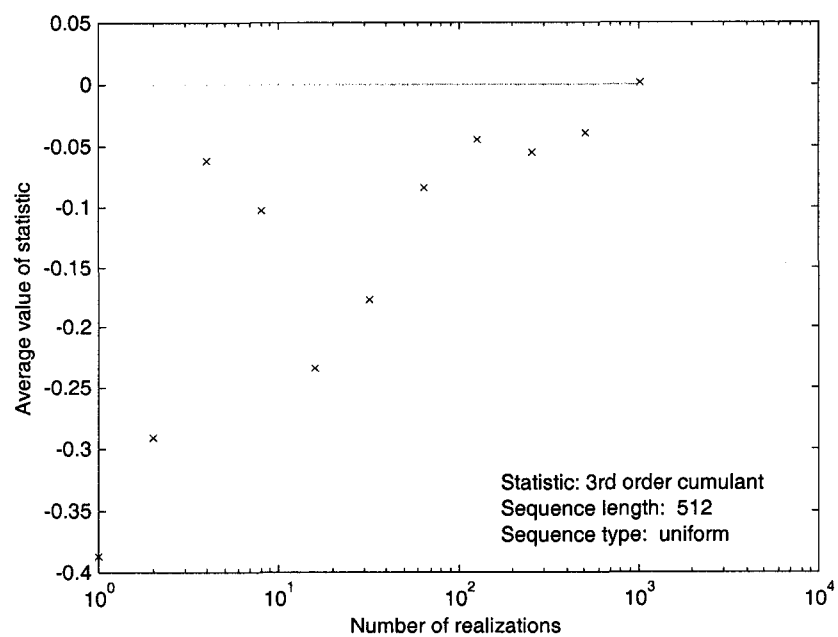


Figure 17. Uniform distribution, statistic: third-order cumulant.

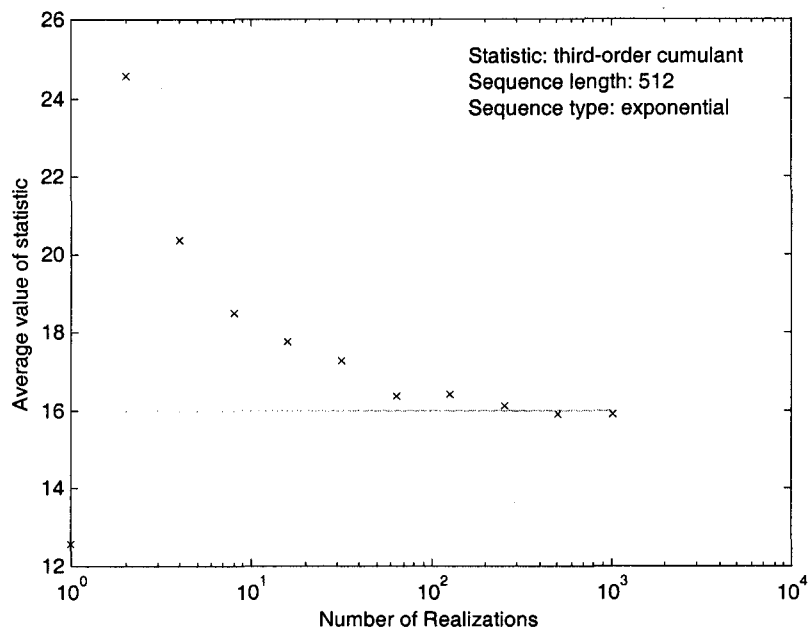


Figure 18. Mean-shifted exponential distribution, statistic: third-order cumulant.

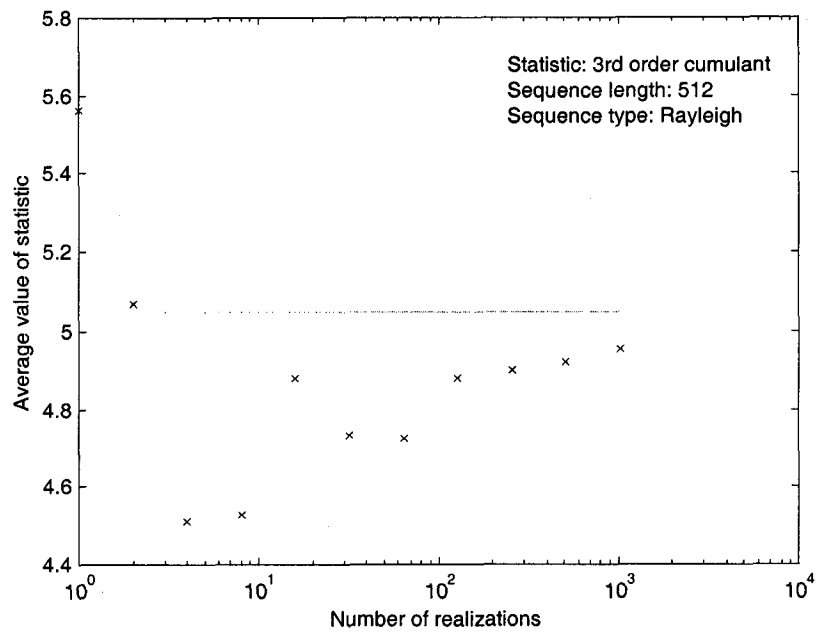


Figure 19. Mean-shifted Rayleigh distribution, statistic: third-order cumulant.

APPENDIX B. PLOTS FOR CHAPTER III

This Appendix contains detailed results of the procedure described in Chapter III. It is divided into multiple Sections, each of which contains *mesh* plots of bispectral estimates of the four distributions for a particular set of *bispeci* parameters. The first Section, for example, contains plots of the bispectral estimates for the four distributions when *bispeci* is given a lag argument of 32 and its DFT length is allowed to remain as its default of 128.

Additionally, the first six Sections contain bispectral estimates of white signals; i.e., those which have not been filtered. In the following eight Sections, the bispectral estimates were generated after the signal had been colored by passing it through the filter of Equation (8).

A. Lag argument = 32, DFT length = 128, white signal

Figures 20 through 23 show the calculated bispectra for the four white signal ensembles (i.e., those that have not been processed by the shaping filter), using a lag argument of 32. All of the other parameters passed to *bispeci* were the defaults, and setting the lag argument to 32 forced the DFT length to 128.

Figure 20 shows the calculated bispectrum of the Gaussian-distributed ensemble. It is flat and has zero amplitude, as would be expected.

Figure 21 shows the calculated bispectrum of the uniform-distributed ensemble. It, too, is flat and has zero amplitude.

Figure 22 shows the calculated bispectrum of the mean-shifted exponentially-distributed ensemble. Also as predicted, this plot has a magnitude of 16, equal to its third-order cumulant (given a signal variance of four). This plot also seems to exhibit more ripple than the previous two.

Figure 23 shows the calculated bispectrum of the mean-shifted Rayleigh-distributed ensemble. As in Figure 22, this bispectrum shows some ripple but no obvious signal characteristics, and has a magnitude of approximately 5, as expected.

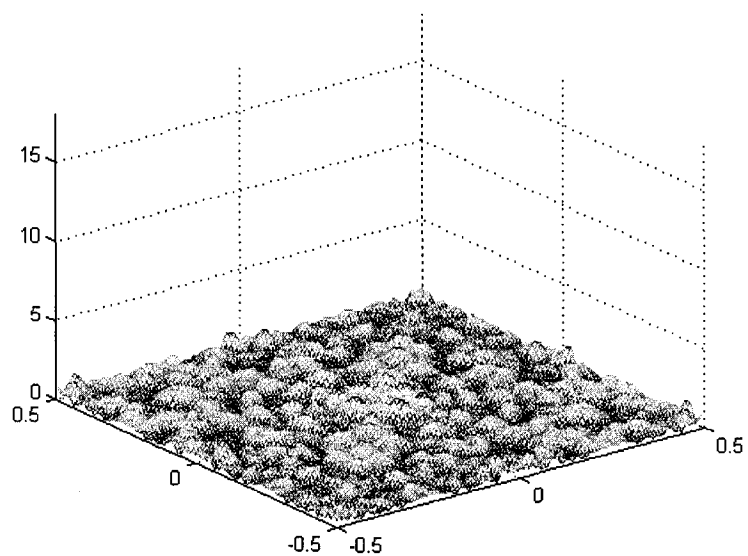


Figure 20 Bispectral estimate of Gaussian distribution, white signal. Estimate was generated with lag argument of 32 and a DFT length of 128.

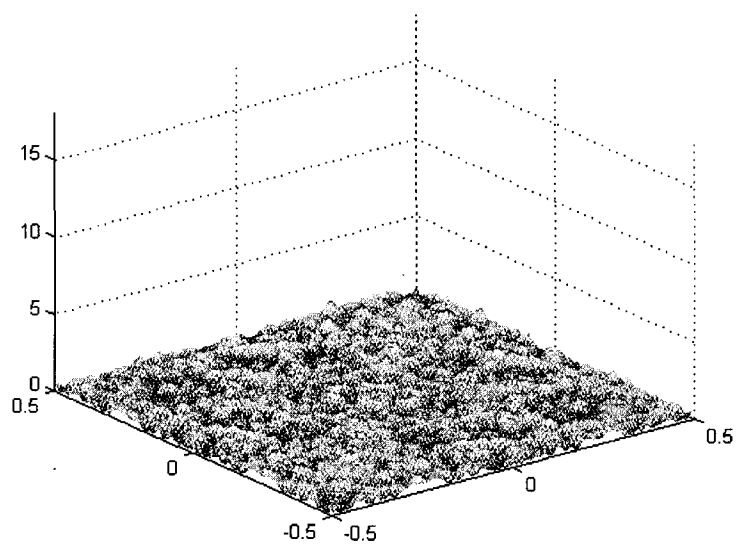


Figure 21 Bispectral estimate of uniform distribution, white signal. Estimate was generated with a lag argument of 32 and a DFT length of 128.

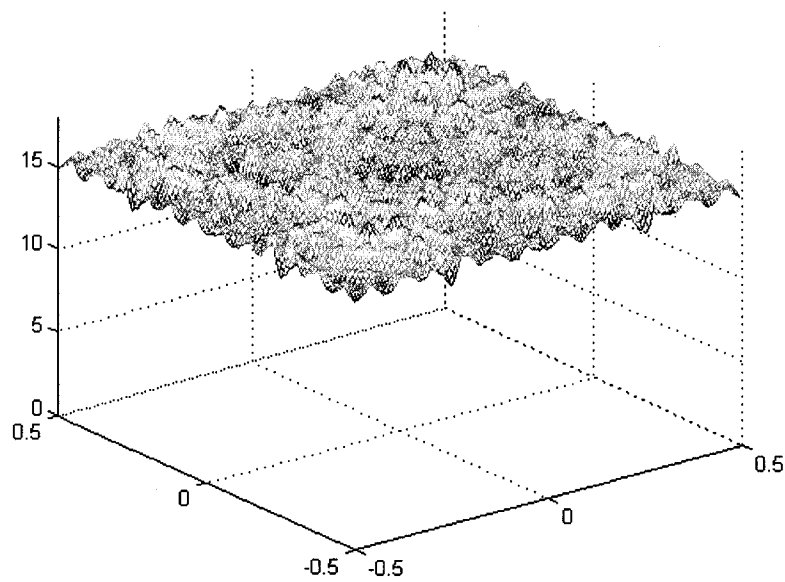


Figure 22 Bispectral estimate of mean-shifted exponential distribution, white signal. Estimate was generated with a lag argument of 32 and a DFT length of 128.

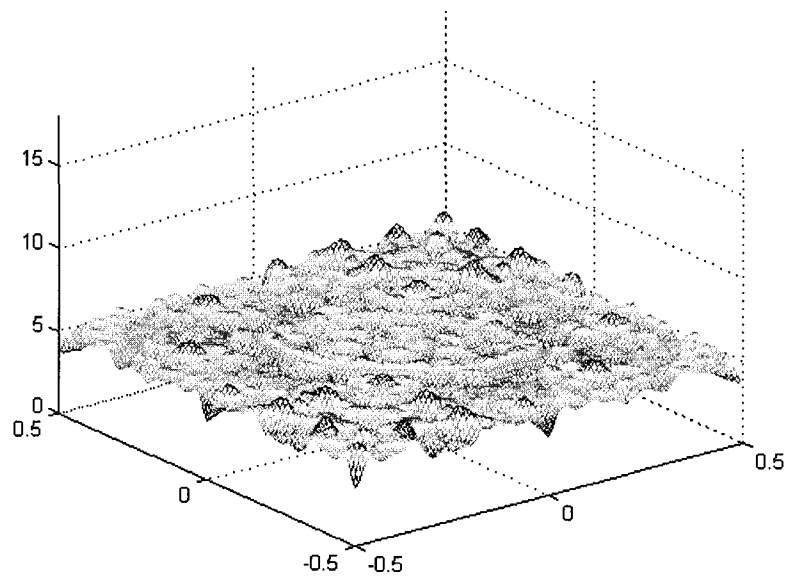


Figure 23 Bispectral estimate of mean-shifted Rayleigh distribution, white signal. Estimate was generated with a lag argument of 32 and a DFT length of 128.

B. Lag argument = 64, DFT length = 256, white signal

Figures 24 through 27 show the calculated bispectra for the four white signal ensembles, using a lag argument of 64. All of the other parameters passed to *bispeci* were the defaults, and setting the lag argument to 64 forced the DFT length to 256.

Figure 24 shows the calculated bispectrum of the Gaussian-distributed ensemble. While still flat and at zero amplitude, it does exhibit more ripple than it did with a lag argument of 32 (compare Figure 20). This is due to the increased variance caused by the use of longer lags.

Figure 25 shows the calculated bispectrum of the uniform-distributed ensemble. It is also flat and has magnitude zero, but has more variance than the bispectrum depicted in Figure 21.

Figure 26 shows the calculated bispectrum of the mean-shifted exponentially-distributed ensemble. The mean level of this signal remains at approximately 16; however, this bispectrum has a much greater variance than any of the other bispectra calculated with these parameters. This is evident in Figure 26, as the plot has a much more jagged appearance than the others.

Figure 27 shows the calculated bispectrum of the mean-shifted Rayleigh-distributed ensemble. This signal amplitude is also where theory says it should be. Also, while the variance is below that of the bispectral estimate of the mean-shifted exponential signal ensemble, it is still noticeably greater than that of either the Gaussian- or uniform-distributed ensemble.

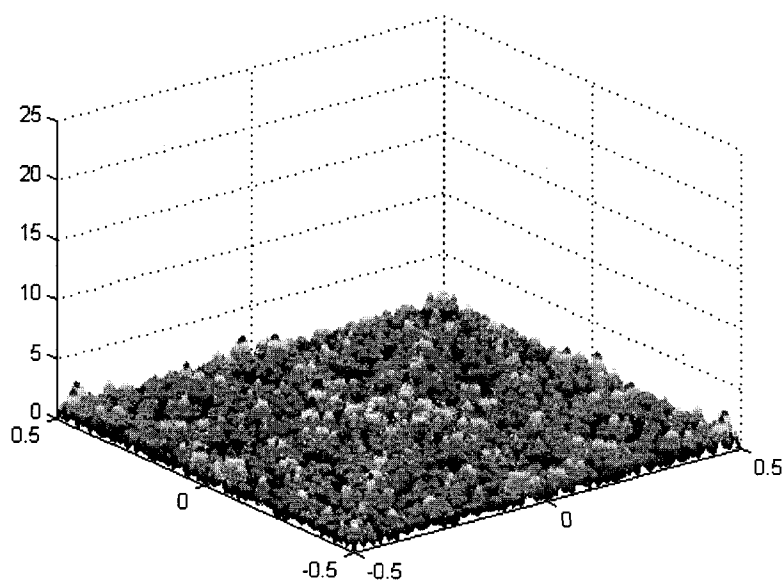


Figure 24 Bispectral estimate of Gaussian distribution, white signal. Estimate was generated with a lag argument of 64 and a DFT length of 256.

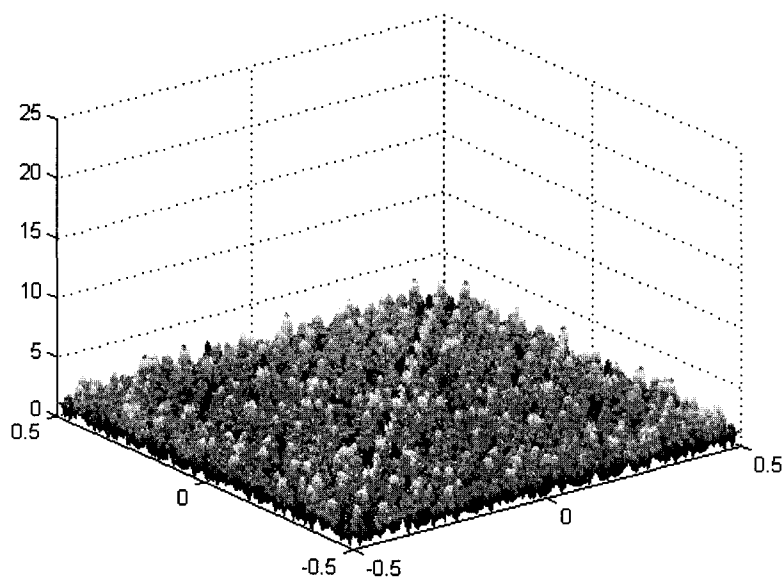


Figure 25 Bispectral estimate of uniform distribution, white signal. Estimate was generated with a lag argument of 64 and a DFT length of 256.

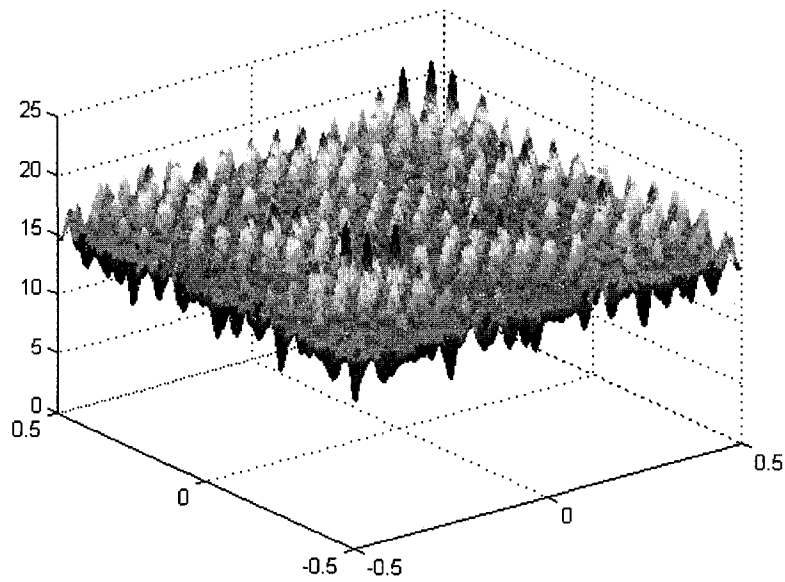


Figure 26 Bispectral estimate of mean-shifted exponential distribution, white signal. Estimate was generated with a lag argument of 64 and a DFT length of 256.

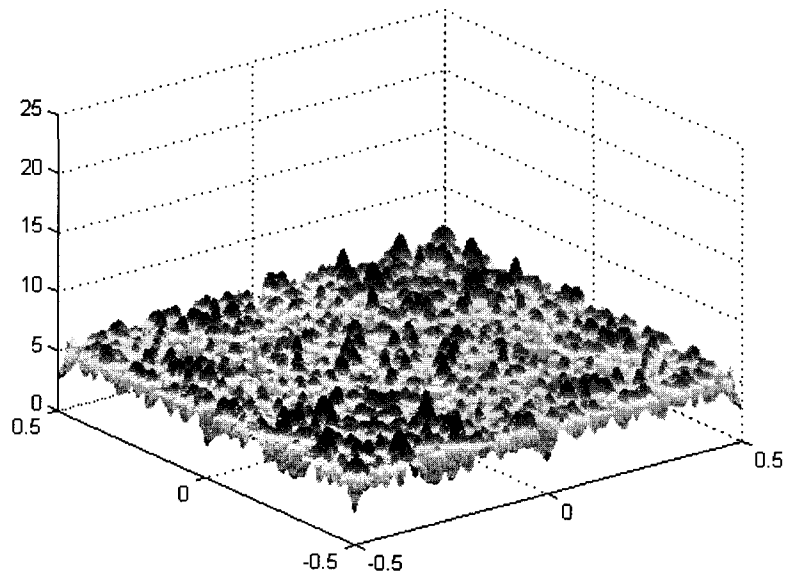


Figure 27 Bispectral estimate of mean-shifted Rayleigh distribution, white signal. Estimate was generated with a lag argument of 64 and a DFT length of 256.

C. Lag argument = 128, DFT length = 512, white signal

Figures 28 through 31 show the calculated bispectra for the four white signal ensembles, using a lag argument of 128. All of the other parameters passed to *bispeci* were the defaults, and setting the lag argument to 128 forced the DFT length to 512.

Figure 28 shows the calculated bispectrum of the Gaussian-distributed ensemble. The signal floor remains at zero amplitude; however, as the variance has continued to increase, its effect has continued to be evident in the plots.

Figure 29 shows the calculated bispectrum of the uniform-distributed ensemble. As in the case of the Gaussian-distributed signal, the floor remains at zero amplitude. Also as in the case of the Gaussian-distributed signal, the variance has obviously increased from previous estimates; compare Figure 25.

Figure 30 shows the calculated bispectrum of the mean-shifted exponentially-distributed ensemble. The large variance of the estimate is even more evident than in previous plots.

Figure 31 shows the calculated bispectrum of the mean-shifted Rayleigh-distributed ensemble. The signal amplitude remains at approximately 5, where theory says it should be. Also as before, this bispectral estimate shows considerably less variance than that of the mean-shifted exponential ensemble (Figure 30), but considerably more than those of the Gaussian- and uniform-distributed ensembles (Figures 28 and 29, respectively).

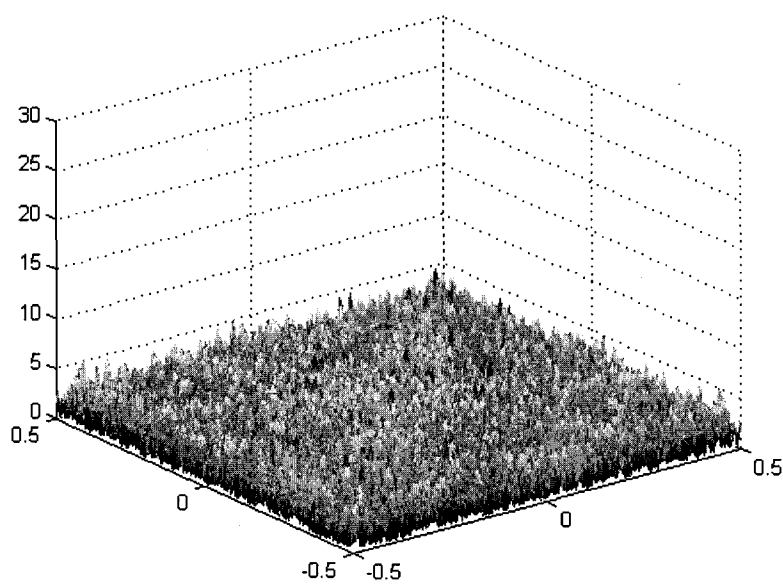


Figure 28 Bispectral estimate of Gaussian distribution, white signal. Estimate was generated with a lag argument of 128 and a DFT length of 512.

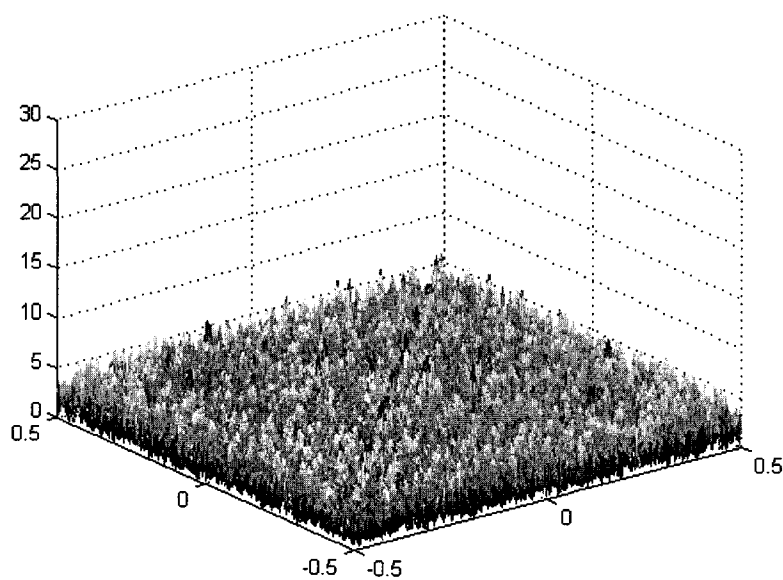


Figure 29 Bispectral estimate of uniform distribution, white signal. Estimate was generated with a lag argument of 128 and a DFT length of 512.

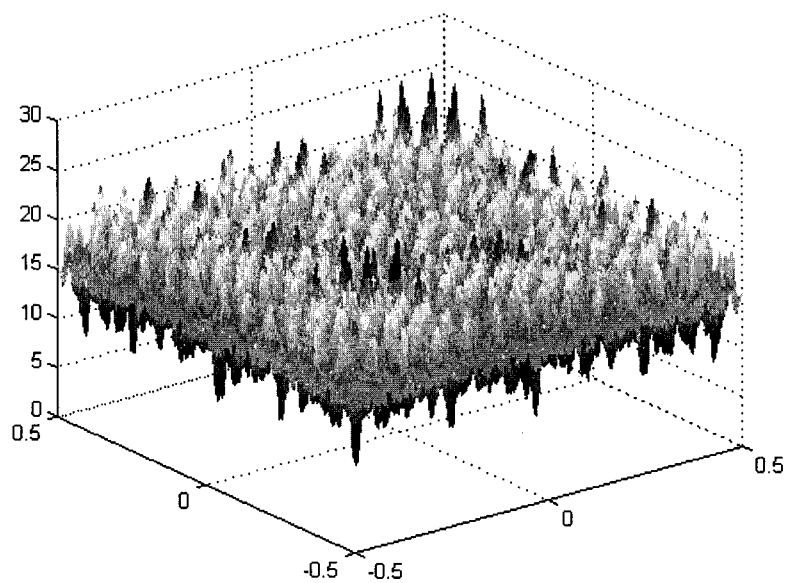


Figure 30 Bispectral estimate of mean-shifted exponential distribution, white signal. Estimate was generated with a lag argument of 128 and a DFT length of 512.

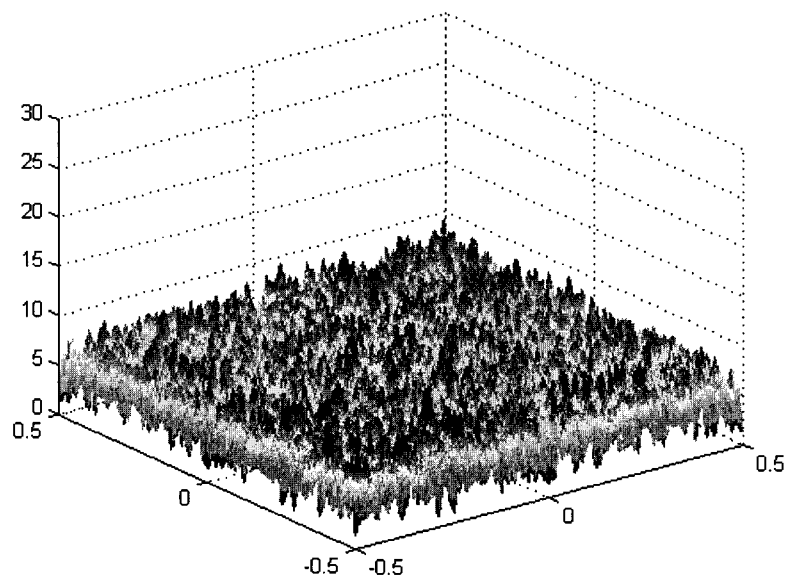


Figure 31 Bispectral estimate of mean-shifted Rayleigh distribution, white signal. Estimate was generated with a lag argument of 128 and a DFT length of 512.

D. Lag argument = 256, DFT length = 1024, white signal

Figures 32 through 35 show the calculated bispectra for the four white signal ensembles using a lag argument of 256. All of the other parameters passed to *bispeci* were the defaults, and setting the lag argument to 256 forced the DFT length to 1024.

Figure 32 shows the calculated bispectrum of the Gaussian-distributed ensemble. The signal floor remains at zero amplitude. The peaks are not as evident as before (compare Figure 28), although the variance has continued to increase. The increase in estimate variance is evident in the fact that the noise "floor" gets thicker and thicker as the lag argument is increased.

Figure 33 shows the calculated bispectrum of the uniform-distributed ensemble. The signal floor also remains at zero amplitude and, as in Figure 32, the peaks that were present in the previous estimate (Figure 29) seem to have disappeared.

Figure 34 shows the calculated bispectrum of the mean-shifted exponentially-distributed ensemble. As usual, the bispectral estimate of the mean-shifted exponential signal ensemble shows a much greater variance than that of the other three distributions. Note that, as with the other three bispectral plots, the "floor" of this signal is thicker (although still centered at a magnitude of 16).

Figure 35 shows the calculated bispectrum of the mean-shifted Rayleigh-distributed ensemble. The bispectral amplitude remains at approximately 5, although it is more difficult to tell visually due to the increased variance in the estimate.

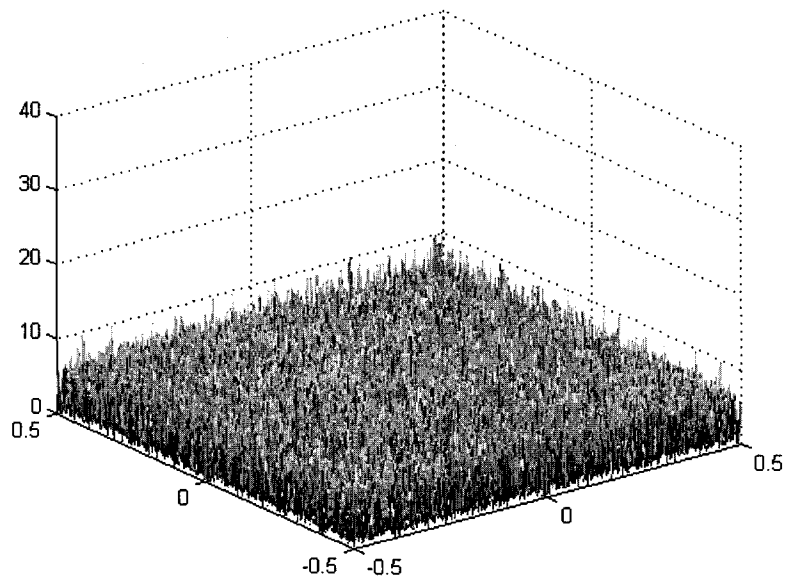


Figure 32 Bispectral estimate of Gaussian distribution, white signal. Estimate was generated with a lag argument of 256 and a DFT length of 1024.

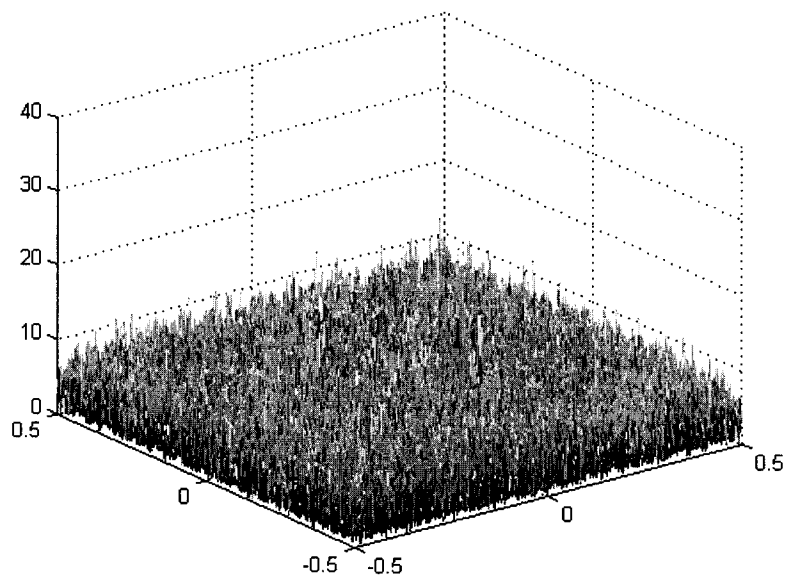


Figure 33 Bispectral estimate of uniform distribution, white signal. Estimate was generated with a lag argument of 256 and a DFT length of 1024.

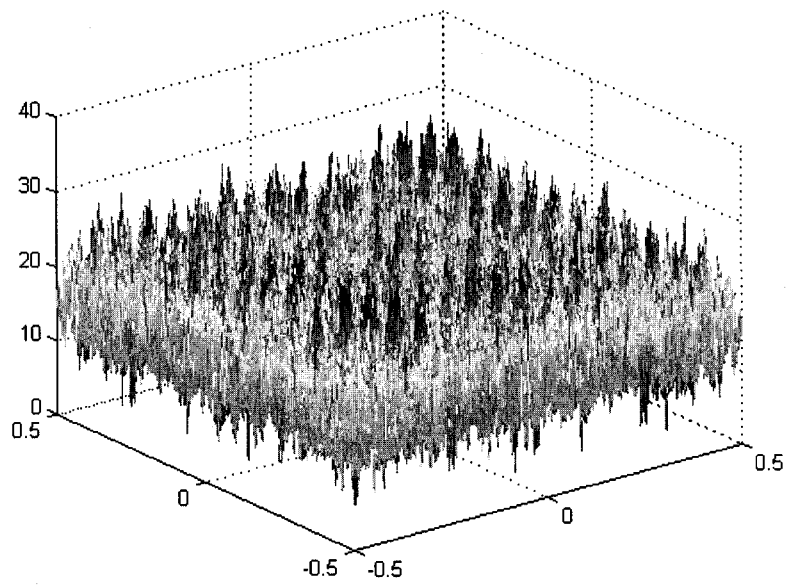


Figure 34 Bispectral estimate of mean-shifted exponential distribution, white signal. Estimate was generated with a lag argument of 256 and a DFT length of 1024.

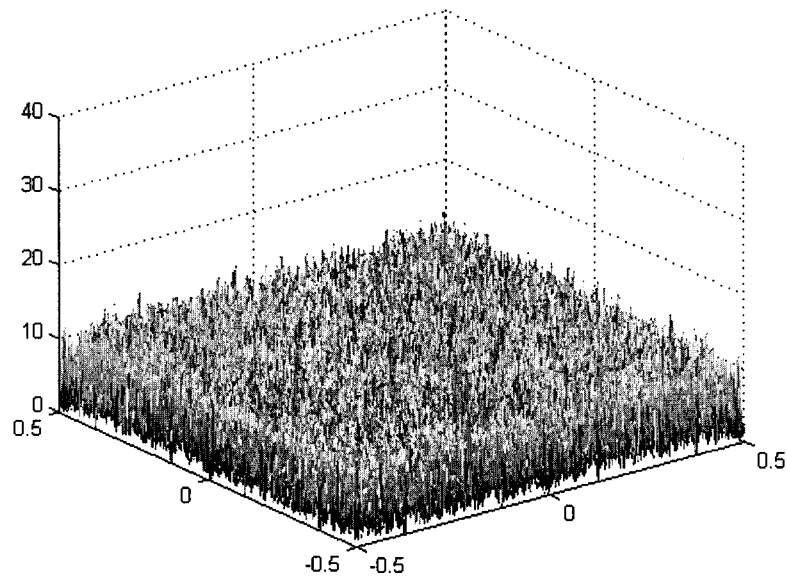


Figure 35 Bispectral estimate of mean-shifted Rayleigh distribution, white signal. Estimate was generated with a lag argument of 256 and a DFT length of 1024.

E. Lag argument = 512, DFT length = 2048, white signal

Figures 36 through 39 show the calculated bispectra for the four white signal ensembles, using a lag argument of 512. All of the other parameters passed to *bispeci* were the defaults, and setting the lag argument to 512 forced the DFT length to 2048.

Figure 36 shows the calculated bispectrum of the Gaussian-distributed ensemble. This bispectral plot does not seem to be significantly different from that produced by 256 lags (i.e., Figure 32).

Figure 37 shows the calculated bispectrum of the uniform-distributed ensemble. As with the Gaussian-distributed signal, there is no noticeable difference between the bispectrum produced with 512 lags (i.e., Figure 37), and that produced with 256 lags (i.e., Figure 33).

Figure 38 shows the calculated bispectrum of the mean-shifted exponentially-distributed ensemble. Although the variance is still very large, the increased DFT size appears to have smoothed the bispectrum somewhat, so that the large peaks are not as evident as they are in (for example) Figure 34.

Figure 39 shows the calculated bispectrum of the mean-shifted Rayleigh-distributed ensemble. As with the other three bispectral plots in this set, the plot of this bispectral estimate shows no distinctive peaks of energy.

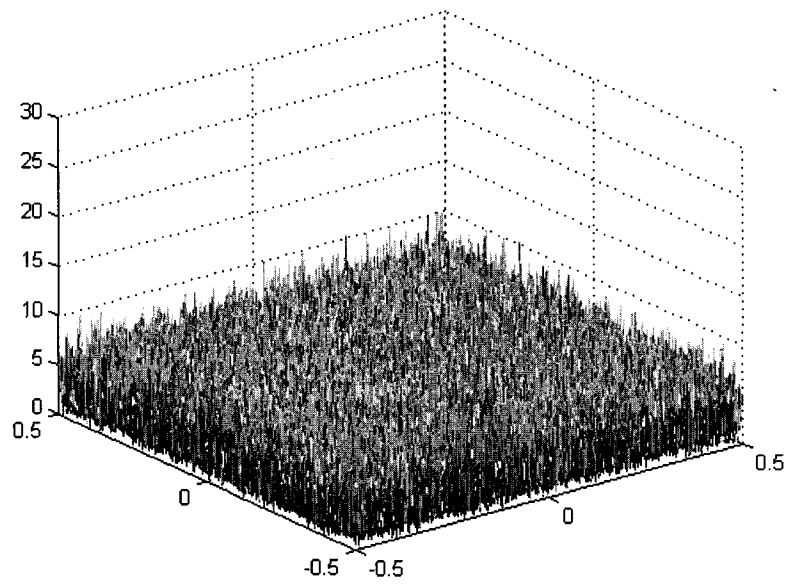


Figure 36 Bispectral estimate of Gaussian distribution, white signal. Estimate was generated with a lag argument of 512 and a DFT length of 2048.

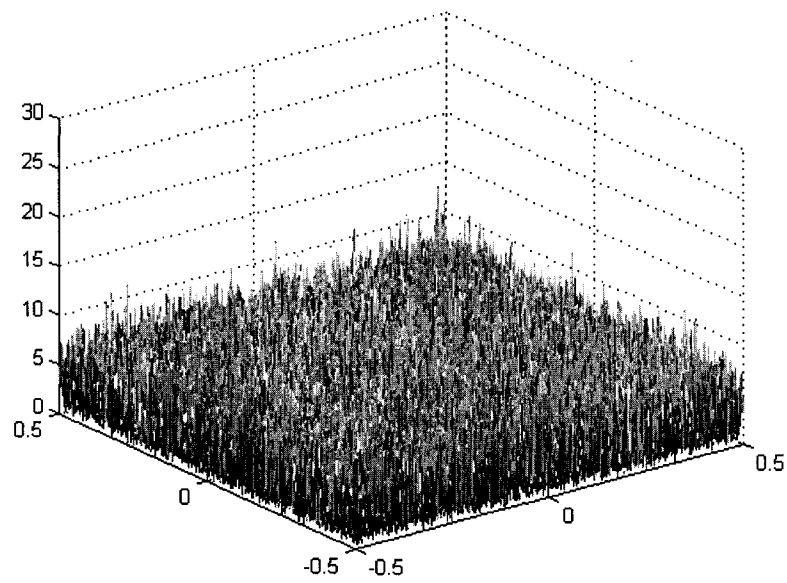


Figure 37 Bispectral estimate of uniform distribution, white signal. Estimate was generated with a lag argument of 512 and a DFT length of 2048.

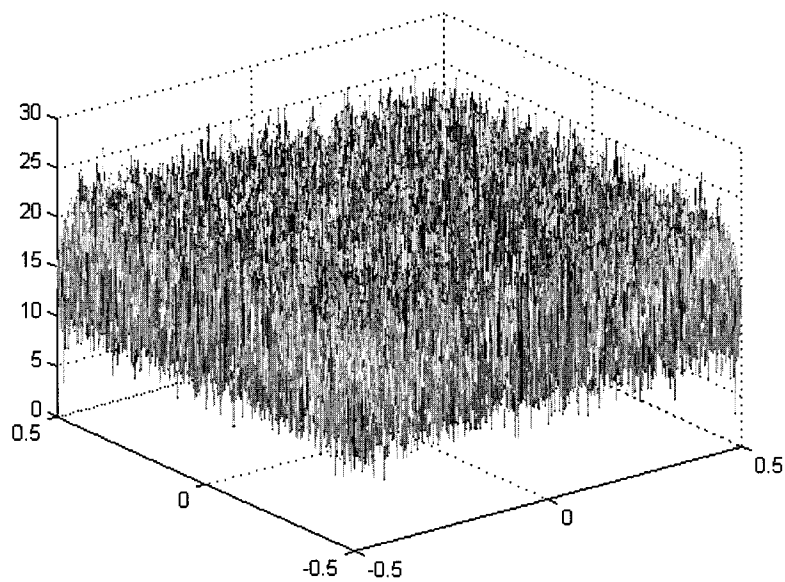


Figure 38 Bispectral estimate of mean-shifted exponential distribution, white signal. Estimate was generated with a lag argument of 512 and a DFT length of 2048.

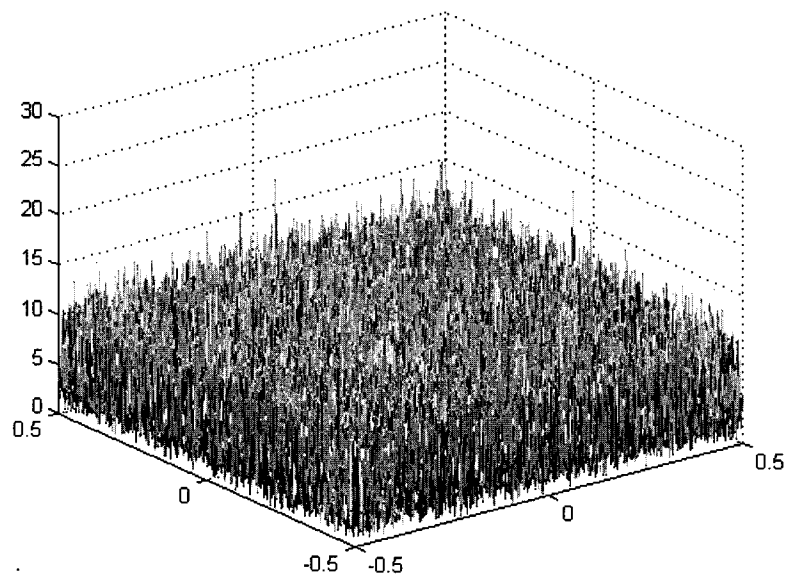


Figure 39 Bispectral estimate of mean-shifted Rayleigh distribution, white signal. Estimate was generated with a lag argument of 512 and a DFT length of 2048.

F. Lag argument = 256, DFT length = 2048, white signal

Figures 40 through 43 show the calculated bispectra for the four white signal ensembles. In this case, the lag argument has been returned to 256. However, the DFT length has been increased to a length of 2048. (Had it not been explicitly set, the DFT length would have defaulted to 1024 points.) All of the other parameters passed to *bispeci* remained as the defaults.

Figure 40 shows the calculated bispectrum of the Gaussian-distributed ensemble. Returning the lag argument to 256 seems to have brought back a few bins of energy poking up above the noise floor.

Figure 41 shows the calculated bispectrum of the uniform-distributed ensemble. While this signal floor remains at zero amplitude, the peak energy seen here is higher than that of the Gaussian-distributed signal.

Figure 42 shows the calculated bispectrum of the mean-shifted exponentially-distributed ensemble. Returning the lag argument to 256 has caused the signal peaks in this bispectral display to become obvious again (compare Figures 39 and 35).

Figure 43 shows the calculated bispectrum of the mean-shifted Rayleigh-distributed ensemble. The increased DFT size seems to have drawn signal energy of some sort up out of the floor in this case as well.

Judging solely by the numbers in Table 5, the results depicted in Figures 40 through 43 should be largely indistinguishable from those seen in Figures 32 through 35. This prediction seems largely borne out by a cursory visual examination of the plots. This fact, coupled with the results of Table 5, suggests that the DFT length is largely irrelevant to the mean of the bispectral estimate of a white signal.

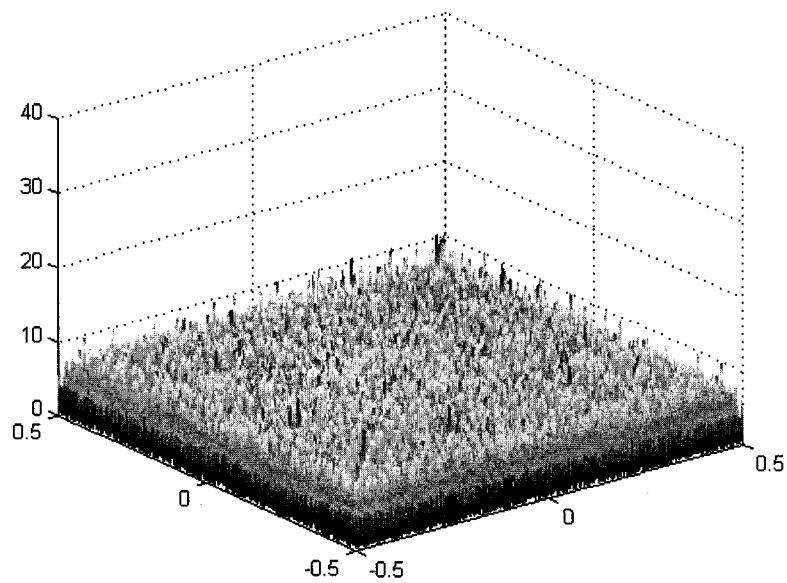


Figure 40 Bispectral estimate of Gaussian distribution, white signal. Estimate was generated with a lag argument of 256 and a DFT length of 2048.

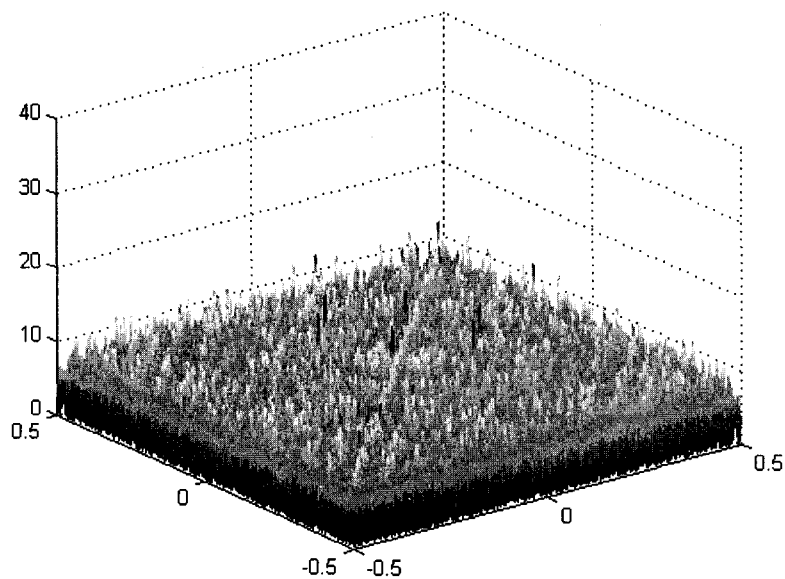


Figure 41 Bispectral estimate of uniform distribution, white signal. Estimate was generated with a lag argument of 256 and a DFT length of 2048.

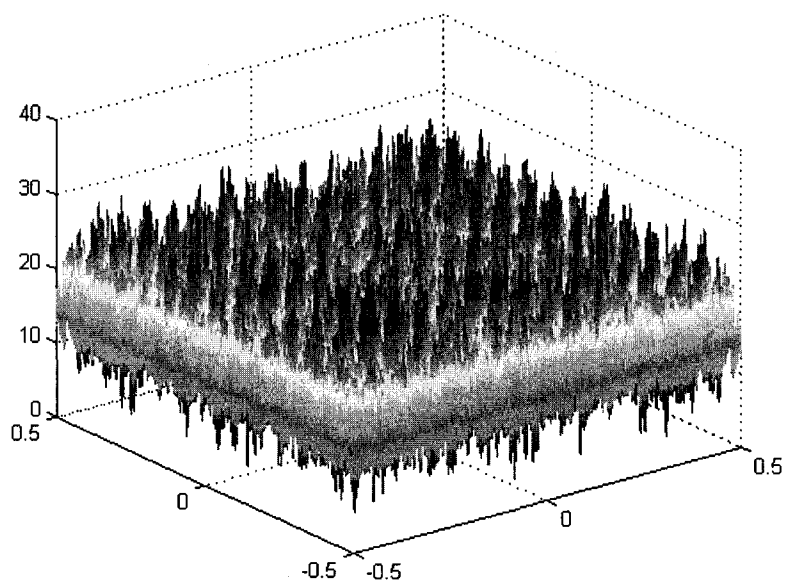


Figure 42 Bispectral estimate of mean-shifted exponential distribution, white signal. Estimate was generated with a lag argument of 256 and a DFT length of 2048.

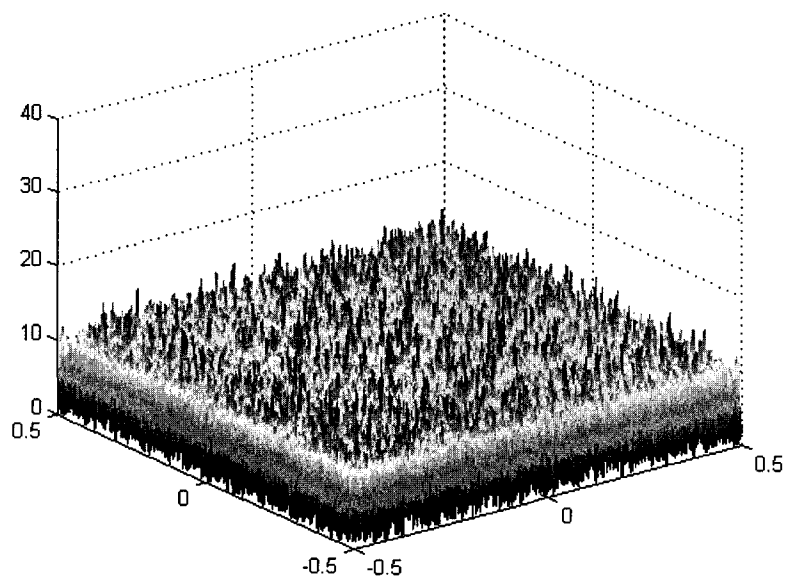


Figure 43 Bispectral estimate of mean-shifted Rayleigh distribution, white signal. Estimate was generated with a lag argument of 256 and a DFT length of 2048.

G. Lag argument = 32, DFT length = 128, colored signal ($\beta = 0.25$)

Figures 44 through 47 show the bispectral estimates of the datasets after they have been passed through the filter of Equation (8) with a parameter β of 0.25.

In this case, the bispectrum has been estimated with *bispeci*, which has been given a lag argument of 32 and an DFT length of 128. All the other *bispeci* parameters have been allowed to remain as their defaults.

Figure 44 shows the calculated bispectrum of the colored Gaussian-distributed ensemble. The filter does not appear to have had a significant effect, as the bispectrum is flat and has zero magnitude.

Figure 45 shows the calculated bispectrum of the colored uniform-distributed data ensemble. It too, as predicted, has zero magnitude and is flat.

Figure 46 shows the bispectral estimate of the mean-shifted exponential signal ensemble, after the signal has been processed by the filter. It shows that the filter has indeed shaped the bispectrum, as the signal peaks well up from its third-order cumulant value of 16. It is instructive to compare Figure 45 with Figure 21 which shows the plot of the identical (but unfiltered) signal ensemble, generated with the same parameters passed to *bispeci*.

Figure 47 shows the bispectral estimate of the mean-shifted Rayleigh signal ensemble, after the signal has been processed by the filter. As with the exponentially-distributed signal ensemble, the filter has obviously shaped the bispectrum here as well, although it is not as pronounced as in the case of the mean-shifted exponential distribution. Again, comparison is instructive. Figure 23 shows the (flat) bispectrum produced by the same signal ensemble, with the same parameters passed to *bispeci*. But where Figure 23 is flat, Figure 47 shows definite signal energy. The floor of this estimate is just below its theoretical value of 5.04 as well.

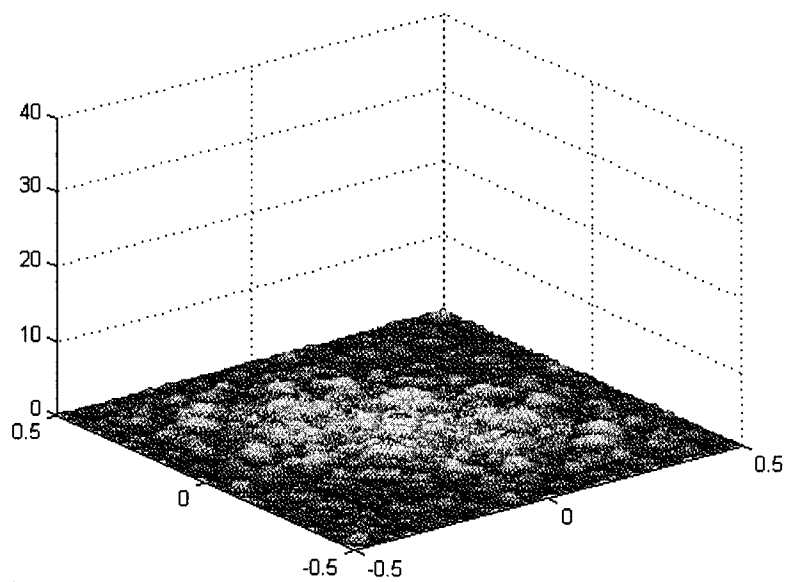


Figure 44 Bispectral estimate of Gaussian distribution, colored signal ($\beta = 0.25$).
Estimate was generated with a lag argument of 32 and a DFT length of 128.

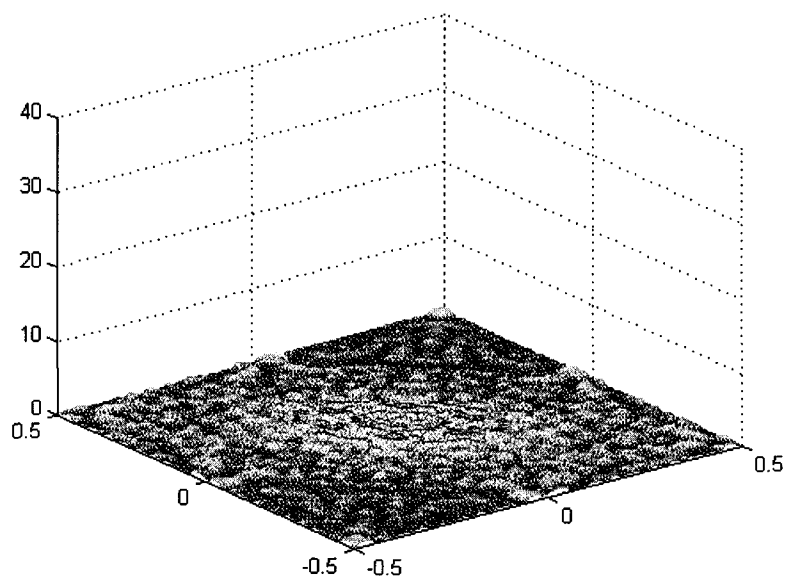


Figure 45 Bispectral estimate of uniform distribution, colored signal ($\beta = 0.25$).
Estimate was generated with a lag argument of 32 and a DFT length of 128.

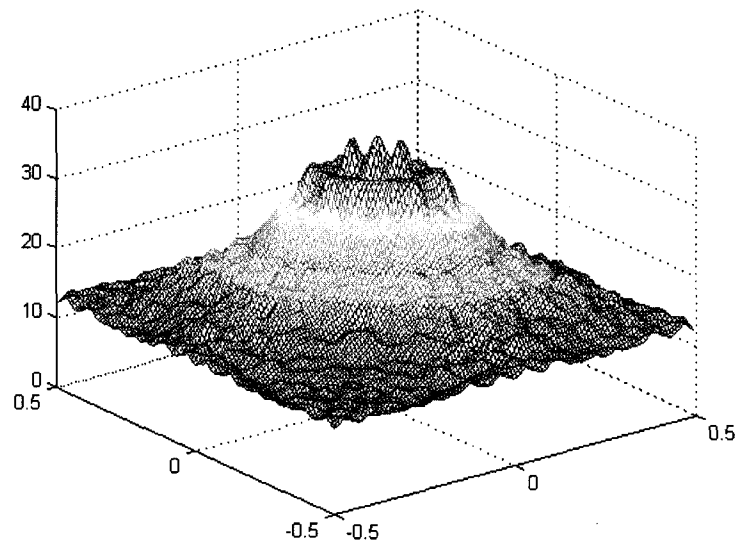


Figure 46 Bispectral estimate of mean-shifted exponential distribution, colored signal ($\beta = 0.25$). Estimate was generated with a lag argument of 32 and a DFT length of 128.

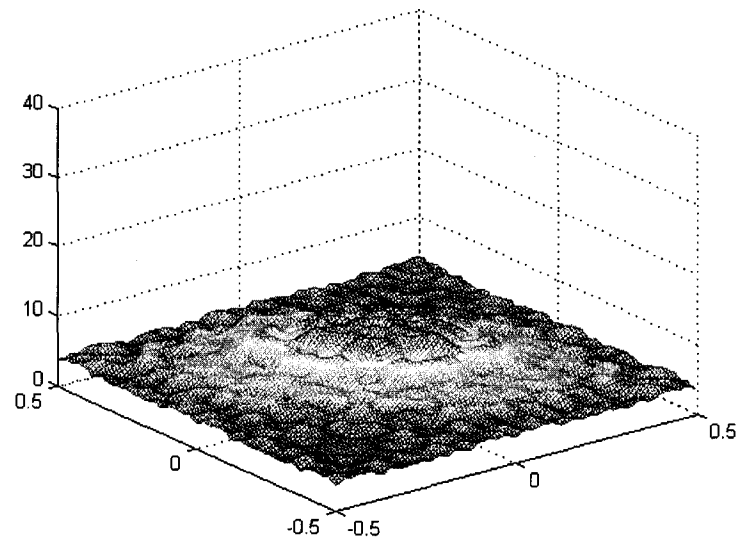


Figure 47 Bispectral estimate of mean-shifted Rayleigh distribution, colored signal ($\beta = 0.25$). Estimate was generated with a lag argument of 32 and a DFT length of 128.

H. Lag argument = 64, DFT length = 256, colored signal ($\beta = 0.25$)

Figures 48 through 51 show the bispectral estimates of the colored signals generated when *bispeci* has been given a lag argument of 64 and a DFT length of 256. All the other *bispeci* parameters have been allowed to remain as their defaults.

Figure 48 shows the calculated bispectrum of the colored Gaussian-distributed ensemble. The filter does not appear to have had a significant effect, as the bispectrum is flat and has zero magnitude.

Figure 49 shows the calculated bispectrum of the colored uniform-distributed data ensemble. It too, as predicted, has zero magnitude and is flat, although a little ripple is evident.

Figure 50 shows the bispectral estimate of the mean-shifted exponential signal ensemble, after the signal has been processed by the filter. It shows that the filter has shaped this bispectrum as well, as the signal peaks well up from its third-order cumulant value of 16.

Figure 51 shows the bispectral estimate of the mean-shifted Rayleigh signal ensemble, after the signal has been processed by the filter. As with the exponentially-distributed signal ensemble, the filter has obviously shaped the bispectrum here as well, although again, it is not as pronounced as in the case of the mean-shifted exponential distribution.

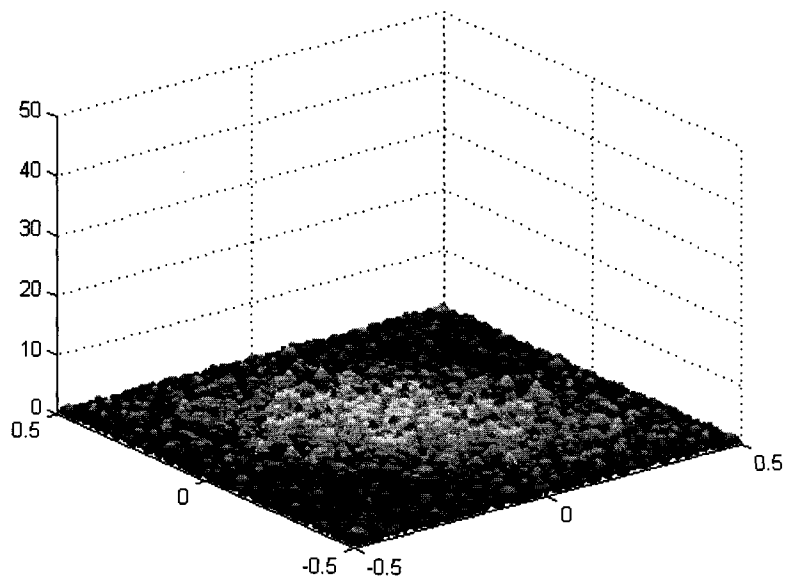


Figure 48 Bispectral estimate of Gaussian distribution, colored signal ($\beta = 0.25$).
 Estimate was generated with a lag argument of 64 and a DFT length of 256.

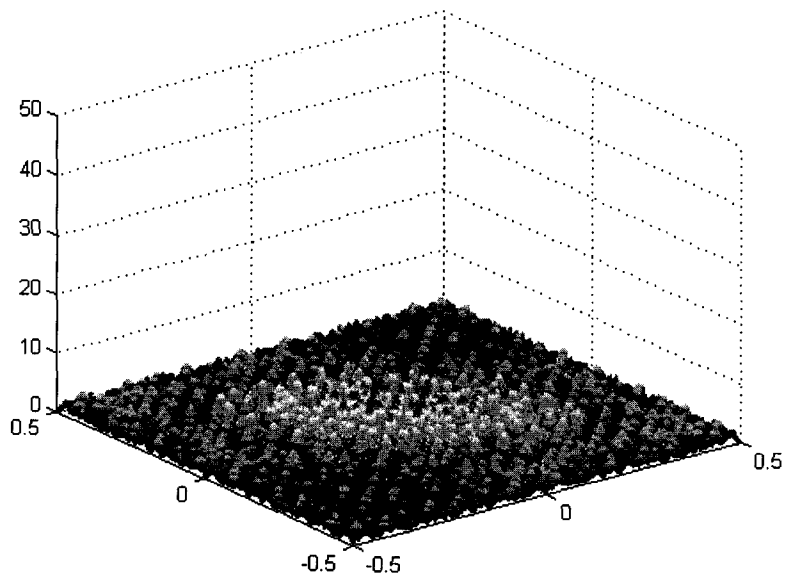


Figure 49 Bispectral estimate of uniform distribution, colored signal ($\beta = 0.25$).
 Estimate was generated with a lag argument of 64 and a DFT length of 256.

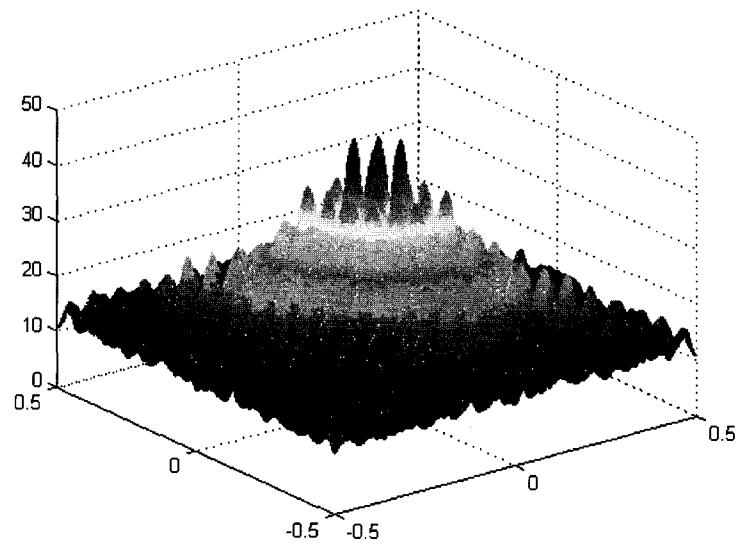


Figure 50 Bispectral estimate of mean-shifted exponential distribution, colored signal ($\beta = 0.25$). Estimate was generated with a lag argument of 64 and a DFT length of 256.

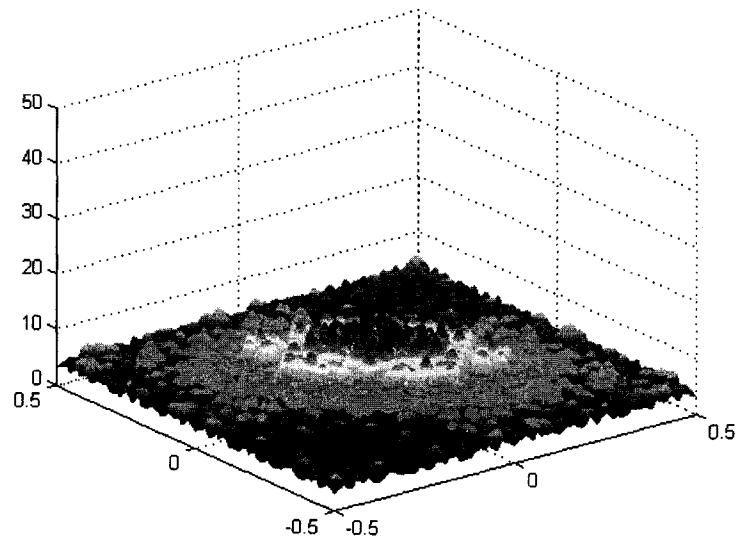


Figure 51 Bispectral estimate of mean-shifted Rayleigh distribution, colored signal ($\beta = 0.25$). Estimate was generated with a lag argument of 64 and a DFT length of 256.

I. Lag argument = 128, DFT length = 512, colored signal ($\beta = 0.25$)

Figures 52 through 55 show the bispectral estimates of the colored signals generated when *bispeci* has been given a lag argument of 128 and a DFT length of 512. All the other *bispeci* parameters have been allowed to remain as their defaults.

Figure 52 shows the calculated bispectrum of the colored Gaussian-distributed ensemble. The bispectrum of the Gaussian ensemble, as before, remains flat and has zero magnitude.

Figure 53 shows the calculated bispectrum of the colored uniform-distributed data ensemble. It too, as predicted, has zero magnitude and is flat.

Figure 54 shows the bispectral estimate of the mean-shifted exponential signal ensemble, after the signal has been processed by the filter. It shows that the filter has significantly shaped this bispectrum.

Figure 55 shows the bispectral estimate of the mean-shifted Rayleigh signal ensemble, after the signal has been processed by the filter. As with the exponentially-distributed signal ensemble, the filter has obviously shaped the bispectrum here as well, although as before, it is not as pronounced as in the case of the mean-shifted exponential distribution.

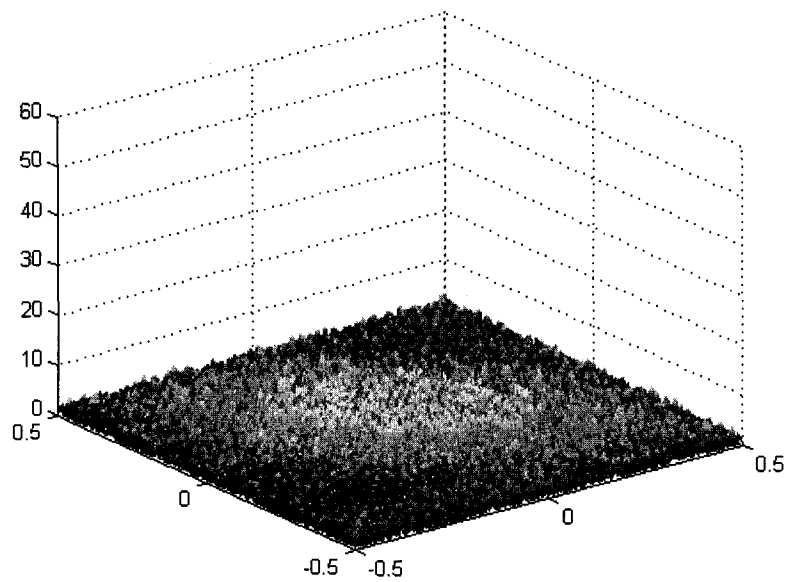


Figure 52 Bispectral estimate of Gaussian distribution, colored signal ($\beta = 0.25$). Estimate was generated with a lag argument of 128 and a DFT length of 512.

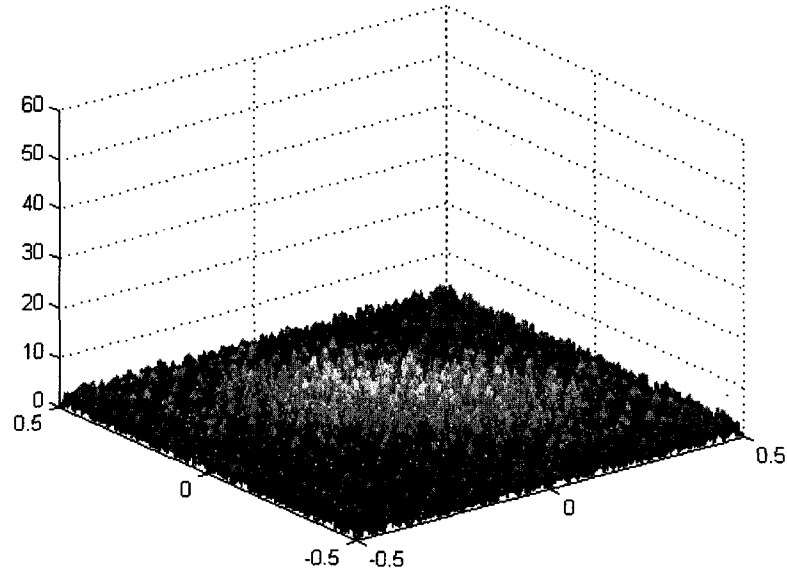


Figure 53 Bispectral estimate of uniform distribution, colored signal ($\beta = 0.25$). Estimate was generated with a lag argument of 128 and a DFT length of 512.

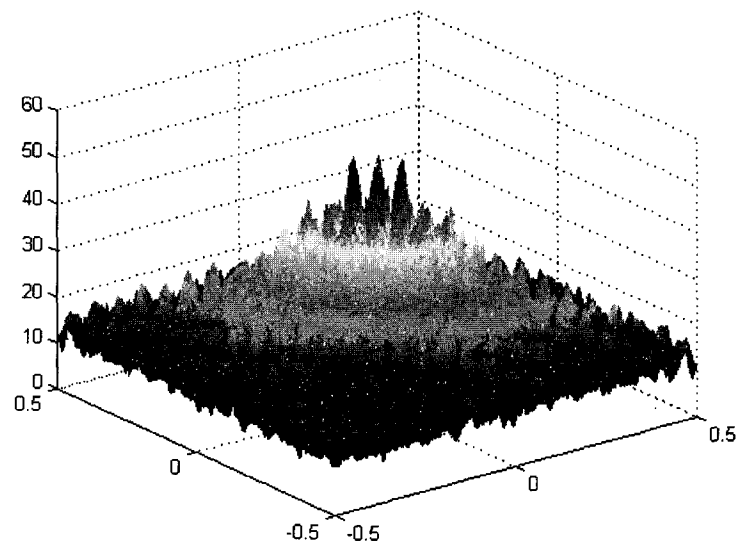


Figure 54 Bispectral estimate of mean-shifted exponential distribution, colored signal ($\beta = 0.25$). Estimate was generated with a lag argument of 128 and a DFT length of 512.

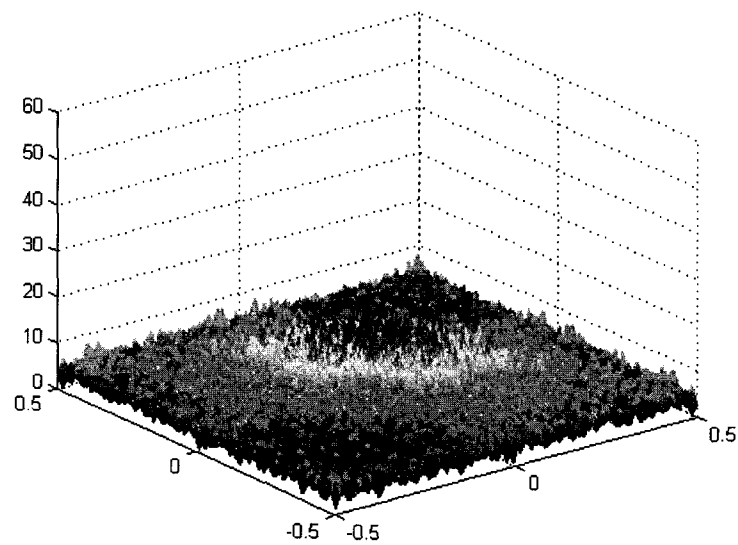


Figure 55 Bispectral estimate of mean-shifted Rayleigh distribution, colored signal ($\beta = 0.25$). Estimate was generated with a lag argument of 128 and a DFT length of 512.

J. Lag argument = 256, DFT length = 1024, colored signal ($\beta = 0.25$)

Figures 56 through 59 show the bispectral estimates of the colored signals generated when *bispeci* has been given a lag argument of 256 and a DFT length of 1024. All the other *bispeci* parameters have been allowed to remain as their defaults.

Figure 56 shows the calculated bispectrum of the colored Gaussian-distributed ensemble. The bispectrum of this dataset remains flat and has zero magnitude.

Figure 57 shows the calculated bispectrum of the colored uniform-distributed data ensemble. It too, as predicted, has zero magnitude and is flat, although a little ripple is evident.

Figure 58 shows the bispectral estimate of the mean-shifted exponential signal ensemble, after the signal has been processed by the filter. The peaks seen in previous plots (compare Figure 54) seem less distinct in this plot, possibly as a result of the increased variance due to increased data length. It is still apparent that the filter has shaped the signal, though, especially in comparison with Figures 56 and 57.

Figure 59 shows the bispectral estimate of the mean-shifted Rayleigh signal ensemble, after the signal has been processed by the filter. The effects of the filtering are less pronounced here as well, although still evident.

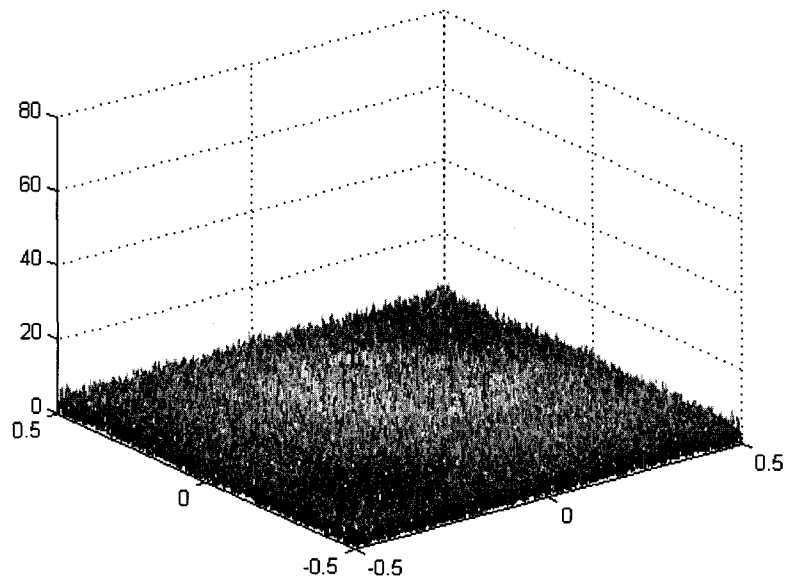


Figure 56 Bispectral estimate of Gaussian distribution, colored signal ($\beta = 0.25$). Estimate was generated with a lag argument of 256 and a DFT length of 1024.

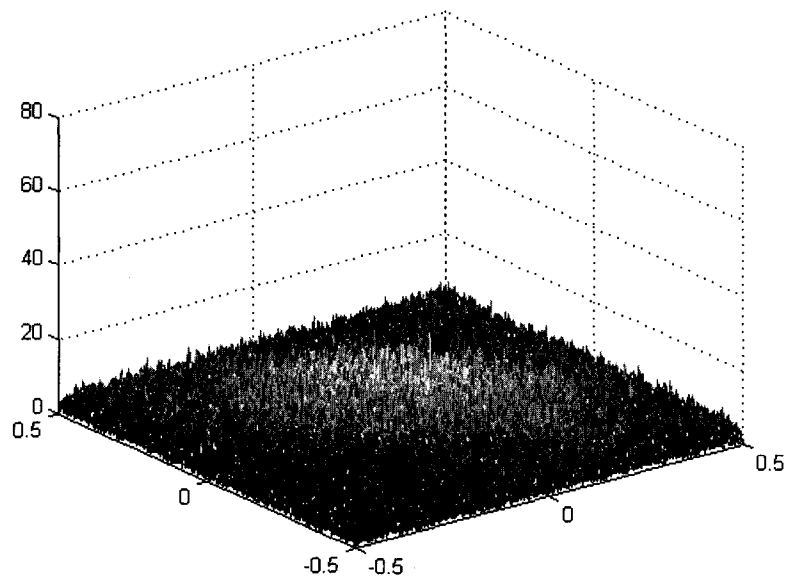


Figure 57 Bispectral estimate of uniform distribution, colored signal ($\beta = 0.25$). Estimate was generated with a lag argument of 256 and a DFT length of 1024.

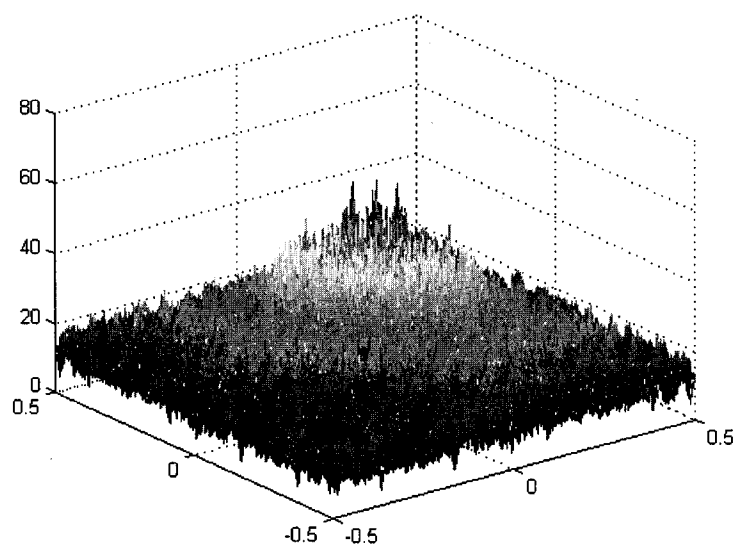


Figure 58 Bispectral estimate of mean-shifted exponential distribution, colored signal ($\beta = 0.25$). Estimate was generated with a lag argument of 256 and a DFT length of 1024.

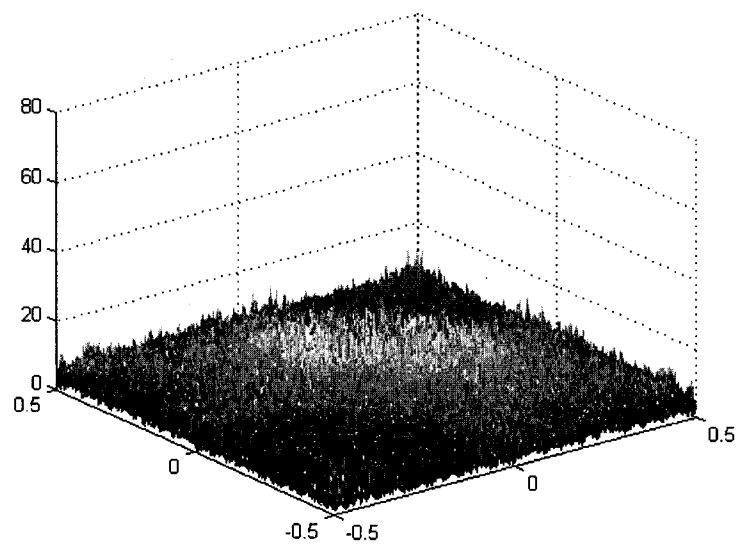


Figure 59 Bispectral estimate of mean-shifted Rayleigh distribution, colored signal ($\beta = 0.25$). Estimate was generated with a lag argument of 256 and a DFT length of 1024.

K. Lag argument = 512, DFT length = 2048, colored signal ($\beta = 0.25$)

Figures 60 through 63 show the bispectral estimates of the colored signals generated when *bispeci* has been given a lag argument of 512 and a DFT length of 2048. All the other *bispeci* parameters have been allowed to remain as their defaults.

Figure 60 shows the calculated bispectrum of the colored Gaussian-distributed ensemble. This signal shows what appear to be bins of energy; these are artifacts of the increased variance due to the greater signal length.

Figure 61 shows the calculated bispectrum of the colored uniform-distributed data ensemble. The increased variance is even more evident here than in Figure 60, although the signal overall remains at zero and is essentially flat.

Figure 62 shows the bispectral estimate of the mean-shifted exponential signal ensemble, after the signal has been processed by the filter. The greater variance continues to swamp the shaping effects of the filter.

Figure 63 shows the bispectral estimate of the mean-shifted Rayleigh signal ensemble, after the signal has been processed by the filter. The signal energy, which should be evident, is here almost invisible as the variance causes the plot to become more and more jagged.

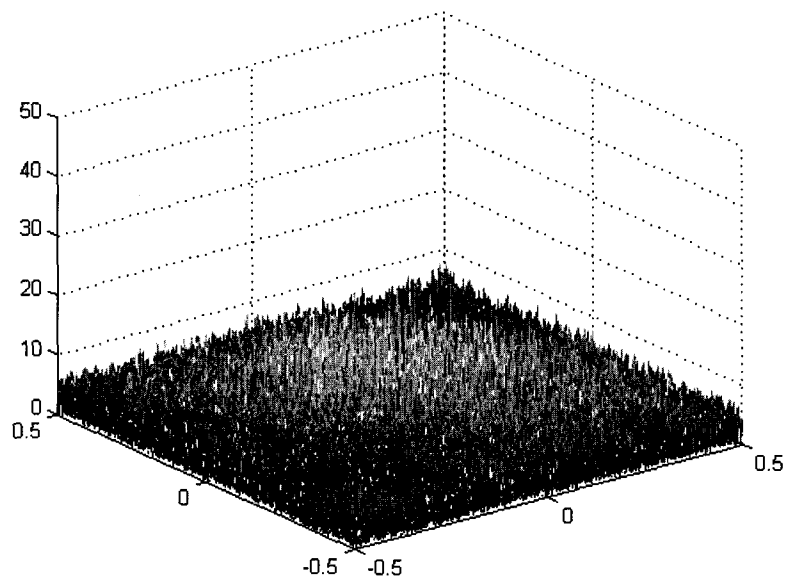


Figure 60 Bispectral estimate of Gaussian distribution, colored signal ($\beta = 0.25$). Estimate was generated with a lag argument of 512 and a DFT length of 2048.

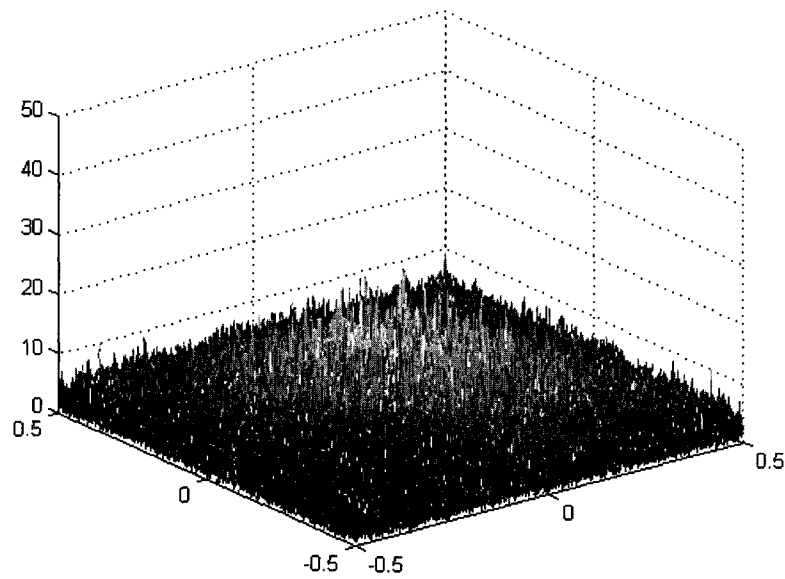


Figure 61 Bispectral estimate of uniform distribution, colored signal ($\beta = 0.25$). Estimate was generated with a lag argument of 512 and a DFT length of 2048.

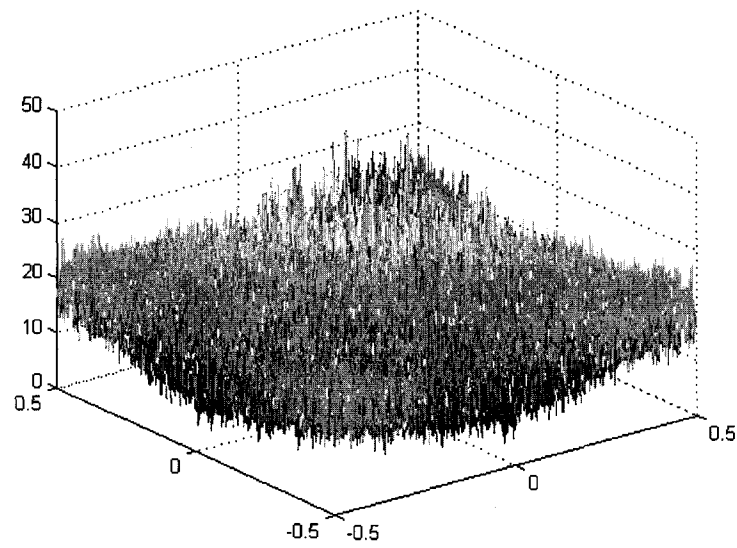


Figure 62 Bispectral estimate of mean-shifted exponential distribution, colored signal ($\beta = 0.25$). Estimate was generated with a lag argument of 512 and a DFT length of 2048.

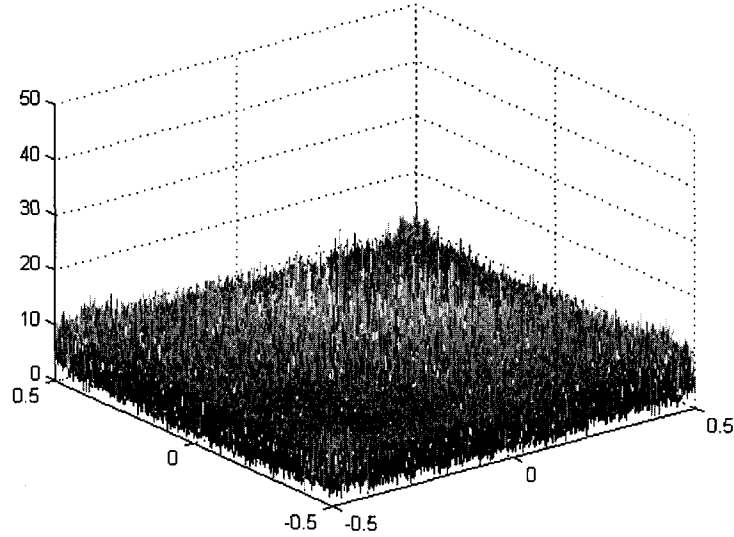


Figure 63 Bispectral estimate of mean-shifted Rayleigh distribution, colored signal ($\beta = 0.25$). Estimate was generated with a lag argument of 512 and a DFT length of 2048.

L. Lag argument = 1024, DFT length = 4096, colored signal ($\beta = 0.25$)

Figures 64 through 67 show the bispectral estimates of the colored signals generated when *bispeci* has been given a lag argument of 1024 and a DFT length of 4096. All the other *bispeci* parameters have been allowed to remain as their defaults.

The plots here are largely indistinguishable from the previous four (Figures 60-63). As the lag argument and DFT size has continued to grow, so has the variance of the bispectral estimate. In the cases of the Gaussian and uniform distributions, this has resulted in the appearance of spikes of signal energy where there should be none. In the cases of the mean-shifted exponential and mean-shifted Rayleigh distributions, this has made the signal energy which is present, less distinguishable from the noise floor. Figure 64 shows the calculated bispectrum of the colored Gaussian-distributed ensemble. The filter does not appear to have had a significant effect, as the bispectrum is flat and has zero magnitude.

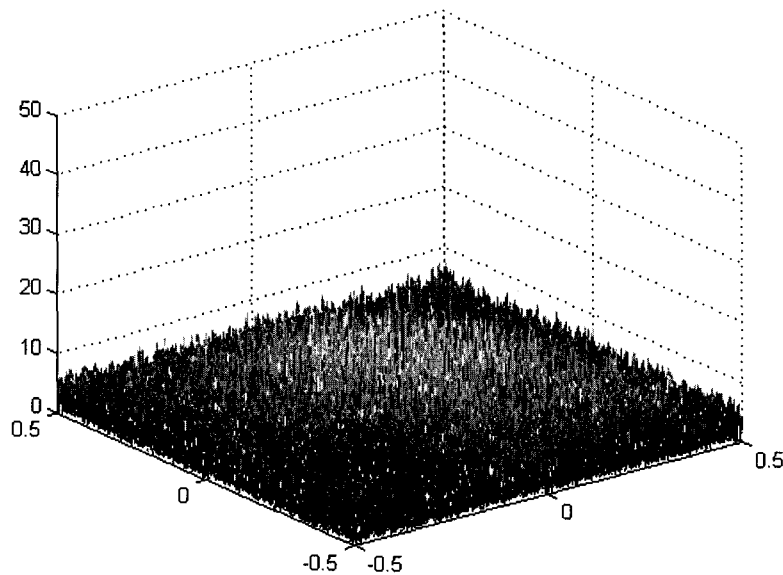


Figure 64 Bispectral estimate of Gaussian distribution, colored signal ($\beta = 0.25$). Estimate was generated with a lag argument of 1024 and a DFT length of 4096.

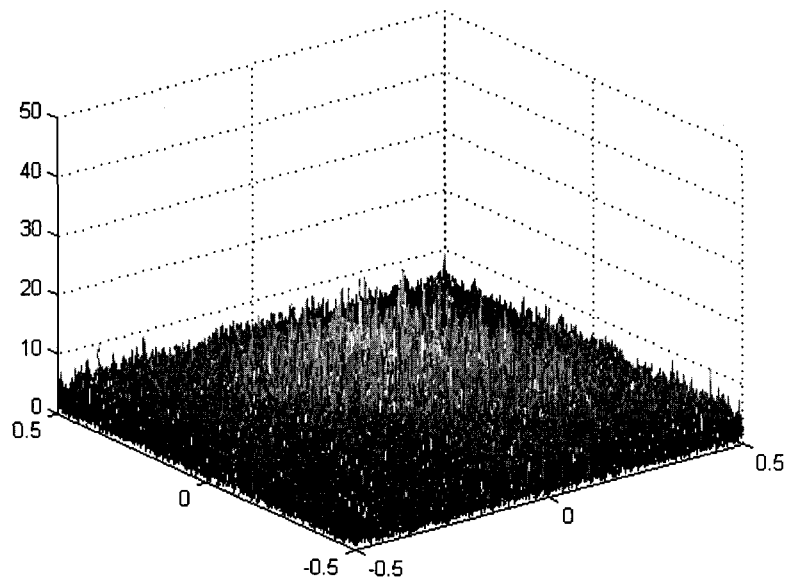


Figure 65 Bispectral estimate of uniform distribution, colored signal ($\beta = 0.25$). Estimate was generated with a lag argument of 1024 and a DFT length of 4096.

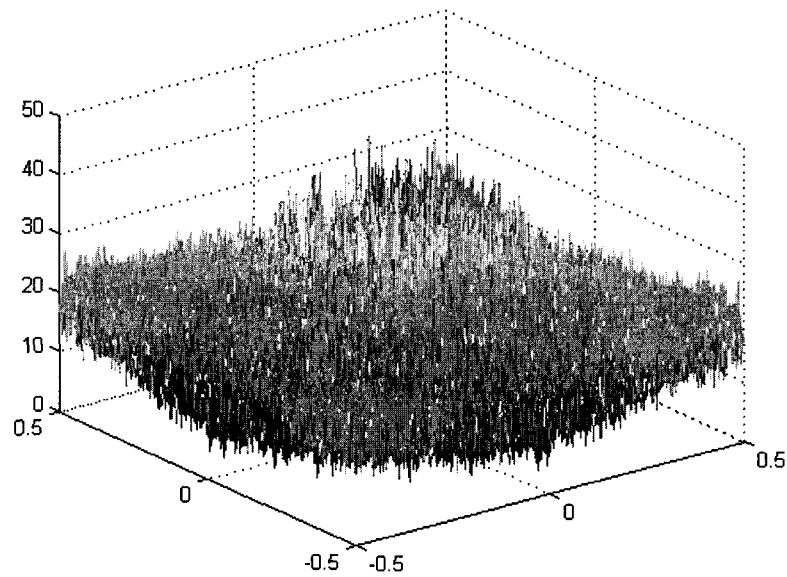


Figure 66 Bispectral estimate of mean-shifted exponential distribution, colored signal ($\beta = 0.25$). Estimate was generated with a lag argument of 1024 and a DFT length of 4096.

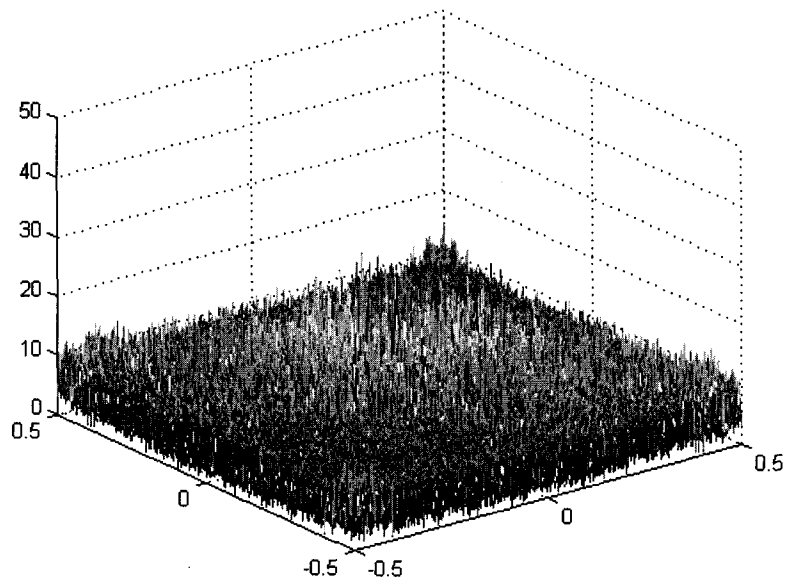


Figure 67 Bispectral estimate of mean-shifted Rayleigh distribution, colored signal ($\beta = 0.25$). Estimate was generated with a lag argument of 1024 and a DFT length of 4096.

M. Lag argument = 128, DFT length = 1024, colored signal ($\beta = 0.25$)

Figures 68 through 71 show the bispectral estimates of the colored signals generated when the lag argument to *bispeci* has been returned to 128, but the DFT length is increased to 1024. It is useful to compare these plots to those of Figures 52 through 55, in which the lag argument was 128 but the DFT size was the default of 512. In each case, the plot with the increased DFT length is largely the same as that generated with a shorter DFT length. This is in keeping with the result suggested earlier, that DFT length is not terribly significant (at least, it does not seem so when visually comparing bispectral estimates).

Figure 68 shows the calculated bispectrum of the colored Gaussian-distributed ensemble. The filter does not appear to have had a significant effect, as the bispectrum is flat and has zero magnitude.

Figure 69 shows the calculated bispectrum of the colored uniform-distributed data ensemble. It too, as predicted, has zero magnitude and is flat, although a little ripple is evident.

Figure 70 shows the bispectral estimate of the mean-shifted exponential signal ensemble, after the signal has been processed by the filter. It shows that the filter has shaped this bispectrum as well, as the signal peaks well up from its third-order cumulant value.

Figure 71 shows the bispectral estimate of the mean-shifted Rayleigh signal ensemble, after the signal has been processed by the filter. As with the exponentially-distributed signal ensemble, the filter has obviously shaped the bispectrum here as well, although again, it is not nearly as pronounced as it is in the case of the mean-shifted exponential distribution.

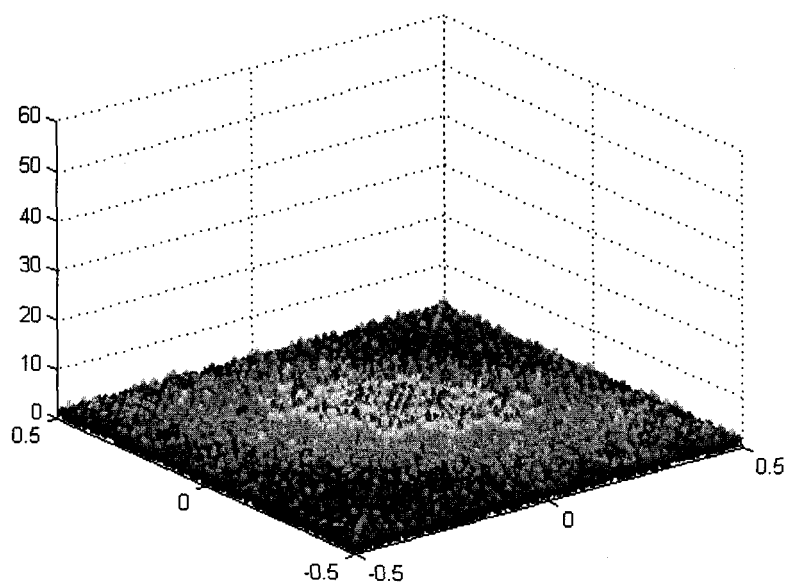


Figure 68 Bispectral estimate of Gaussian distribution, colored signal ($\beta = 0.25$). Estimate was generated with a lag argument of 128 and a DFT length of 1024.

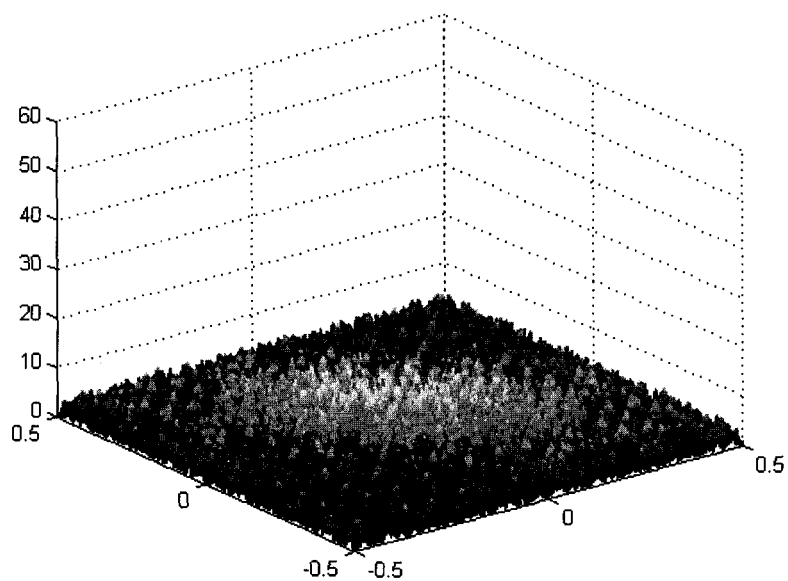


Figure 69 Bispectral estimate of uniform distribution, colored signal ($\beta = 0.25$). Estimate was generated with a lag argument of 128 and a DFT length of 1024.

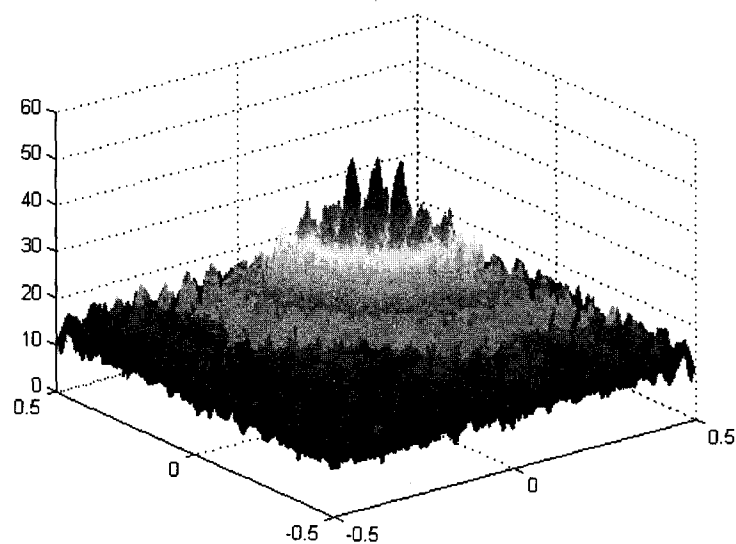


Figure 70 Bispectral estimate of mean-shifted exponential distribution, colored signal ($\beta = 0.25$). Estimate was generated with a lag argument of 128 and a DFT length of 1024.

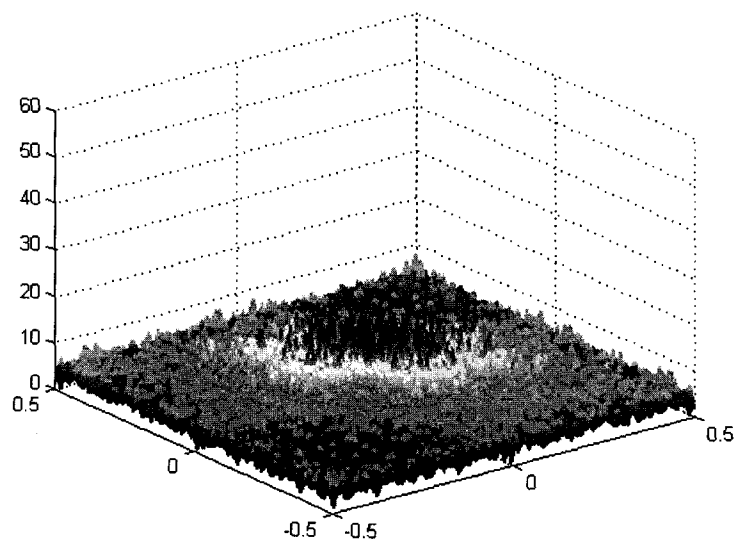


Figure 71 Bispectral estimate of mean-shifted Rayleigh distribution, colored signal ($\beta = 0.25$). Estimate was generated with a lag argument of 128 and a DFT length of 1024.

N. Lag argument = 128, DFT length = 2048, colored signal ($\beta = 0.25$)

Figures 72 through 75 show the bispectral estimates of the colored signals generated when the lag argument to *bispeci* has been held at 128, but the DFT length is increased to 2048. As before, it is useful to compare these plots to those of Figures 52 through 55, in which the lag argument was 128 but the DFT size was the default of 512. Also as in the previous case, increasing the DFT length did not seem to significantly change the bispectrum.

Figure 72 shows the calculated bispectrum of the colored Gaussian-distributed ensemble. The filter does not appear to have had a significant effect, as the bispectrum is flat and has zero magnitude.

Figure 73 shows the calculated bispectrum of the colored uniform-distributed data ensemble. It also has zero magnitude and is flat.

Figure 74 shows the bispectral estimate of the mean-shifted exponential signal ensemble, after the signal has been processed by the filter. It shows that the filter has shaped this bispectrum to a noticeable degree.

Figure 75 shows the bispectral estimate of the mean-shifted Rayleigh signal ensemble, after the signal has been processed by the filter. The filter has obviously shaped this signal as well, although not to the extent of the mean-shifted exponential signal.

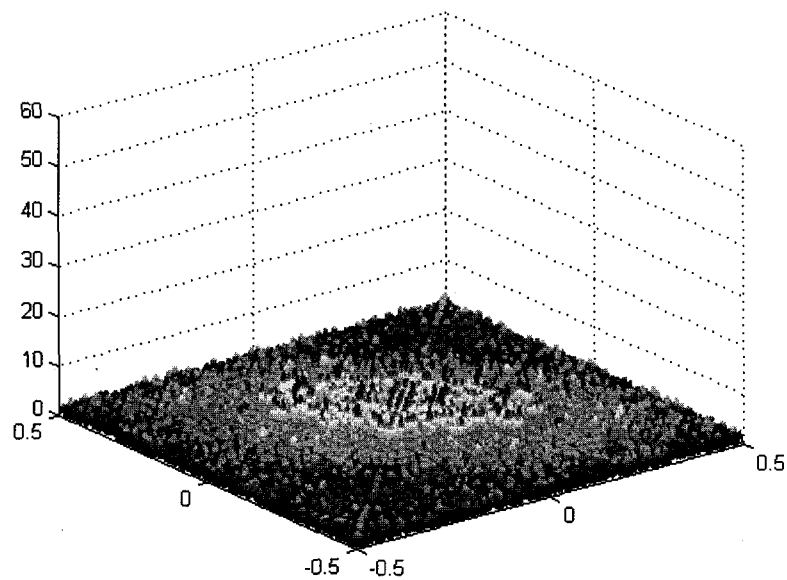


Figure 72 Bispectral estimate of Gaussian distribution, colored signal ($\beta = 0.25$). Estimate was generated with a lag argument of 128 and a DFT length of 2048.

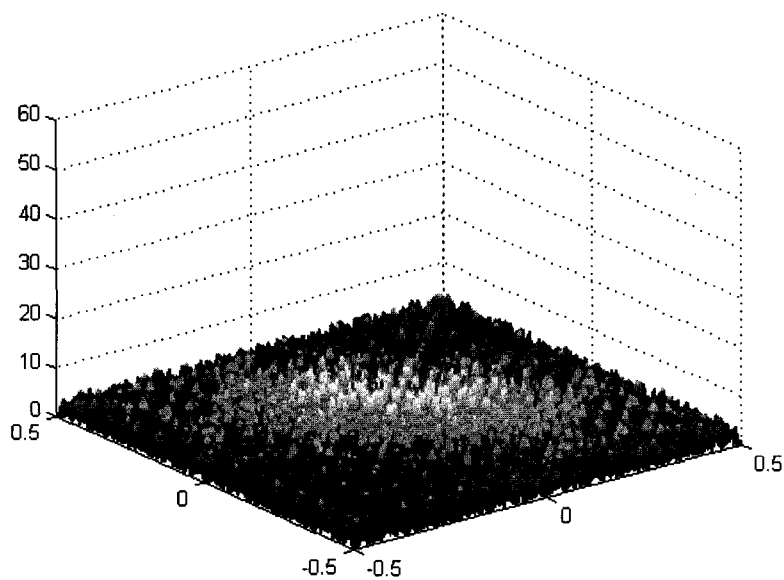


Figure 73 Bispectral estimate of uniform distribution, colored signal ($\beta = 0.25$). Estimate was generated with a lag argument of 128 and a DFT length of 2048.

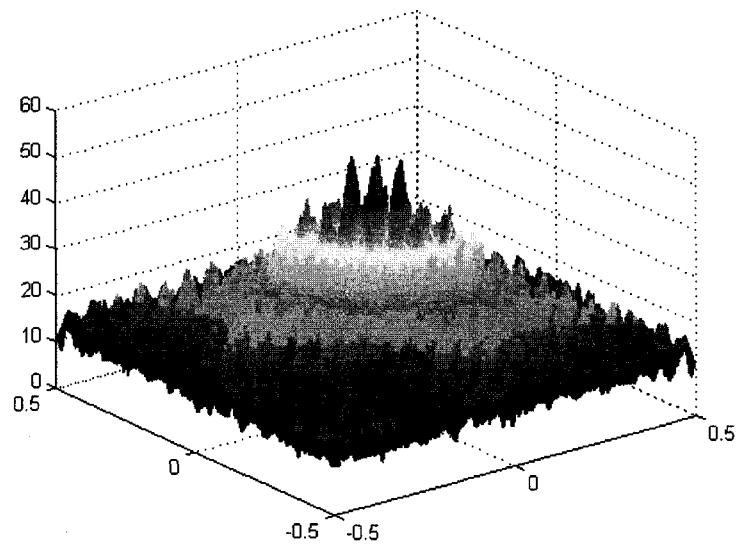


Figure 74 Bispectral estimate of mean-shifted exponential distribution, colored signal ($\beta = 0.25$). Estimate was generated with a lag argument of 128 and a DFT length of 2048.

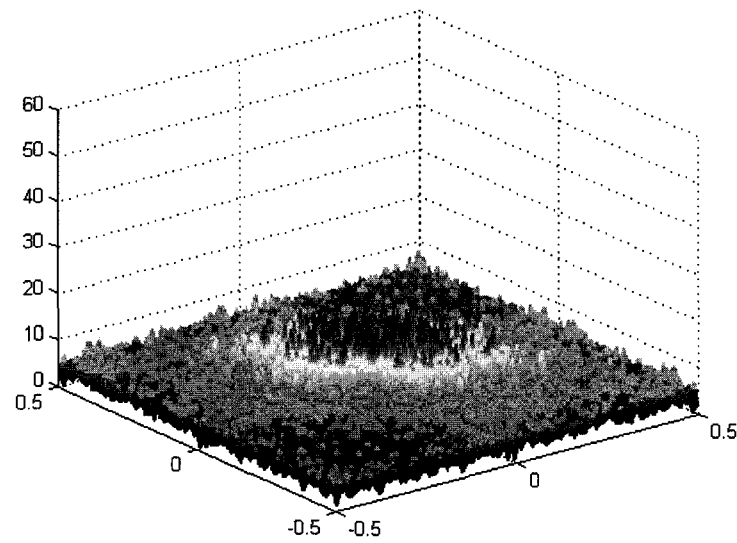


Figure 75 Bispectral estimate of mean-shifted Rayleigh distribution, colored signal ($\beta = 0.25$). Estimate was generated with a lag argument of 128 and a DFT length of 2048.

APPENDIX C. PLOTS FOR CHAPTER IV

This Appendix contains detailed results of the procedure described in Chapter IV. In Chapter IV, the two data ensembles with asymmetric distributions (mean-shifted exponential and mean-shifted Rayleigh) were colored with a filter and then added to Gaussian noise signals of varying strengths. The bispectral estimates of the resulting signal and noise ensembles were then generated using *bispeci*, and *mesh* plots of these estimates are given in this Appendix.

A. Mean-shifted exponential distribution, lag argument = 128, colored signal

Figures 76 through 80 show the estimated bispectra of the mean-shifted exponentially-distributed dataset, after it has been run through a filter (with parameter β equal to 0.1). The bispectral estimates in this case were generated with *bispeci*, which was given a lag argument of 128 (all other parameters were left to their defaults). As in the previous plots in the Appendices, these are magnitude plots of bispectral estimates with phase components introduced by the *bispeci* function.

Figure 76 shows the original signal, with no noise signal added. It is the same as Figure 54, scaled to the same axes as Figures 77 through 80, and is included for reference.

Figure 77 shows the bispectral estimate when the original colored signal is added to a Gaussian noise signal with a variance of 0.5. The bispectrum here looks essentially the same as that shown in Figure 76; the only obvious difference is that the entire signal level seems to have been raised somewhat by the additive noise, as would be expected.

Figure 78 shows the bispectral estimate when the original colored signal is added to a Gaussian noise signal with unity variance. Again, the signal level seems to have risen slightly—but in this case it is evident that the outermost peaks of signal energy have begun to disappear into the rising noise floor.

Figure 79 shows the bispectral estimate when the original colored signal is added to a Gaussian noise signal with variance four. The increased noise variance is evident as the bispectral floor becomes even more jagged. The signal energy peaks appear to have been raised somewhat; however, they have also moved as a result of coupling with the

noise signal.

Figure 80 shows the bispectral estimate when the original colored signal is added to a Gaussian noise signal with variance 16. Here, the increased variance is immediately evident, as is the fact that the signal energy is spread around the spectrum.

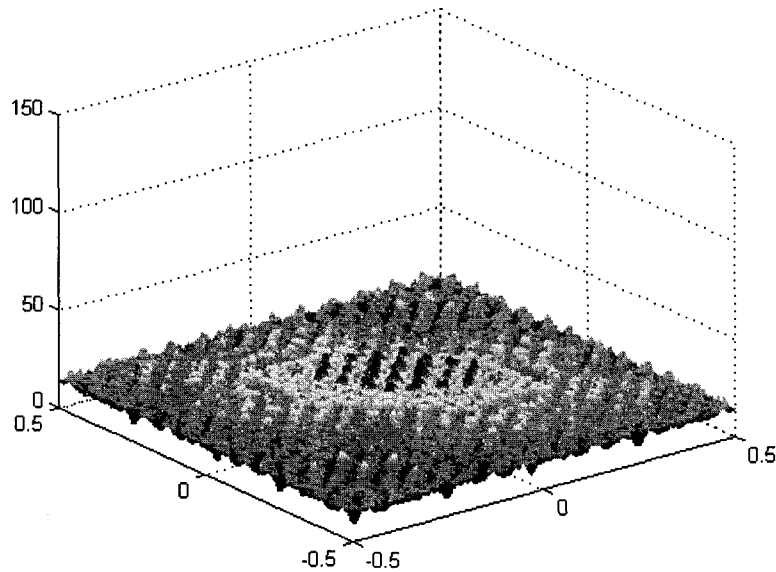


Figure 76 Bispectral estimate of mean-shifted exponentially-distributed signal in no noise.

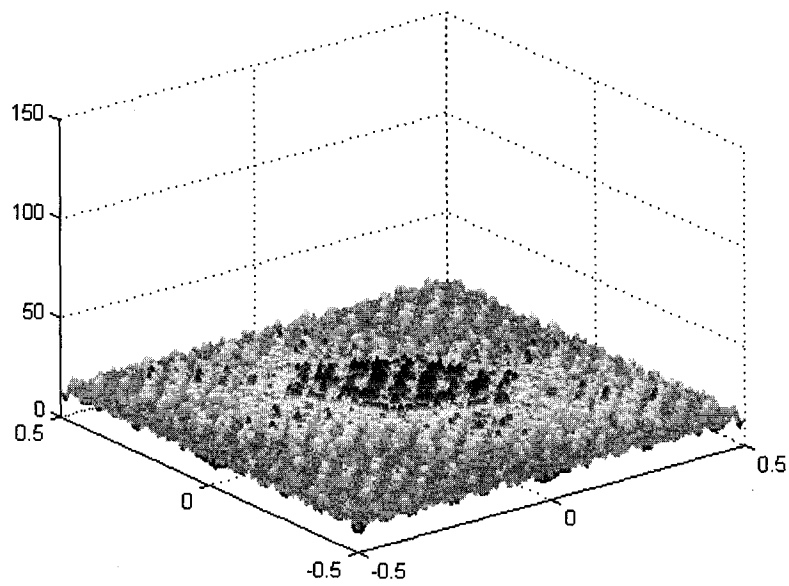


Figure 77 Bispectral estimate of mean-shifted exponentially-distributed signal in noise with variance = 0.5.

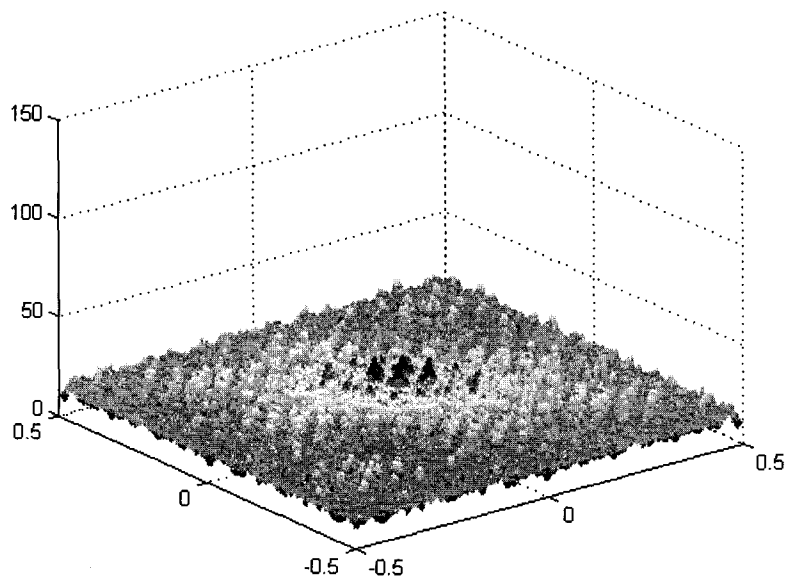


Figure 78 Bispectral estimate of mean-shifted exponentially-distributed signal in noise with variance = 1.

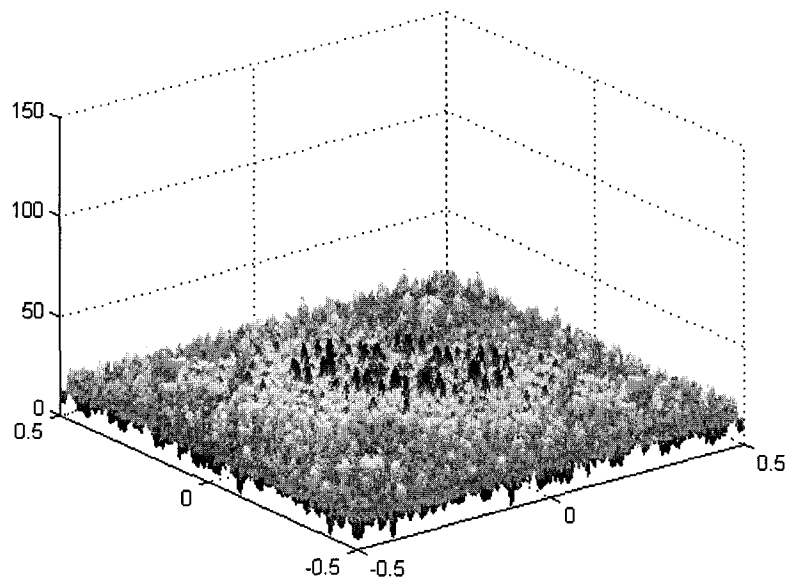


Figure 79 Bispectral estimate of mean-shifted exponentially-distributed signal in noise with variance = 4.

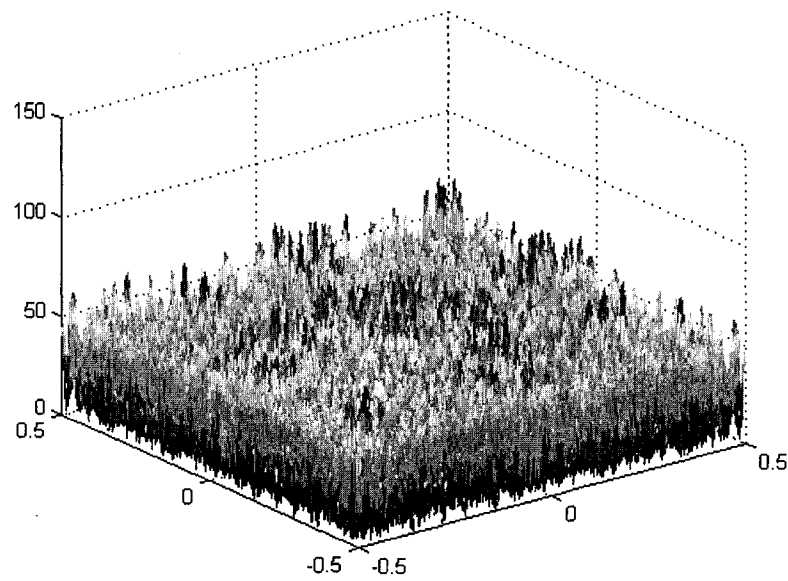


Figure 80 Bispectral estimate of mean-shifted exponentially-distributed signal in noise with variance = 16.

B. Mean-shifted Rayleigh distribution, lag argument = 128, colored signal

Figures 81 through 85 show the estimated bispectra of the mean-shifted Rayleigh-distributed dataset, after it has been run through a filter (with parameter β equal to 0.1). The bispectral estimates in this case were generated with *bispeci*, which was given a lag argument of 128 (all other parameters were left to their defaults). As in the previous plots in the Appendices, the bispectral estimates shown in these plots had phase components introduced by the *bispeci* function, but the plots here are of the magnitudes of the estimates.

Figure 81 shows the original signal, with no noise signal added. It is the same as Figure 55, scaled to the same axes as Figures 82 through 85, and is included for reference. As in Chapter III, it is evident that a signal with the mean-shifted Rayleigh distribution does not show up as well as one with a mean-shifted exponential distribution.

Figure 82 shows the bispectral estimate when the original colored signal is added to a Gaussian noise signal with a variance of 0.5. The bispectrum here looks essentially the same as that shown in Figure 81, although there is some evident spreading of the signal energy caused by the noise signal.

Figure 83 shows the bispectral estimate when the original colored signal is added to a Gaussian noise signal with unity variance. It looks essentially the same as Figure 82, and shows the same low signal level and slight degree of spreading.

Figure 84 shows the bispectral estimate when the original colored signal is added to a Gaussian noise signal with variance four. Here, the increased noise strength is somewhat more evident, in the form of increased variance in the noise floor. There are also a few bins of energy visible above the noise floor. While some of them would appear to correspond to the signal itself, others are obviously simply the effects of the noise.

Figure 85 shows the bispectral estimate when the original colored signal is added to a Gaussian noise signal with variance 16. As in the case of the mean-shifted exponential signals, the effects of raising the noise strength this much are immediately obvious. There does appear to be some signal energy at the center of plot.

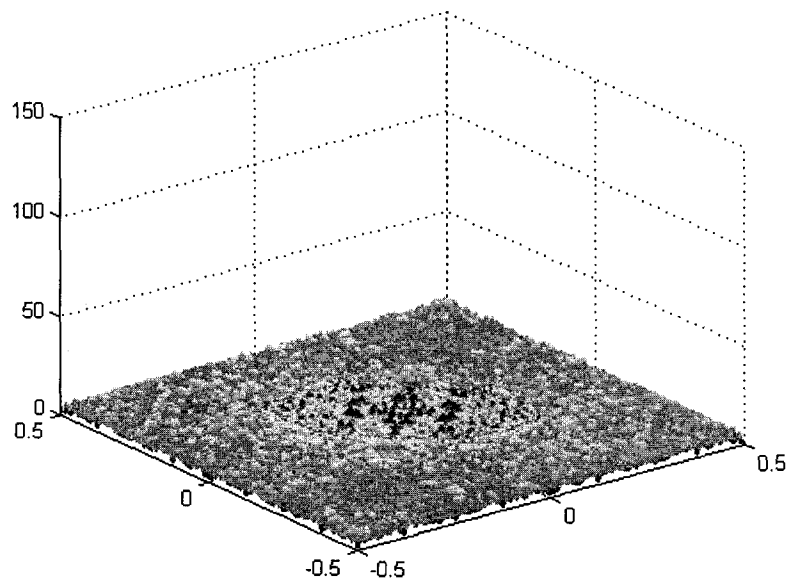


Figure 81 Bispectral estimate of mean-shifted Rayleigh distributed signal in no noise.

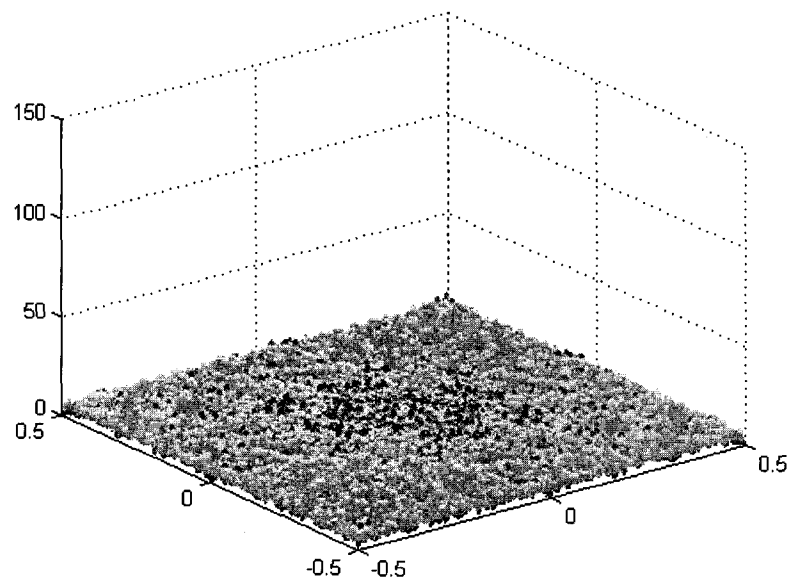


Figure 82 Bispectral estimate of mean-shifted Rayleigh distributed signal in noise with variance = 0.5.

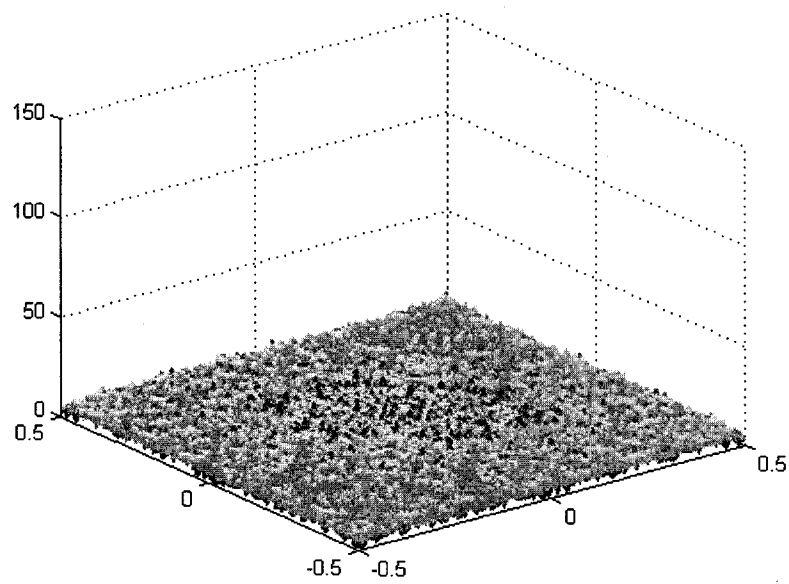


Figure 83 Bispectral estimate of mean-shifted Rayleigh distributed signal in noise with variance = 1.

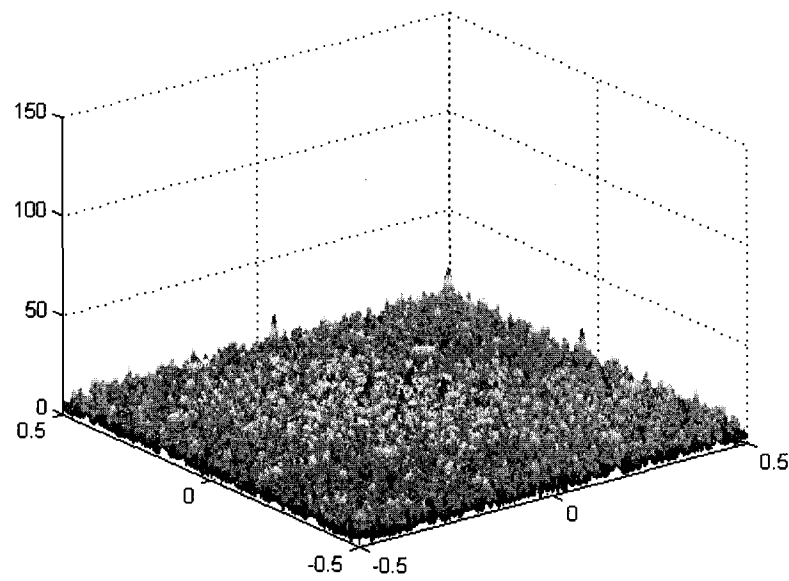


Figure 84 Bispectral estimate of mean-shifted Rayleigh distributed signal in noise with variance = 4.

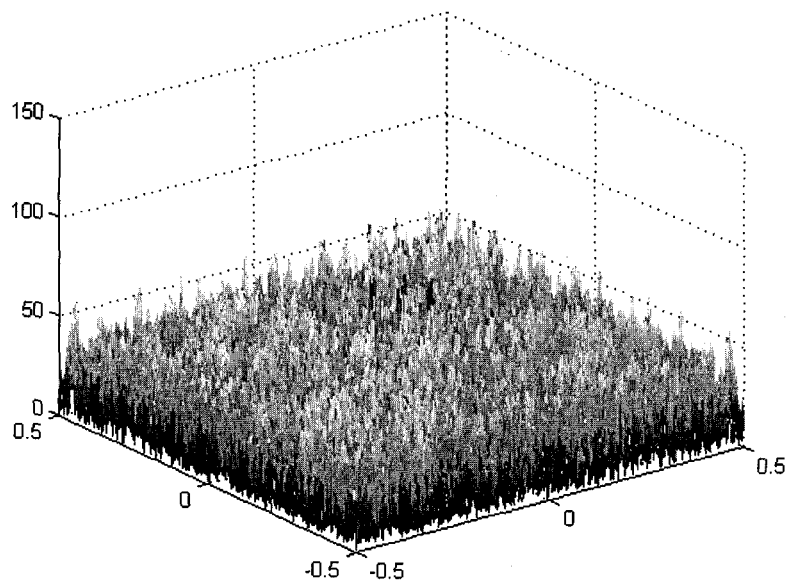


Figure 85 Bispectral estimate of mean-shifted Rayleigh distributed signal in noise with variance = 16.

C. Mean-shifted exponential distribution, lag argument = 256, colored signal

Figures 86 through 90 show the estimated bispectra of the mean-shifted exponentially-distributed dataset, after it has been run through a filter (with parameter β equal to 0.1). The bispectral estimates in this case were generated with *bispeci*, which was given a lag argument of 256 (all other parameters were left to their defaults). As in the previous plots in the Appendices, these are magnitude plots of bispectral estimates with phase components introduced by the *bispeci* function.

Figure 86 shows the original signal, with no noise signal added. It is the same as Figure 58, scaled to the same axes as Figures 87 through 90, and is included for reference.

Figure 87 shows the bispectral estimate when the original colored signal is added to a Gaussian noise signal with a variance of 0.5. The bispectrum here looks essentially the same as that shown in Figure 86; the only obvious difference is that the entire signal level seems to have been raised somewhat by the additive noise, as would be expected. It appears that the signal energy in the center has also been smeared somewhat by the addi-

tive noise.

Figure 88 shows the bispectral estimate when the original colored signal is added to a Gaussian noise signal with unity variance. The signal is still largely unchanged from the previous two plots.

Figure 89 shows the bispectral estimate when the original colored signal is added to a Gaussian noise signal with variance four. The original signal energy is still visible in the center of the plot. The additive noise has raised its level somewhat; however, it has obviously also raised the noise floor significantly, and coupled with the mean-shifted exponential signal as well.

Figure 90 shows the bispectral estimate when the original colored signal is added to a Gaussian noise signal with variance 16. The effects of increasing the noise strength are vividly evident in this plot, as the variance of the bispectral estimate has hidden any signal energy that might be present.

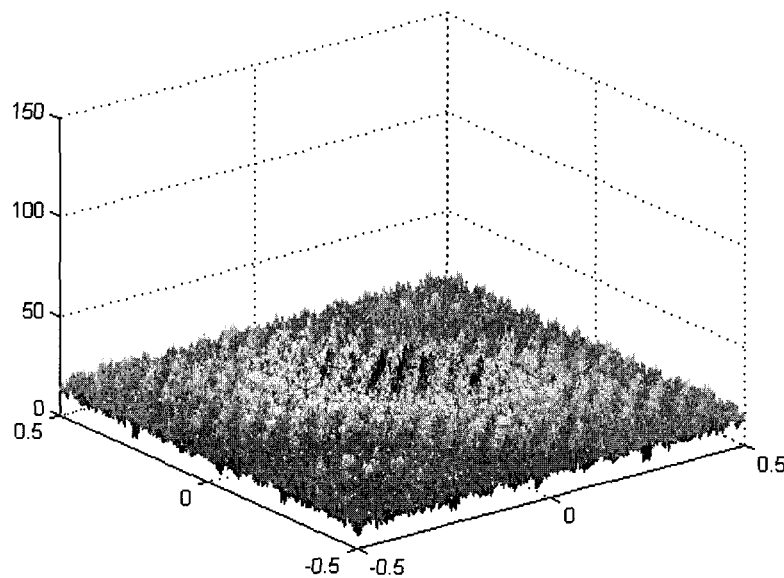


Figure 86 Bispectral estimate of mean-shifted exponentially distributed signal in no noise.

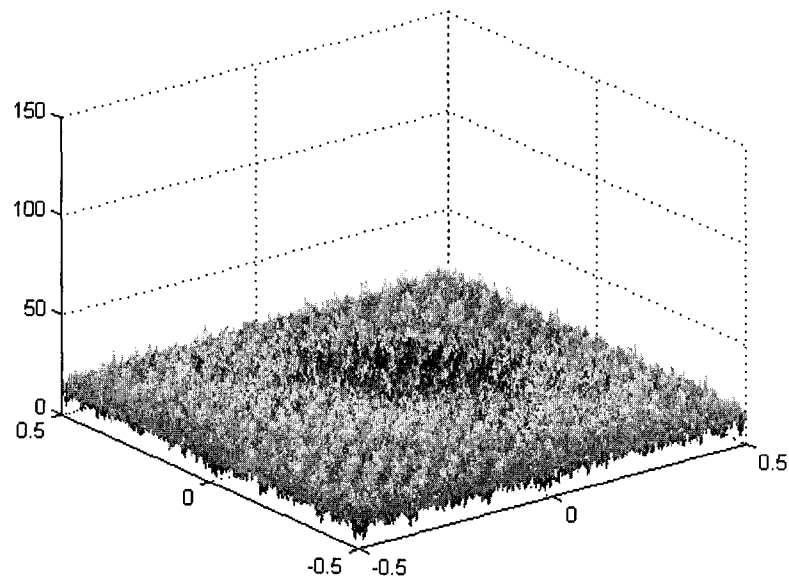


Figure 87 Bispectral estimate of mean-shifted exponentially distributed signal in noise with variance = 0.5.

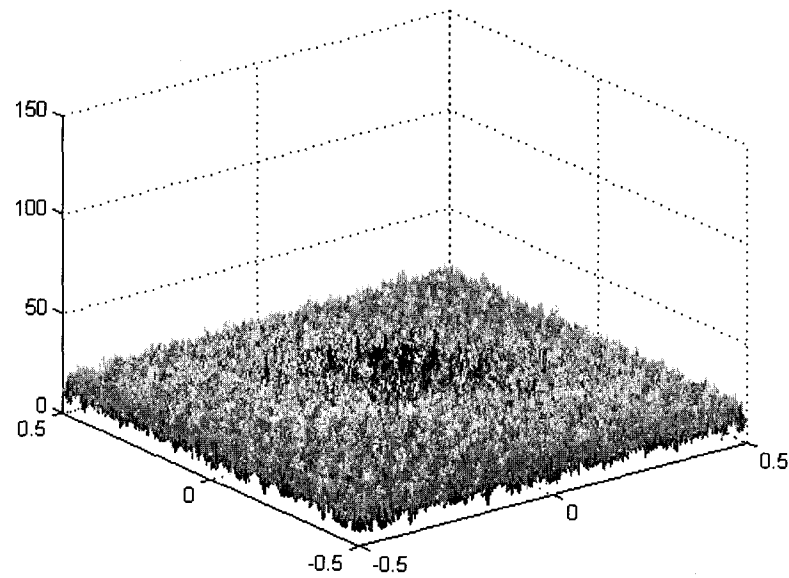


Figure 88 Bispectral estimate of mean-shifted exponentially distributed signal in noise with variance = 1.

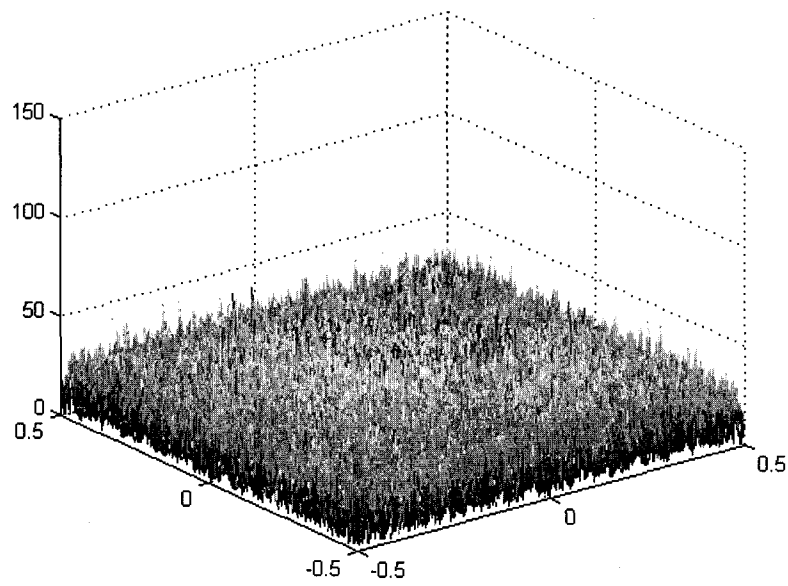


Figure 89 Bispectral estimate of mean-shifted exponentially distributed signal in noise with variance = 4.

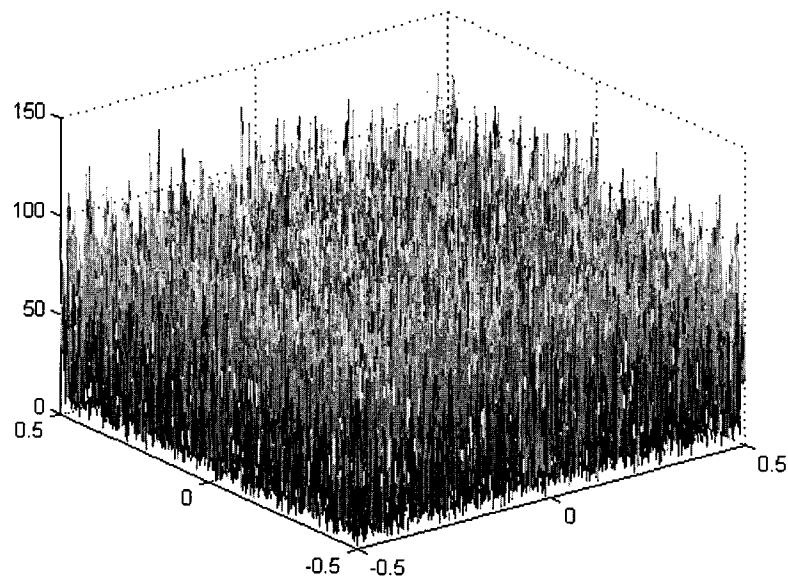


Figure 90 Bispectral estimate of mean-shifted exponentially distributed signal in noise with variance = 16.

D. Mean-shifted Rayleigh distribution, lag argument = 256, colored signal

Figures 91 through 95 show the estimated bispectra of the mean-shifted Rayleigh-distributed dataset, after it has been run through a filter (with parameter β equal to 0.1). The bispectral estimates in this case were generated with *bispeci*, which was given a lag argument of 256 (all other parameters were left to their defaults). As in the previous plots in the Appendices, these are magnitude plots of bispectral estimates with phase components introduced by the *bispeci* function.

Figure 91 shows the original signal, with no noise signal added. It is the same as Figure 60, scaled to the same axes as Figures 92 through 95, and is included for reference.

Figure 92 shows the bispectral estimate when the original colored signal is added to a Gaussian noise signal with a variance of 0.5. What signal energy was evident in Figure 91 appears to have vanished in this plot.

Figure 93 shows the bispectral estimate when the original colored signal is added to a Gaussian noise signal with unity variance. The signal is still largely unchanged from that shown in Figures 91 and 92, although the increasing variance in the bispectral estimate is just starting to be apparent.

Figure 94 shows the bispectral estimate when the original colored signal is added to a Gaussian noise signal with variance four. It is still difficult to distinguish any signal components here, but the increased noise strength is continuing to cause the variance (and the mean) of the noise floor to grow.

Figure 95 shows the bispectral estimate when the original colored signal is added to a Gaussian noise signal with variance 16. As in Figure 90, the large noise strength dominates any signal components that might be visible in the bispectrum (and there had been almost none visible before the noise floor was raised).

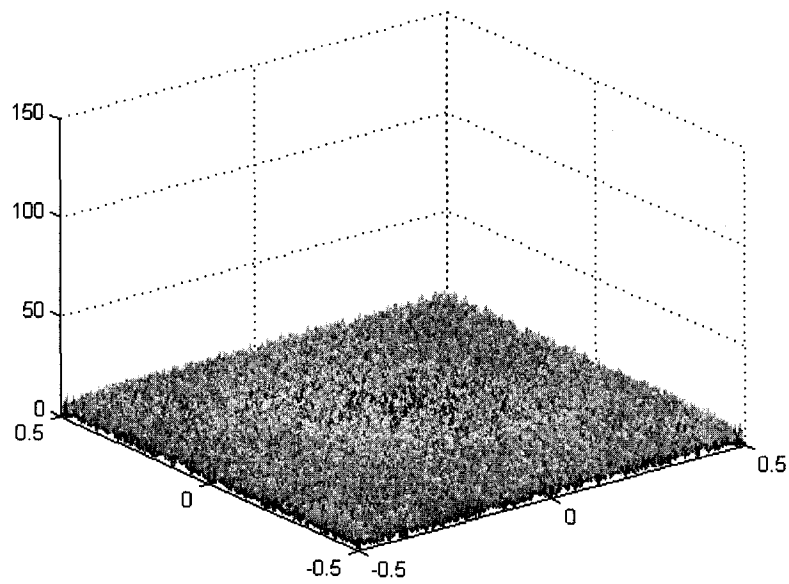


Figure 91 Bispectral estimate of mean-shifted Rayleigh distributed signal in no noise.

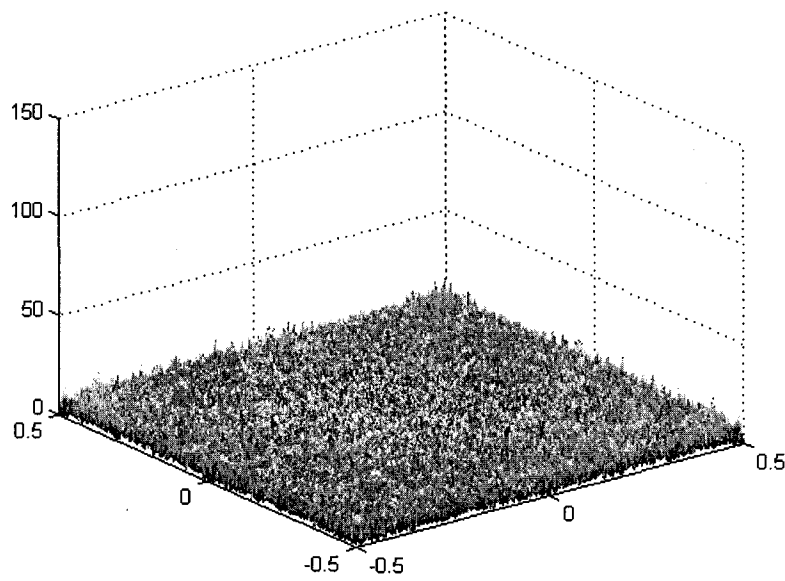


Figure 92 Bispectral estimate of mean-shifted Rayleigh distributed signal in noise with variance = 0.5.

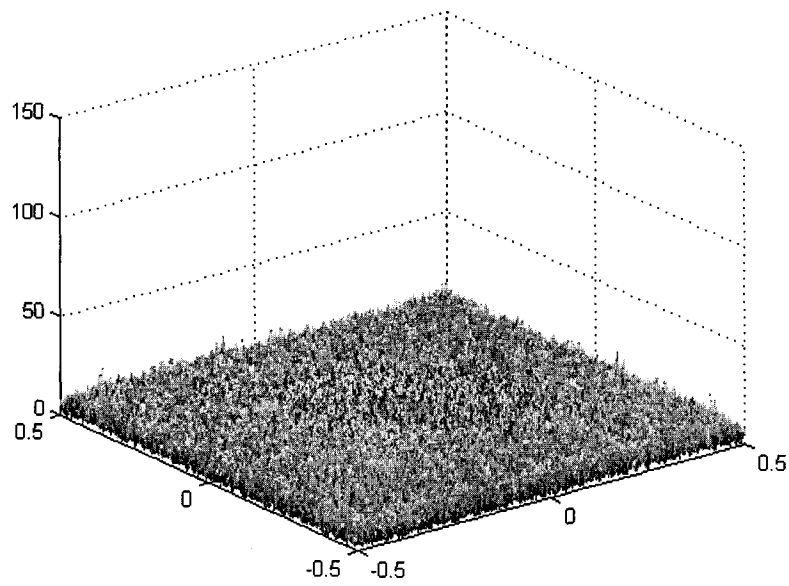


Figure 93 Bispectral estimate of mean-shifted Rayleigh distributed signal in noise with variance = 1.

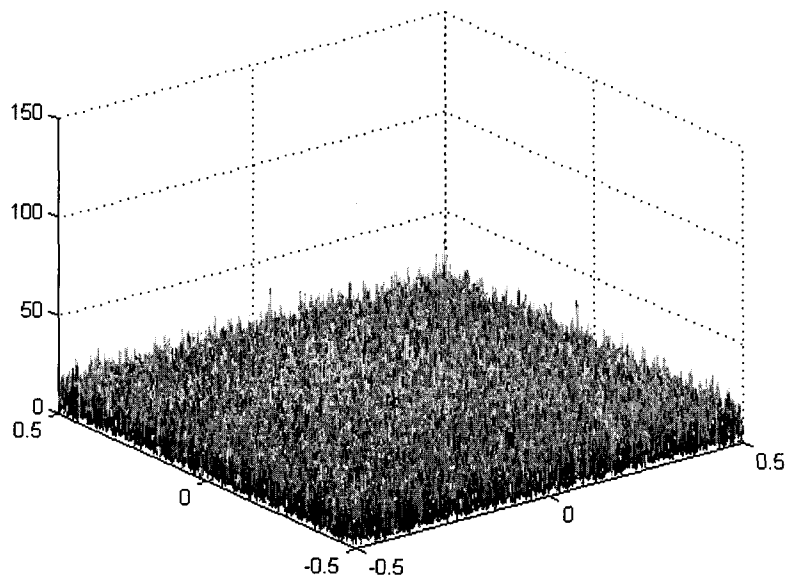


Figure 94 Bispectral estimate of mean-shifted Rayleigh distributed signal in noise with variance = 4.

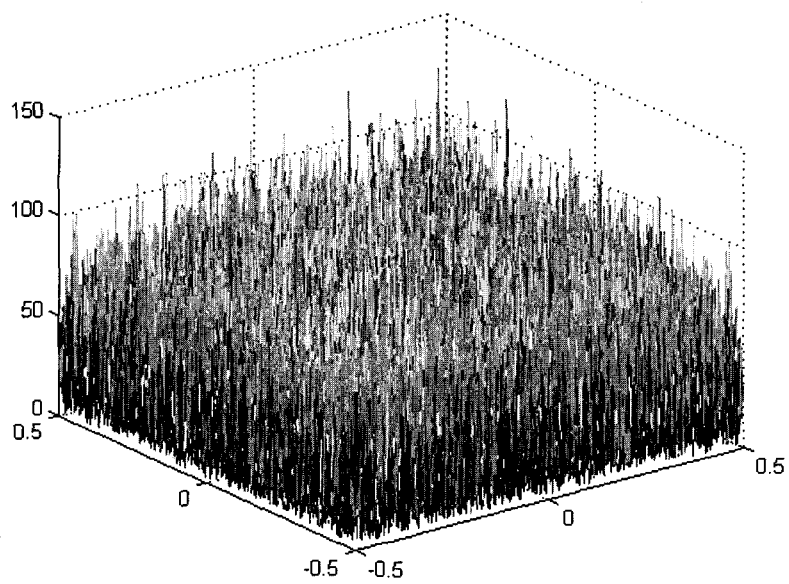


Figure 95 Bispectral estimate of mean-shifted Rayleigh distributed signal in noise with variance = 16.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX D. GENERAL EXPRESSIONS FOR THIRD-ORDER CUMULANTS

This Appendix contains a derivation of a general expression for the third-order cumulant of a zero-mean independent, identically-distributed ("i.i.d.") process. The calculations here demonstrate why the expected values given in Table 1 are scalars.

Let x_i , x_j , and x_k be samples from any zero-mean i.i.d. process. Then the third-order cumulant is given by [Ref. 7] as

$$C_x^3(x_i, x_j, x_k) = E\{x_i x_j x_k\} - E\{x_i\} \cdot E\{x_j x_k\} - E\{x_j\} \cdot E\{x_i x_k\} - E\{x_k\} \cdot E\{x_i x_j\} + 2E\{x_i\} \cdot E\{x_j\} \cdot E\{x_k\}, \quad (D.1)$$

where $E\{x_i\}$ is expectation (thus the above terms are the first, second, and third moments of the process).

Now for a zero mean process, $E\{x_i\} = E\{x_j\} = E\{x_k\} = 0$. Therefore, Equation (D.1) reduces to

$$C_x^3(x_i, x_j, x_k) = E\{x_i x_j x_k\}. \quad (D.2)$$

Since the samples are independent, we can write

$$E\{x_i x_j x_k\} = \begin{cases} E\{x_i\} \cdot E\{x_j\} \cdot E\{x_k\} & \text{for } i \neq j \neq k \\ E\{x_i\} \cdot E\{x_j^2\} & \text{for } i \neq j = k \\ E\{x_i^3\} & \text{for } i = j = k. \end{cases} \quad (D.3)$$

Since the samples have zero mean, the above result is zero in all except for the last case.

Thus we can write

$$C_x^3(x_i, x_j, x_k) = \alpha \delta(i-j) \delta(i-k) \quad (D.4)$$

where

$$\alpha = E\{x_i^3\}. \quad (D.5)$$

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX E. THEORETICAL THIRD-ORDER CUMULANT EXPRESSIONS FOR DISTRIBUTIONS USED IN THIS THESIS

This Appendix contains derivations for the expected third-order cumulants of the four distributions used throughout this study. In the cases of the mean-shifted exponential and mean-shifted Rayleigh distributions, it also examines the effects that biasing the distribution has on its probability density function.

A. Normal (Gaussian) distribution

Given a mean μ and variance σ^2 , the probability density function of a normal (Gaussian) distribution function is given by [Ref. 5]:

$$f_x(x) = \frac{1}{\sigma \sqrt{2\pi}} e^{\frac{-(x-\mu)^2}{(2\sigma^2)}} \quad \text{for } -\infty < x < \infty, \quad (\text{E.1})$$

and its third-order cumulant (and every odd cumulant) is zero due to the symmetric nature of the distribution. Recall that for this procedure $\mu = 0$ and $\sigma^2 = 4$; in the case of the normal distribution these values are simply provided to MATLAB during the sequence generation.

B. Uniform distribution

Given a mean μ and variance σ^2 , the probability density function of a uniform (rectangular) distribution function is given by [Ref. 5]:

$$f_{(x)}(x) = \begin{cases} \frac{1}{b-a} & \text{for } a < x < b \\ 0 & \text{otherwise} \end{cases}, \quad (\text{E.2})$$

where a is the lower limit and b is the upper limit of the distribution. The mean of a uniform distribution is $(a + b)/2$ and forcing zero mean ($\mu = 0$) requires that $a = -b$; i.e., a and b are on opposite sides of and equidistant from zero.

The variance of the uniform distribution is given by [Ref. 5]:

$$\sigma^2 = \frac{(b-a)^2}{12} = \frac{b^2}{3}, \quad (\text{E.3})$$

and for $\sigma^2 = 4$ we require $a = -b = 2\sqrt{3}$.

Because the uniform distribution is symmetric about its mean (here, zero), it also has the property that its third- (and every odd-)order moment is zero.

C. Mean-shifted exponential distribution

Given a mean μ and variance σ^2 , the probability density function of an (unaltered) exponential distribution function is given by [Ref. 5]:

$$f_{(x)}(x) = \begin{cases} \lambda e^{-\lambda x} & \text{for } x \geq 0 \\ 0 & \text{for } x < 0 \end{cases}, \quad (\text{E.4})$$

where λ is the parameter of the distribution.

The mean of an unaltered exponential-distributed sequence is $1/\lambda$. In order to force the mean to zero, $1/\lambda$ is simply subtracted from the probability density function, biasing it across the zero point. The probability density function of this new "mean-shifted exponential" sequence is then given by

$$f_{(x)}(x) = \begin{cases} \lambda e^{-\lambda(x+1/\lambda)} & \text{for } x \geq -\frac{1}{\lambda} \\ 0 & \text{otherwise} \end{cases}. \quad (\text{E.5})$$

This new sequence is zero-mean, but retains the variance of the unaltered exponential sequence which is $\sigma^2 = 1/\lambda^2$. Again, forcing the variance to be 4 requires that λ take a specific value, in this case 0.5.

The r^{th} -order cumulant ($r > 1$) is given by [Ref. 5]:

$$C_r^x(x_1, x_2, \dots, x_{r-1}) = (r-1)! \left(\frac{1}{\lambda^r} \right), \quad (\text{E.6})$$

and thus the third-order cumulant is equal to $2/\lambda^3$. When λ is set to 0.5, as it is here, the third-order cumulant evaluates to 16.

D. Mean-shifted Rayleigh distribution

Given a mean μ and variance σ^2 , the probability density function of an (unaltered) Rayleigh distribution function is defined as [Ref. 5]:

$$f_{(x)}(x) = \begin{cases} \frac{x}{b} e^{\frac{-x^2}{2b^2}} & \text{for } x > 0 \\ 0 & \text{otherwise} \end{cases}, \quad (\text{E.7})$$

where b is a scale parameter and defines the shape of the curve. The mean of an unaltered Rayleigh-distributed sequence is $b\sqrt{\pi/2}$.

The variance of an unaltered Rayleigh-distributed sequence is given by [Ref. 5]. As in the previous examples, a sequence with a variance of four is desired. For $\sigma^2 = 4$, b has a value of $\sqrt{4/(2 - [\pi/2])}$, which evaluates to approximately 3.0528.

The standard (unaltered) Rayleigh distribution is nonzero mean. It was desired to have two asymmetric (exponential and Rayleigh) distributions to compare to the symmetric Gaussian and uniform distributions; however, all the sample sequences needed to be zero-mean. Therefore, as with the exponential distribution, the Rayleigh distribution had its mean subtracted to bias it about the zero point.

When the mean is subtracted from the standard Rayleigh function, the probability density function produced is

$$f_{(x)}(x) = \begin{cases} \frac{\left(x + b\sqrt{\frac{\pi}{2}}\right)}{b^2} e^{\frac{-\left(x + b\sqrt{\frac{\pi}{2}}\right)^2}{2b^2}} & \text{for } x \geq -b\sqrt{\frac{\pi}{2}} \\ 0 & \text{otherwise} \end{cases} \quad (\text{E.8})$$

and, just as in the case of the mean-shifted exponential distribution, the variance of this sequence is unchanged.

The r^{th} -order moment of a Rayleigh-distributed sequence is given by [Ref. 5]:

$$E\{x^r\} = (\sqrt{2}b)^r \left(\frac{r}{2}\right) \Gamma\left(\frac{r}{2}\right), \quad (\text{E.9})$$

where $\Gamma(c)$ denotes the standard gamma function. Now the third-order cumulant will be, in the zero-mean case, the same as the third-order moment, which is given by

$$E\{x^3\} = (\sqrt{2}b)^3 \left(\frac{3}{2}\right) \Gamma\left(\frac{3}{2}\right) \approx (b^3 \cdot 3 \cdot \sqrt{2} \cdot (0.88623)), \quad (\text{E.10})$$

where the last term is the truncated evaluation of the gamma function.

As in the preceding examples, the requirement of a specific variance forces the scale parameter b to a specific value as well, which in turn dictates the third-order cumulant. In this case, the third-order cumulant of a zero-mean Rayleigh-distributed sequence with variance $\sigma^2 = 4$ is approximately $b = 5.04739$.

APPENDIX F. FLOW DIAGRAM OF MATLAB SIMULATIONS

This Appendix details the flow of the MATLAB script used in Chapter II. The script itself is in Appendix G, and was used to generate mean, variance, and third-order cumulant estimates for ensembles of varying sizes of each of the four distributions examined here.

I. First, an ensemble of m realizations is created. Each realization is of length n and is distributed according to the desired PDF (e.g., Gaussian, uniform, mean-shifted exponential, or mean-shifted Rayleigh).

$$\begin{pmatrix} x_{11} & x_{12} & x_{13} & \cdots & x_{1n} \\ x_{21} & x_{22} & x_{23} & \cdots & x_{2n} \\ x_{31} & x_{32} & x_{33} & \cdots & x_{3n} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ x_{m1} & x_{m2} & x_{m3} & \cdots & x_{mn} \end{pmatrix} \quad (\text{F.1})$$

II. Next, a subset of each realization in the ensemble is used to generate an estimate of the mean, variance, and third-order cumulant estimate of the sequence. This produces m mean estimates (μ_{11} - μ_{m1}), m variance estimates (σ_{11} - σ_{m1}), and m third-order cumulant estimates (c^3_{11} - c^3_{m1}).

$$\begin{pmatrix} \{x_{11} & x_{12}\} & x_{13} & \cdots & x_{1n} \\ \{x_{21} & x_{22}\} & x_{23} & \cdots & x_{2n} \\ \{x_{31} & x_{32}\} & x_{33} & \cdots & x_{3n} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ \{x_{m1} & x_{m2}\} & x_{m3} & \cdots & x_{mn} \end{pmatrix} \rightarrow \begin{pmatrix} \mu_{11} & \sigma_{11} & c^3_{11} \\ \mu_{21} & \sigma_{21} & c^3_{21} \\ \mu_{31} & \sigma_{31} & c^3_{31} \\ \vdots & \vdots & \vdots \\ \mu_{m1} & \sigma_{m1} & c^3_{m1} \end{pmatrix} \quad (\text{F.2})$$

III. The m estimates of each statistic are then averaged to produce mean values of the sequence mean, variance, and third-order cumulant estimate. These three value are then saved.

$$\begin{pmatrix} \{x_{11} & x_{12}\} & x_{13} & \cdots & x_{1n} \\ \{x_{21} & x_{22}\} & x_{23} & \cdots & x_{2n} \\ \{x_{31} & x_{32}\} & x_{33} & \cdots & x_{3n} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ \{x_{m1} & x_{m2}\} & x_{m3} & \cdots & x_{mn} \end{pmatrix} \rightarrow \begin{pmatrix} \mu_{11} & \sigma_{11} & c_{11}^3 \\ \mu_{21} & \sigma_{21} & c_{21}^3 \\ \mu_{31} & \sigma_{31} & c_{31}^3 \\ \vdots & \vdots & \vdots \\ \mu_{m1} & \sigma_{m1} & c_{m1}^3 \end{pmatrix} \quad (F.3)$$

$$\begin{matrix} \downarrow & \downarrow & \downarrow \\ (\mu_1 & \sigma_1 & c_1^3) \end{matrix}$$

IV. The size of the subset is then increased exponentially, and the process is repeated, yielding a new mean value for the sequence mean, variance, and third-order cumulant estimate.

$$\begin{pmatrix} \{x_{11} & x_{12} & x_{13} & \cdots\} & x_{1n} \\ \{x_{21} & x_{22} & x_{23} & \cdots\} & x_{2n} \\ \{x_{31} & x_{32} & x_{33} & \cdots\} & x_{3n} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ \{x_{m1} & x_{m2} & x_{m3} & \cdots\} & x_{mn} \end{pmatrix} \rightarrow \begin{pmatrix} \mu_{12} & \sigma_{12} & c_{12}^3 \\ \mu_{22} & \sigma_{22} & c_{22}^3 \\ \mu_{32} & \sigma_{32} & c_{32}^3 \\ \vdots & \vdots & \vdots \\ \mu_{m2} & \sigma_{m2} & c_{m2}^3 \end{pmatrix} \quad (F.4)$$

$$\begin{matrix} \downarrow & \downarrow & \downarrow \\ \begin{pmatrix} \mu_1 & \sigma_1 & c_1^3 \\ \{\mu_2 & \sigma_2 & c_2^3\} \end{pmatrix} \end{matrix}$$

V. As the subset size is increased exponentially (base 2) from 1 to m , the resultant matrix grows to become size $3m$. It consists of three column vectors, each representing a statistic (mean, variance, or third-order cumulant estimate) as the subset size is increased. This provides a good measure for the effects of increased realization length (sample size) without producing a different sample set for each realization. These vectors may be plotted logarithmically against sequence length to illustrate the effects of increased realization length.

$$\begin{pmatrix} \mu_1 & \sigma_1 & c_1^3 \\ \mu_2 & \sigma_2 & c_2^3 \\ \vdots & \vdots & \vdots \\ \mu_m & \sigma_m & c_m^3 \end{pmatrix} \quad (F.5)$$

APPENDIX G. MATLAB CODE

A. MATLAB script from Chapter II

```
% (This is the script used to generate the values for phase I)

clear sequence;
clear averages;
clear variances;
clear estimates;
clear estimate2;

% Instead of generating an ensemble (and its third-order cumulant estimate)
% for each desired length, we will generate a full ensemble of 1024
% realizations (sequences) and generate estimates for lengths of
% (1, 2, 4, 8... 512, 1024).

    for index = 1:1024,
        sequence(:,index) = rpiid_var(n,type,4);
    end

% Now we have an array ('ensemble') consisting of 1024 elements, each
% of which is an n-point sample ('realization') of iid variables
% distributed according to the type specified.

% The next step will be to grab subsets of these 1024 512-point
sequences,
% and generate mean, variance, and third-order cumulant estimates for
the subsets.

    a = (0:10);
    b = 2.^a;

    for index = 1:11,
```

```

averages(index) = mean(mean(sequence(:,1:b(index)))));
variances(index) = mean(var(sequence(:,1:b(index)))));

for index2 = 1:b(index),
    estimate2(index2) = cumest(sequence(:,index2),3);
end
estimates(index) = mean(estimate2);
end

```

B. MATLAB FUNCTION *rpiid_var*

```

function [seq] = rpiid_var(nsamp,in_type,var)
%
% rpiid_var - a program to generate sequences of a specified
% probability density function, with a specified variance.
%
% Syntax:
%
% [seq] = rpiid_var(nsamp,in_type,var);
%
% where
%
% seq = output sequence
%
% nsamp = number of samples to be generated
% in_type = string specifying pdf to be generated, either:
%         'nor' for Gaussian/normal,
%         'uni' for uniform/rectangular,
%         'exp' for exponential,
%         'h'   for "shifted" exponential
%         'lap' for Laplace,
%         'ray' for Rayleigh,
%         'two' for "two-sided Rayleigh" (zero-mean), or
%         's'   for "shifted" Rayleigh (zero-mean)
% var = desired variance

```

```

%
% The mean of a Gaussian, uniform, or Laplace sequence
% generated here is zero. The mean of a Rayleigh sequence
% works out to something like sqrt((pi*var)/(4 - pi)) where
% 'var' is the variance you specify. (So unity variance
% produces a mean of 1.91, for example). The mean of an
% exponential sequence is the square root of the specified
% variance.
%
% For the "shifted" versions of Rayleigh and exponential,
% the mean of the sequence is calculated and subtracted.
% This is a quick and dirty way to provide a zero-mean but
% asymmetric sequence. In both cases the variance remains
% as specified.
%
% The "two-sided" version of a Rayleigh sequence is created
% by generating a Rayleigh sequence and negating every other
% value. It just provides another zero-mean, symmetric
% sequence (variance argument here is no longer valid).
%
% As in the original rpiid, only the first character of
% in_type is checked.
%

% Don Green
% NPS/DoD Distance Learning
% 24 April 2000/last updated 9 June 2000

pdf = in_type(1);

% Uniform distribution uses MATLAB's built-in rand function,
% and subtracts 0.5 to make it zero-mean. The sequence is
% then multiplied by the square root of 12 * (the desired
% variance) which sets the variance.
%
% Normal (Gaussian) distribution uses MATLAB's built-in
% randn function (which is already zero-mean, unity variance).
% Multiplying the sequence by the square root of the desired
% variance sets the variance to the correct value.

```

```

%
% Laplace distribution is generated by producing two independent
% exponential distributions and subtracting one from the other.
% The scale parameter of the Laplace distribution, b, is
% generated by taking the square root of half the desired
% variance. This parameter b is then used to generate the
% exponential distributions as  $-b \cdot \log(R)$  where R is a standard
% rectangular distribution and log is the natural logarithm.
% Perform the subtraction, and you have a Laplace distribution.
%
% A Rayleigh distribution may be generated by taking the
% square root of the sum of the squares of two Normal
% distributions. The b which is used to multiply the
% Normal sequences is derived from the expression for the
% variance of a Laplace sequence ( $\text{variance} = (2 - \pi/2) \cdot b^2$ ).
% For the modified version below, a for loop is then executed
% which negates the sign on every other data point to provide
% a zero-mean sequence. The "shifted version" is simply an
% ordinary Rayleigh distribution, with the mean subtracted,
% to provide an asymmetric, zero-mean sequence (variance is
% still as specified).
%
% Exponential distribution is generated as described in Laplace,
% above, except that the b used in  $-b \cdot \log(R)$  is simply the
% square root of the desired variance. As with the Rayleigh
% sequence, if a "shifted" version is desired the mean is
% calculated and subtracted. This forces the mean to zero
% and leaves the variance as desired.
%
% Most of the formulae above have been lifted from
% Statistical Distributions, 2nd Ed. by Evans, Hastings,
% and Peacock.

if ((pdf == 'u') | (pdf == 'U'))      % Uniform/rectangular distri-
bution
    uni = rand(nsamp,1) - 0.5;
    seq = uni.*sqrt(12*var);
    clear uni;

```

```

elseif ((pdf == 'n') | (pdf == 'N')) % Normal/Gaussian distribution
    norm = randn(nsamp,1);
    seq = norm.*sqrt(var);
    clear norm;
elseif ((pdf == 'l') | (pdf == 'L')) % Laplace distribution
    b = sqrt(var/2);
    uni1 = rand(nsamp,1);
    uni2 = rand(nsamp,1);
    exp1 = -b*log(uni1);
    exp2 = -b*log(uni2);
    seq = exp1 - exp2;
    clear b uni1 uni2 exp1 exp2;
elseif ((pdf == 't') | (pdf == 'T')) % "Two-sided" Rayleigh
    b = sqrt(var/(2 - pi/2)); % distribution
    norm1 = randn(nsamp,1);
    norm2 = randn(nsamp,1);
    norm1 = norm1.*(b);
    norm2 = norm2.*(b);
    seq = sqrt(norm1.^2 + norm2.^2);
    for index = 1:nsamp, % This stuff is for a "two-sided Rayleigh" pdf
        if (mod(index,2) == 0)
            seq(index,1) = seq(index,1) * -1;
        end
    end
    clear norm1 norm2 b index;
elseif ((pdf == 'r') | (pdf == 'R')) % Rayleigh distribution
    b = sqrt(var/(2 - pi/2));
    norm1 = randn(nsamp,1);
    norm2 = randn(nsamp,1);
    norm1 = norm1.*(b);
    norm2 = norm2.*(b);
    seq = sqrt(norm1.^2 + norm2.^2);
    clear norm1 norm2 b;
elseif ((pdf == 's') | (pdf == 'S')) % Shifted Rayleigh
    b = sqrt(var/(2 - pi/2));
    norm1 = randn(nsamp,1);
    norm2 = randn(nsamp,1);

```

```

    norm1 = norm1.*(b);
    norm2 = norm2.*(b);
    seq = sqrt(norm1.^2 + norm2.^2);
    seq = seq - mean(seq);
    clear norm1 norm2 b;
elseif ((pdf == 'e') | (pdf == 'E'))      % Exponential distribution
    b = sqrt(var);
    uni = rand(nsamp,1);
    seq = -b*log(uni);
    clear b uni;
elseif ((pdf == 'h') | (pdf == 'H'))      % Shifted exponential
                                         % distribution
    b = sqrt(var);
    uni = rand(nsamp,1);
    exp = -b*log(uni);
    seq = exp - mean(exp);
    clear b uni;
else
    disp('Error: in_type ',in_type,' not recognized.');
```

end

C. MATLAB script from Chapter III

```

% This script was used to generate the theoretical values of the
% colored signals (i.e., those produced when the
% mean-shifted exponential and mean-shifted Rayleigh distributed signals
% were passed through a filter before their bispectra were estimated.)
% It was not used to generate theoretical values for the Gaussian- and
% uniform-distributed signals, as the theoretical values in those two
% cases were zero across the board.
% The "expected_val" parameter was changed by hand to 5.04739 for the
% mean-shifted Rayleigh case, and the range of the "index" and "om"
% indices was changed for the larger bispectral estimates (i.e.,
% 512x512)

beta=0.1;
expected_val=16;
j
```

```

om1=linspace(-pi,pi,129);
om2=linspace(-pi,pi,129);
for index1=1:128
    for index2=1:128
%        om1=(index1/128)-(0.5+1/128);
%        om2=(index2/128)-(0.5+1/128);
        element1=1/(1-(beta*exp(j*(om1(index1)+om2(index2)))));
        element2=1/(1-(beta*exp(-j*om1(index1)))));
        element3=1/(1-(beta*exp(-j*om2(index2)))));

output(index1,index2)=conj(element1)*element2*element3*expected_val;
    end
end

```

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

1. Mysore R. Raghuvver and Chrysostomos L. Nikias, "Bispectrum Estimation: A Parametric Approach," *IEEE Trans. on Acoustics, Speech, and Signal Processing*, Vol. ASSP-33, No. 4, pp. 1213-1230, October 1985
2. Julius S. Bendat, *Principles and Applications of Random Noise Theory*, John Wiley & Sons, New York, 1958
3. Charles W. Therrien, *Discrete Random Signals and Statistical Signal Processing*, Prentice Hall, Englewood Cliffs, NJ, 1992
4. Jerry M. Mendel, "Tutorial on Higher-Order Statistics (Spectra) in Signal Processing and System Theory: Theoretical Results and Some Applications," *IEEE Proceedings*, vol. 79, No. 3, pp. 278-305, March 1991
5. Merran Evans, Nicholas Hastings, and Brian Peacock, *Statistical Distributions: Second Edition*, Wiley Interscience, New York, 1993
6. Ananthram Swami, Jerry M. Mendel, and Chrysostomos L. Nikias, *Higher-Order Spectral Analysis Toolbox*, The MathWorks, Inc., Natick, MA, 1995
7. Chrysostomos L. Nikias and Athina P. Petropulu, *Higher-Order Spectra Analysis: A Nonlinear Signal Processing Framework*, Prentice Hall, Englewood Cliffs, NJ, 1993
8. Alan V. Oppenheim and Alan S. Willsky with Ian T. Young, *Signals and Systems*, Prentice Hall, Englewood Cliffs, NJ, 1983

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical information Center
Ft. Belvoir, Virginia
2. Dudley Knox Library
Naval Postgraduate School
Monterey, California
3. Chairman, Code EC
Department of Electrical and Computer Engineering
Naval Postgraduate School
Monterey, California
4. Professor Charles W. Therrien, Code EC/Ti
Department of Electrical and Computer Engineering
Naval Postgraduate School
Monterey, California
5. Professor Ralph Hippenstiel
Department of Electrical Engineering
University of Texas, Tyler
Tyler, Texas