MODELLING AND SIMULATION OF NON-GAUSSIAN PROCESSES

A Dissertation

Submitted to the Graduate School
of the University of Notre Dame
in Partial Fulfillment of the Requirements
of the Degree of

Doctor of Philosophy

by

Kurtis Robert Gurley, B.S., M.S.

____
Ahsan Kareem, Director

Department of Civil Engineering
and Geological Sciences
April, 1997

MODELLING AND SIMULATION OF NON-GAUSSIAN PROCESSES

Abstract

by

Kurtis Robert Gurley

The assurance of the safety and reliability of structures subjected to environmental loads requires the consideration of structural response due to severe loads, where the statistical description of input differs from Gaussian, and may lead to increased expected damage. In the context of reliability analysis and fatigue, an accurate representation of input forces in the extreme region is essential for meaningful results.

Among a host of techniques developed for the analysis of nonlinear structural response, simulation methods are gaining popularity as computational efficiency increases. Implementation of these methods require simulated load time histories with specific statistical and spectral characteristics. When the system loads deviate significantly from Gaussian, techniques for their accurate simulation must be available. This is particularly significant when considering the extreme response of these systems, which are sensitive to the tail region of the probabilistic description of input.

This research concerns the development of new methods for the simulation of non-Gaussian processes, geared toward the accurate representation of a wide range of extreme environmental conditions. Several techniques developed during the early stages of the work are first considered, followed by the four main components of the dissertation.

Methods are developed to model the tail region of the probability distribution of measured non-Gaussian data. These models are used to describe the desired non-Gaussian characteristics for simulation of the measured data.

A simulation method is developed, referred to as spectral correction, which accurately represents both the non-Gaussian characteristics, and the distribution of energy with respect to frequency. This method is demonstrated to perform as well, and, in some cases, better than existing methods of non-Gaussian simulation. Additionally, spectral correction is shown to perform well in cases where the existing methods always fail.

Spectral correction is extended to simulate several correlated non-Gaussian processes simultaneously (multi-variate simulation), and to simulate non-Gaussian realizations at one location based on measurements at other locations (conditional simulation).

These tools will aid in the analysis of nonlinear structural systems and their components by more accurately representing the fluctuating non-Gaussian loads to which they are exposed.

Acknowledgments

I wish to express my sincere gratitude to my research director, Dr. Ahsan Kareem. His professional guidance has been invaluable for the completion of this dissertation, and his high standards, encouragement, and support have been instrumental in my professional development. I thank him also for showing me there’s always a place for friendship.

Thanks are also due to the other members of the faculty in the Department of Civil Engineering and Geological Sciences. Many have contributed, in ways large and small, to this achievement. Thanks to everyone for the good advice, critiques, encouragement, and the examples you set. The assistance of the department staff has also been vital. Thanks to Tammy, Judy and Denise.

The Office of Navel Research, the Department of Education, and the University of Notre Dame are gratefully acknowledged for their financial support.

The love of my family and friends has been my constant source of inspiration. I learned to ask why from my parents, taught to never give up by Kevin, Kim and Kathy, and pushed by Kyle, for whom the bar is never high enough. The Boys have kept me sane, grounded, and happy for all these years, thanks guys.

Table of Contents

List of Figures

List of Tables

CHAPTER 1

INTRODUCTION, MOTIVATION, ORGANIZATION

1.1 Introduction

The assurance of the safety and reliability of structures subjected to environmental loads requires the consideration of their extreme response. Of particular interest is the prediction of structural response due to severe loads, where the statistical description of input differs significantly from Gaussian, and may lead to increased expected damage and higher fatigue effects. Additionally, nonlinear systems under either Gaussian or non-Gaussian input will result in a non-Gaussian response. Thus, in the context of reliability analysis for serviceability and / or survivability, an accurate probabilistic representation of input forces and structural response in the extreme region is essential for arriving at meaningful results.

1.2 Motivation

Significant effort has been put forth in recent decades toward the solution of the response of mechanical systems with uncertainty in the input, system parameters, or both.

A host of numerical techniques and analytical approximations have been developed to address these problems (e.g. statistical linearization, quadratization, and cubicization, stochastic averaging, perturbation, closure techniques), however, a large class of problems are still only tractable through Monte Carlo Simulation-based methods [12]. These include highly nonlinear and / or complex structures, and some forms of failure analysis. Monte Carlo simulation remains the preferred tool for the validations of the various solution techniques.

For probabilistic analysis of complex nonlinear systems, Monte Carlo simulation is not a viable solution method unless the system input can be properly represented. In some cases, non-Gaussian environmental input may appropriately be reduced to Gaussian loads through the central limit theorem. An example is the integral effect of wind loads on overall building response. However, for cladding loads the Gaussian assumption is not valid and loads remain strongly non-Gaussian, especially in the regions of separated flow.

In situations where the system response is a function of loads that deviate significantly from Gaussian, techniques for the accurate simulation of these loads must be available in order to apply Monte Carlo simulation. Examples include offshore structural system response to severe non-Gaussian wave loading, cladding components of buildings subjected to wind pressure in the separation zone, and structural response to hurricane winds. This is particularly significant when considering the extreme response of these systems, which are sensitive to the tail region of the probabilistic description of input. Small deviations from the Gaussian distribution can result in significantly different failure characteristics in the system. This research concerns the development of new robust methods for simulation of non-Gaussian input.

Often times the non-Gaussian excitations acting on a structure cannot adequately be modeled as a point process at a single location. Large structural systems may be subjected to a number of spatially separated random loads not acting in perfect unison. In order to simulate the response of complex nonlinear multiple input systems under extreme conditions, the non-Gaussian loads must be simulated with the appropriate correlation among components. The robust simulation techniques in this work are extended to include the simulation of multiple correlated non-Gaussian realizations.

Simulated random signals at un-instrumented locations of a structure are often needed in wind, wave and earthquake engineering. For example, malfunctioning equipment may leave a hole in a data set or information may be lacking due to a limited number of instruments. Dynamic analysis schemes which utilize integration techniques may require loading time histories at a number of unmeasured locations. Conditional simulation methods simulate realizations at desired locations conditioned on the measured data at other locations. New non-Gaussian conditional simulation methods are developed as an extension to the uni- and multi-variate non-Gaussian simulation techniques.

1.3 Organization

Chapter 2 provides background material on existing simulation methods. These include methods for uni- and multi-variate simulation of Gaussian and non-Gaussian processes. Chapter 3 addresses new simulation methods that were developed for the special case in which measured realizations of the input and / output are available. This work is separated into two cases: Static methods for cases where only system input is measured, and memory-based nonlinear system identification methods where system input and output are available. The restrictive nature of these methods are the motivation for developing new more robust simulation techniques.

Chapter 4 reviews two methods for modelling the probabilistic characteristics of non-Gaussian data, and presents modifications which improve the match between the PDF measured from data, and the PDF models. The results are applied as input to a new non-Gaussian simulation technique introduced in chapter 5, and referred to as the Spectral Correction method. This technique is capable of reproducing a range of target probabilistic and spectral characteristics equal to that of existing methods, and is also applicable in special situation where existing methods fail. Chapter 6 extends spectral correction to the simulation of multiple correlated realizations, and is shown to retain both the phase and amplitude information between realizations. Chapter 7 further expands multi-variate spectral correction to encompass conditional simulation, where missing non-Gaussian records can be simulated based on the measured records at other locations.

CHAPTER 2

SIMULATION METHODS

2.1 Introduction

The Monte Carlo method of system response simulation is widely applied for complex systems when alternative analytical methods are not applicable. This chapter presents existing techniques for the simulation of random processes necessary for Monte Carlo simulation methods. Gaussian, non-Gaussian, Uni-variate, multi-variate, stationary, and nonstationary processes are considered. The simulation of processes conditioned on existing measured records is also presented. This background material will lead to the introduction of new simulation methods presented in later chapters.

2.2 Uni-variate Gaussian Simulation

2.2.1 Spectral Representation

The well known spectral representation is based on a discretized model of the target power spectral density function for the desired process. The simulation consists of the superposition of harmonics at discrete frequencies that possess either deterministic amplitude and random phase (DARP), or random amplitude and random phase (RARP) [22, 67, 79, 81]. A zero mean stationary Gaussian realization $y(t)$ is simulated by RARP

$$y(t) = \sum_{i=1}^{n} (A_i \cos\omega_i t + B_i \sin\omega_i t)\,, \tag{2.1}$$

or DARP by [80, 86, 103]

$$y(t) = \sum_{i=1}^{n} C_i \cos(\omega_i t + \phi_i)\,. \tag{2.2}$$

In Eq. 2.1, A_i and B_i are independent Gaussian random variables with values at equally spaced discrete frequencies ω_i separated by $\Delta\omega$ with covariance

$$E[A_i^2] = E[B_i^2] = G(\omega_i)\Delta\omega\,, \quad E[A_i B_i] = 0\,. \tag{2.3}$$

$G(\omega_i)$ is the discrete approximation to the one sided power spectral density centered at discrete frequency ω_i, and representing the desired process energy in the frequency range $(\omega_i - \Delta\omega/2, \omega_i + \Delta\omega/2)$.

In Eq. 2.2, C_i is the discrete amplitude given by

$$C_i = \sqrt{2G(\omega_i)\Delta\omega}\,, \tag{2.4}$$

and ϕ_i is the ith realization of a uniformly distributed random variable from 0 to 2π. The superposition of harmonics with random phase produce a Gaussian process through the central limit theorem.

Eq. 2.2 may also be an alternative expression of Eq. 2.1 through the transformation of A_i and B_i to polar coordinates

$$C_i = \sqrt{A_i^2 + B_i^2}\text{, and } \phi_i = -\mathrm{atan}(B_i / A_i)\,. \tag{2.5}$$

Grigoriu [22] has shown that RARP is Gaussian for any value of n, while DARP approaches Gaussian as n approaches infinity. Further, DARP is strongly ergodic, while RARP is ergodic in the weak sense. Studies concerning the performance characteristics of RARP simulation vs. DARP simulation have shown that both models are adequate for the simulation of random ergodic processes under the condition that the number of frequencies used is large, and that $\Delta\omega$ is small enough to adequately model the desired power spectral density.

2.2.2 Time Series Models

Among several time series methods, auto-regressive moving average (ARMA) is a widely applied model which offers numerical efficiency and minimal computer storage. An ARMA system models the value of a time series at an instant in time in terms of previous time steps and a white noise

$$y(t) = -\sum_{k=1}^{p} a_k y(t-k) + \sum_{l=0}^{q} b_l u(t-l)\,, \tag{2.6}$$

where $u(t)$ is a white noise vector, and a_k and b_l are constant coefficients. The problem to be solved is both the appropriate values of the coefficients, and the level of dependence of $y(t)$ on previous $y(t-k)$ and noise $u(t-l)$, known as optimal model order (p, q).

There are several parameter estimation procedures developed in the literature [e.g. 15, 59, 63, 66, 70, 76], usually based on least squares or maximum likelihood approaches and their variations. The objective function to minimize is an error measure between the desired target power spectrum and the power spectrum resulting from the selected coefficients. One such function suggested by Mignolet and Spanos [70] is

$$err = \frac{1}{2\omega_c}\int_{-\omega_c}^{\omega_c} \left|D(e^{i\omega t})Q(\omega) - N(e_{i\omega t})\right|^2 d\omega, \tag{2.7}$$

where

$$D(z) = \sum_{k=0}^{p} a_k z^{-k}, \tag{2.8}$$

$$N(z) = \sum_{l=0}^{q} b_k z^{-l} \tag{2.9}$$

are used to represent $y(t)$ as the output of a discrete dynamic system with a transfer function

$$H(z) = D^{-1}(\omega)N(z), \tag{2.10}$$

and $Q(\omega)$ is a decomposition of the target power spectrum

$$G_{yy}(\omega) = Q^{*}(\omega)Q^{T}(\omega). \tag{2.11}$$

The choice of optimal order is based on the desired level of accuracy in the spectral representation. A commonly used procedure is found in Kozin [59], known as the information criterion.

Variations of adaptations of ARMA models have been developed to address for specific issues in Monte Carlo simulation. For example, Li and Kareem [63] combine ARMA methods and digital interpolation filters to adapt ARMA for multivariate wind load simulation at very small time steps required by the time integration scheme used to estimate system response to wind loads. Wave related processes are simulated using a host of parametric models to efficiently represent input and intermediate processes in the simulation of response [66].

Yamazaki and Shinozuka [102] have observed that although ARMA methods require smaller storage space due to their recursive nature, longer samples are necessary to achieve the same level of statistical accuracy as that of spectral methods.

2.3 Multi-Variate Gaussian Simulation

A large class of problems in stochastic analysis require the generation of multiple random processes which may correlated. The simulation of multivariate correlated processes has been accomplished in the literature either through decomposition of the statistical description of the correlation, the cross-correlation matrix in the time domain, or its fourier transform, the cross-spectral matrix. The methods addressed are the spectral representation method, the covariance decomposition method, and ARMA methods.

2.3.1 Spectral Decomposition

The simulation of M correlated variates requires knowledge of the auto-spectrum of each process, and the cross spectrum between each of the M variates. Each variate will simulated as a time series realization through the assumption of ergodicity. A Cholesky decomposition is used to represent the cross spectral matrix as [85, 100]

$$G(\omega) = \begin{bmatrix} G_{11}(\omega) & G_{12}(\omega) & \dots & G_{1M}(\omega) \\ G_{21}(\omega) & G_{22}(\omega) & \dots & G_{2M}(\omega) \\ : & : & : & : \\ G_{M1}(\omega) & G_{M2}(\omega) & \dots & G_{MM}(\omega) \end{bmatrix} = H(\omega)H^*(\omega) \;=\; , \qquad (2.12)$$

$$\begin{bmatrix} H_{11}(\omega) & 0 & \dots & 0 \\ H_{21}(\omega) & H_{22}(\omega) & \dots & 0 \\ : & : & : & : \\ H_{M1}(\omega) & H_{M2}(\omega) & \dots & H_{MM}(\omega) \end{bmatrix} \begin{bmatrix} H^{*}{}_{11}(\omega) & 0 & \dots & 0 \\ H^{*}{}_{21}(\omega) & H^{*}{}_{22}(\omega) & \dots & 0 \\ : & : & : & : \\ H^{*}{}_{M1}(\omega) & H^{*}{}_{M2}(\omega) & \dots & H^{*}{}_{MM}(\omega) \end{bmatrix}^{T}, \quad (2.13)$$

where $G_{ii}(\omega)$ is the auto-spectral density functions of the ith variate, and

$$G_{ik}(\omega) = \sqrt{G_{ii}(\omega)G_{kk}(\omega)}\gamma_{ik}(\omega) \quad (2.14)$$

is the cross spectral density between the ith and kth variates. Here $\gamma_{ik}(\omega)$ is the coherence function describing the linear correlation between the ith and kth variates, and is generally complex valued to account for varying phase lag at different frequencies.

Analogous to the uni-variate RARP simulation of Eq. 2.1, two independent Gaussian white noise vectors $\xi = \eta + j\zeta$ are generated with mean and variance

$$E[\eta] = E[\zeta] = 0\,, \quad (2.15)$$

$$E[\eta^2] = E[\zeta^2] = \Delta\omega\,. \quad (2.16)$$

The components of the Fourier series of the M processes are then constructed from

$$Y_i(\omega) = \sum_{k=1}^{i} H_{ik}(\omega)\xi_k(\omega)\,, \quad i = 1\dots M\,, \quad (2.17)$$

or in matrix form

$$\begin{bmatrix} Y_1(\omega) \\ Y_2(\omega) \\ : \\ Y_M(\omega) \end{bmatrix} = \begin{bmatrix} H_{11}(\omega) & 0 & \dots & 0 \\ H_{21}(\omega) & H_{22}(\omega) & \dots & 0 \\ : & : & : & : \\ H_{M1}(\omega) & H_{M2}(\omega) & \dots & H_{MM}(\omega) \end{bmatrix} \begin{bmatrix} \xi_1(\omega) \\ \xi_2(\omega) \\ : \\ \xi_M(\omega) \end{bmatrix}. \quad (2.18)$$

An inverse FFT now gives the $i = 1\dots M$ components $y_i(t)$ [84].

An alternative is to express the decomposed spectral matrix in polar form

$$H_{ik}(\omega) = |H_{ik}(\omega)|\exp(j\theta_{ik}) , \tag{2.19}$$

where

$$\theta_{ik} = \operatorname{atan}\left(\frac{Im[H_{ik}(\omega)]}{Re[H_{ik}(\omega)]}\right) . \tag{2.20}$$

The simulation through summation of cosines in the time domain is then

$$y_i(t) = \sum_{k=1}^{M}\sum_{l=1}^{N} |H_{ik}(l\Delta\omega)|\sqrt{\Delta\omega}\cos l\Delta(\omega t - \theta_{ik}(l\Delta\omega) + \Phi_{kl}) , \tag{2.21}$$

where Φ_{kl} are random uniformly distributed phase angles between 0 and 2π. The FFT may be applied to Eq. 2.21 for efficiency [12].

Deodatis [12] has recently introduced a variation of the above multivariate simulation technique which improves upon the ergodic properties by providing samples that precisely matches the temporal cross-correlation matrix. Li and Kareem [67] propose a digital filtering scheme to synthesize segments of time series simulated with a multivariate FFT method.

2.3.2 Covariance Matrix Decomposition

This method [e.g. 17, 50, 69, 102] transforms a set of independent Gaussian random variables into a set of M correlated random variables $\boldsymbol{Y}$ with a prescribed covariance matrix

$$C = \begin{bmatrix} C_{11} & C_{12} & \dots & C_{1M} \\ C_{21} & C_{22} & \dots & C_{2M} \\ : & : & : & : \\ C_{M1} & C_{M2} & \dots & C_{MM} \end{bmatrix} . \tag{2.22}$$

The eigenvalues Λ and eigenvectors Φ of C are found by solving

$$C\Phi = \Phi\Lambda. \tag{2.23}$$

A set of uncorrelated processes are introduced by

$$\boldsymbol{Z} = \Phi^T\boldsymbol{Y}, \tag{2.24}$$

with an covariance matrix of

$$C_Z = E[(\boldsymbol{Z}-E[\boldsymbol{Z}])(\boldsymbol{Z}-E[\boldsymbol{Z}])] = \Lambda . \tag{2.25}$$

A simulation of $\boldsymbol{Y}$, denoted $\boldsymbol{Y}^s$ is then achieved by simulating first a set on independent processes $\boldsymbol{Z}^s$ and inverting Eq. 2.24

$$\boldsymbol{Y}^s = (\Phi^T)^{-1}\boldsymbol{Z}^s. \tag{2.26}$$

2.3.3 Time Series Models

The extension of ARMA models for multivariate simulation is accomplished directly by rewriting Eq. 2.6 using vectors $\boldsymbol{y}(t), \boldsymbol{u}(t)$ to replace the scalar components of the simulated and white noise processes, and replacing the two sets of parameter vectors a_k, b_k with MxM matrices

$$\boldsymbol{y}(t) = -\sum_{k=1}^{p}\boldsymbol{a}_k\boldsymbol{y}(t-k)+\sum_{l=0}^{q}\boldsymbol{b}_l\boldsymbol{u}(t-l) , \tag{2.27}$$

where M is the total number of variates. The minimization of the error between the cross spectral matrix and the target cross spectral matrix is again the criterion to determine the order (p, q) and the value of the parameter matrices [20, 59, 63, 66, 70].

2.4 Non-Stationary Gaussian Simulation

Time dependent amplitude and frequency characteristics are sometimes observed in measured environmental loads. A typical example of which is seismic excitation. A number of approaches are available to include evolutionary properties in simulation.

2.4.1 Spectral Representation

Evolutionary power spectra are included in the multi-variate simulation by modulating the target power spectral matrix with a time frequency function $A_i(\omega, t)$, where the subscript i indicates the ith variate [75]. The spectral components in a multi-variate simulation are represented by

$$G_{ii}(\omega, t) = |A_i(\omega, t)|^2 G_{ii}(\omega), \tag{2.28}$$

$$G_{ik}(\omega, t) = A_i(\omega, t) A_k(\omega, t) \sqrt{G_{ii}(\omega) G_{kk}(\omega)} \gamma_{ik}(\omega) . \tag{2.29}$$

The components of the decomposed spectral matrix are now a function of time, $H_{ik}(\omega, t)$, and Eq. 2.21 is now written [13]

$$y_i(t) = \sum_{k=1}^{M} \sum_{l=1}^{N} |H_{ik}(l\Delta(\omega, t))| \sqrt{\Delta\omega} \cos(l\Delta\omega t - \theta_{ik}(l\Delta\omega) + \Phi_{kl}) . \tag{2.30}$$

The efficient FFT technique cannot be applied to non-stationary simulation in Eq. 2.30 due to the time/frequency dependence of $H_{ik}(l\Delta(\omega, t))$. However, a special case of non-stationary process which is uniformly modulated over all frequencies reduces $A_i(\omega, t)$ to $A_i(t)$. In this case the non-stationary process can be represented by a stationary processes multiplied with a uniform modulator

$$y_i(t) = A_i(t)y_i^s(t)\,, \tag{2.31}$$

where $y_i^s(t)$ is a stationary simulation generated using a standard spectral method and employing FFT.

Li and Kareem [65] also used the modulated stationary time series concept, where a nonstationary process is expressed as a sum of correlated stationary processes modulated by a deterministic time function. A later study presented a highly efficient multivariate nonstationary simulation method combining spectral representation and digital filtering techniques [68].

2.4.2 Time Series and Wavelet Methods

ARMA methods have also been applied to the simulation of non-stationary processes [15, 19, 59, 92]. In this case the parameters are time dependent and must be solved for at each time step using the same methods as with the stationary technique (see section 2.2.2). This can become time consuming for multivariate processes, where the two MxM matrices of parameters $a(t), b(t)$ must be found for every time step. This is not necessarily reason to abandon ARMA for non-stationary simulation, as this class of random process is typically of short duration.

Signal decomposition using a class of localized basis functions, or wavelets, has become a popular new method of nonstationary signal analysis. Gurley and Kareem [26] have applied wavelet decomposition to the simulation of nonstationary signals through a stochastic manipulation of the decomposition coefficients.

2.5 Gaussian Conditional Simulation Methods

The problem of conditional simulation is how to simulate one or more sample records at unmeasured spatial locations conditioned by recorded data at a set of measured locations, given a known covariance between measured and desired processes. Conditional simulation generates time histories at new locations when one or more time series are known, and extends existing records beyond the total sampling time for cases where conditioning time series are a sub-interval of the desired full length.

Fundamentally, two approaches have been introduced in which the simulation is either based on a linear estimation or kriging [60], or on a conditional probability density function (CPDF) [48]. Shinozuka and Zhang [87], show that these methods are equivalent in their final form, although the methodology is different in their development.

This section summarizes the kriging and CPDF methods, and also shows that the kriging method has been developed using probability theory [8]. This development further demonstrates the equivalence of kriging and CPDF.

2.5.1 Kriging Method (Best Linear Unbiased Estimator)

Kriging is a method developed first by Krige [60] to linearly interpolate a sample field based on observed data. Consider an n component random vector $\boldsymbol{V}$ with mean μ and covariance matrix $\boldsymbol{C}$. We have realizations of the first $n-1$ variates, while realizations of the nth variate are to be conditionally simulated The best linear unbiased estimate of an unknown stochastic variate based on the known variates is expressed as

$$V_n^e = \sum_{i=1}^{n-1} \lambda_{in} V_i \,, \tag{2.32}$$

where λ_{in} are the kriging weights which provide an unbiased estimate if

$$E[(V_n^e - V_n)] = 0 , \tag{2.33}$$

and a best linear estimate if

$$E[(V_n^e - V_n)^2] = E\left[\left(\sum_{i=1}^{n-1} \lambda_{in} V_i - V_n\right)^2\right] = \sum_{i=1}^{n-1}\sum_{j=1}^{n-1} \lambda_{in}\lambda_{jn} C_{ij} - 2\sum_{i=1}^{n-1} \lambda_{in} C_{in} + C_{nn} \tag{2.34}$$

is a minimum, where C_{ij} is the correlation between known variates, C_{in} is the correlation between known the unknown variates, and C_{nn} is the correlation of the unknown variate. Equations 2.33 and 2.34 are used to solve for the kriging weights in Eq. 2.32.

The Lagrange multiplier method is used to create a Hamiltonian based on Eq. 2.34 under the constraint of Eq. 2.33. This provides the kriging weights by solving

$$\sum_{i=1}^{n-1} \lambda_{in} C_{ij} = C_{jn}, \quad j = 1, \ldots n-1 . \tag{2.35}$$

The best linear unbiased estimator V_n^e is applied to simulation by using it in conjunction with its associated error between the exact and interpolated variate

$$V_n = V_n^e + E_n . \tag{2.36}$$

where E_n is the error

$$E_n = V_n - V_n^e . \tag{2.37}$$

In terms of a single realization, $e_n = v_n - v_n^e$ is not available, since v_n is the unknown process to be simulated. An independent set of realizations for both the known and unknown processes, i.e. $v_i^s, i = 1 \ldots n$, may be simulated using unconditional Gaussian multivariate simulation techniques and the covariance matrix C. The error e_n^s is available through knowledge of v_n^s and the calculation of v_n^{se}, and may be used to replace the unknown error in a realization of Eq. 2.36. This requires the statistical independence of the estimator and the error, which holds for Gaussian processes since the best linear estimator and its error are orthogonal [47]. The method is thus restricted to the conditional simulation of Gaussian processes to validate the replacement of the error with the simulation v^s.

A conditional simulation is then represented by

$$v_n | v_{1 \ldots n-1} = v_n^c = v_n^e + e_n^s , \tag{2.38}$$

where

$$v_n^e = \sum_{i=1}^{n-1} \lambda_{in} v_i , \tag{2.39}$$

$$e_n^s = v_n^s - \sum_{i=1}^{n-1} \lambda_{in} v_i^s , \tag{2.40}$$

and λ_{in} are solved for from

$$\sum_{i=1}^{n-1} \lambda_{in} C_{ij} = C_{jn}, \quad j = 1, \ldots n-1 . \tag{2.41}$$

Hoshiya [37-40] proposed that without loss of generality, the simulation if v^s and evaluation of v_n^{se} may be avoided by directly simulating e_n^s in Eq. 2.38. By virtue of the kriging estimate being unbiased,

$$E[E_n] = E[(V_n - V_n^e)] = 0 . \tag{2.42}$$

If the total number of variates is N, where $i = 1 \ldots n-1$ are measured records, and $i = n \ldots N$ are the unknown variates, the error variance between any two unknown locations n, m is [38]

$$E[E_n E_m] = E[(V_n - V_n^e)(V_m - V_m^e)] = C_{nm} - \sum_{i=1}^{n-1} \lambda_{in} C_{in} \ . \tag{2.43}$$

A simulation of the error using these statistics replaces the simulation of $v_i^s, i = 1 \ldots n$ used for e_n^s in Eq. 2.38. This is referred to as the modified kriging method.

For numerical efficiency, Hoshiya further suggests a step by step method of simulating one unknown process at a time, adding it to the known field for simulation of the next unknown process, rather than directly simulating all unknown processes at once. In this manner, the Cholesky or modal decomposition necessary for simulation of a multivariate error field can be avoided. The overall improvement in efficiency is only relevant when there are a large number of unknown locations to simulate.

Vanmarcke and Fenton [97] developed a non-stationary Gaussian conditional simulation technique by synthesizing segments of conditionally simulated realizations. Each segment is assumed stationary, with varying spectral contents between segments. The segments are then pieced together, the end of one with the start of the next, using a linear combination of the two segments near the transition. The complete synthesized realization is then conditioned on know records, with spectral contents which evolve from the first to final segments.

2.5.2 Conditional Probability Density Function Method (CPDF)

2.5.2.1 Development

This method was presented in its current form by Kameda and Morikawa [48]. As in the kriging method above, it will be presented in terms of a vector stochastic process $\boldsymbol{V} = [V_1, V_2, \ldots, V_{n-1}, V_n]$, where the nth component is to be simulated, and realizations of the $V_1, \ldots V_{n-1}$ components are available through measurements. The n component joint probability density for $\boldsymbol{V}$ with mean μ and covariance matrix $\boldsymbol{C}$ is

$$p_{\boldsymbol{V}}(\boldsymbol{v}) = [\sqrt{(2\pi)^n \boldsymbol{C}}]^{-1} \exp[-(\boldsymbol{v}-\mu)^T \boldsymbol{C}^{-1} (\boldsymbol{v}-\mu)/2] . \tag{2.44}$$

The $n-1$ joint density function may be obtained as a marginal density function by integrating Eq. 2.44

$$\begin{aligned} p_{V_1 \ldots V_{n-1}}(v_1 \ldots v_{n-1}) &= \int_{-\infty}^{\infty} p_{\boldsymbol{V}}(\boldsymbol{v}) dv_n = \\ &[\sqrt{(2\pi)^n \boldsymbol{C}}]^{-1} \sqrt{\frac{2\pi}{a_{nn}}} \exp\left\{ \frac{a_{nn}}{2} \left[\sum_{i=1}^{n-1} \frac{a_{in}}{a_{nn}} (v_i - \mu_i) \right]^2 - \frac{1}{2} \sum_{i=1}^{n-1} \sum_{j=1}^{n-1} (v_i - \mu_i)(v_j - \mu_j) a_{ij} \right\} \end{aligned} , \tag{2.45}$$

where a_{ij} are elements of the matrix $\boldsymbol{C}^{-1}$. Now with $p_{V_1 \ldots V_n}(v_1 \ldots v_n)$ and $p_{V_1 \ldots V_{n-1}}(v_1 \ldots v_{n-1})$, the conditional probability of $V_n | (V_1, \ldots V_{n-1})$ is

$$\begin{aligned} p_{V_n | (V_1, \ldots V_{n-1})}(v_n | v_{1 \ldots n-1}) &= \frac{p_{V_1 \ldots V_n}(v_1 \ldots v_n)}{p_{V_1 \ldots V_{n-1}}(v_1 \ldots v_{n-1})} = \\ &[\sqrt{(2\pi)^n \boldsymbol{C}}]^{-1} \exp\left[-\frac{1}{2} \frac{(v_n - \mu_n^c)^2}{(\sigma_n^c)^2} \right] \end{aligned} , \tag{2.46}$$

where

$$\mu_n^c = \mu_n + \sum_{i=1}^{n-1} c_{in}(v_i - \mu_i) , \tag{2.47}$$

$$(\sigma_n^c)^2 = C_{nn} - \sum_{i=1}^{n-1} c_{in} C_{in} , \tag{2.48}$$

and

$$c_{in} = \sum_{i=1}^{n-1} C_{ij}^{-1} C_{jn} . \tag{2.49}$$

Application of Eq. 2.46 then provides the conditional simulation using a Gaussian distribution whose mean and variance have been modified to account for the values of the known realizations.

When the total number of variates is N, where $i = 1 \ldots n-1$ are measured records, and $i = n \ldots N$ are the unknown variates, knowledge of the variance matrix C_{N-n+1} allows the conditional simulation of the unknown variates using modal or cholesky decomposition methods. Alternatively, as in the kriging method, v_n may first be simulated based on $v_{1 \ldots . n-1}$. The dimensions of C are then expanded by augmenting with C_{n+1}, and then simulation of v_{n+1} is performed using the $v_{1 \ldots n}$ known realizations. This continues until the final desired record v_N is simulation based on the $v_{1 \ldots N-1}$ known records.

Kameda and Morikawa [48] present this method in the frequency domain by deriving expressions for the conditional probability of the Fourier coefficients based on the Fourier coefficients of the known records. The derivation is based on the joint Gaussian distribution of Fourier coefficients, and is thus restricted to the conditional simulation of Gaussian processes.

2.5.2.2 Equivalence with Kriging

Referring to Eqs. 2.47-2.48, the CPDF technique is viewed as the simulation of a multi-normal probability density function

$$v_n^c = N(\mu_n^c, (\sigma_n^c)^2) . \tag{2.50}$$

The mean and variance are seen to be dependent on the parameter c_{in}, which is equivalent to the kriging weights (from comparison of Eqs. 2.49 and 2.41).

The form of the kriging simulation in Eq. 2.38 may be written using Eqs. 2.39, 2.40 as

$$v_n^c = v_n^e + e_n^s = \sum_{i=1}^{n-1} \lambda_{in} v_i + v_n^s - \sum_{i=1}^{n-1} \lambda_{in} v_i^s \,. \tag{2.51}$$

Taking expectations and using Eq. 2.43, the kriging method may be represented as

$$v_n^c = \mu_n^c + N(0, (\sigma_n^c)^2) = N(\mu_n^c, (\sigma_n^c)^2) \,, \tag{2.52}$$

which is the same expression in Eq. 2.50 for CPDF conditional simulation [87].

It has been suggested that although the methods are equivalent, the CPDF method is preferable due its basis in probability theory. The next section will show that kriging has been previously derived in an equivalent form to Eqs. 2.38-2.41 using probability theory in place of the concept of the best linear unbiased estimator in Eqs. 2.32-2.36.

2.5.3 A Probability Based Derivation of Kriging

Hoshiya’s work on the modified conditional simulation method [38] is based on the early kriging work [e.g. 60]. Borgman [7] also provides a conditional simulation technique, following a conditional probability derivation in Anderson [3]. Borgman makes no reference to kriging, but the approach is equivalent, and its probabilistic approach nicely bridges the kriging and CPDF methods.

Consider again an n component random vector $\boldsymbol{V}$ with mean μ and covariance matrix $\boldsymbol{C}$. $\boldsymbol{V}$ is a multivariate Gaussian random vector if its probability density follows

$$P_{\boldsymbol{V}}(\boldsymbol{v}) = [(2\pi)^{n/2}\sqrt{|\boldsymbol{C}|}]^{-1} \exp[-(\boldsymbol{v}-\mu)^T \boldsymbol{C}^{-1}(\boldsymbol{v}-\mu)/2] \,. \tag{2.53}$$

The shorthand notation of which is

$$\boldsymbol{V} = N(\mu, \boldsymbol{C}) . \tag{2.54}$$

Suppose that the multinormal variate $\boldsymbol{V}$ is partitioned into a pair of random vectors $\boldsymbol{V}_1$ and $\boldsymbol{V}_2$. Let the multi-variate normal distribution of these variables be denoted

$$p(\boldsymbol{V}) = p\begin{bmatrix}\boldsymbol{V}_1 \\ \boldsymbol{V}_2\end{bmatrix} = N\left(\begin{bmatrix}\mu_1 \\ \mu_2\end{bmatrix}, \begin{bmatrix}\boldsymbol{C}_{11} & \boldsymbol{C}_{12} \\ \boldsymbol{C}_{21} & \boldsymbol{C}_{22}\end{bmatrix}\right) , \tag{2.55}$$

where

$$E[\boldsymbol{V}_1] = \mu_1 , \tag{2.56}$$

$$E[\boldsymbol{V}_2] = \mu_2 , \tag{2.57}$$

$$E[(\boldsymbol{V}_1 - \mu_1)(\boldsymbol{V}_1 - \mu_1)^T] = \boldsymbol{C}_{11} , \tag{2.58}$$

$$E[(\boldsymbol{V}_2 - \mu_2)(\boldsymbol{V}_2 - \mu_2)^T] = \boldsymbol{C}_{22} , \tag{2.59}$$

$$E[(\boldsymbol{V}_1 - \mu_1)(\boldsymbol{V}_2 - \mu_2)^T] = \boldsymbol{C}_{12} . \tag{2.60}$$

If a sample of $\boldsymbol{V}_1$ is measured and denoted as $\boldsymbol{v}_1$, then it is the conditional simulation of $\boldsymbol{V}_2$ based on the measured record $\boldsymbol{v}_1$ that is desired. The conditional probability law for $\boldsymbol{V}_2$ given $\boldsymbol{V}_1 = \boldsymbol{v}_1$ is multivariate normal of the form

$$p(\boldsymbol{V}_2 | \boldsymbol{V}_1 = \boldsymbol{v}_1) = N(\mu_2 + \boldsymbol{C}_{12}^T \boldsymbol{C}_{11}^{-1}(\boldsymbol{v}_1 - \mu_1), \boldsymbol{C}_{22} - \boldsymbol{C}_{12}^T \boldsymbol{C}_{11}^{-1} \boldsymbol{C}_{12}) , \tag{2.61}$$

which is equivalent to the CPDF method in Eqs. 2.47, 2.48, and the kriging method in Eqs. 2.38-2.41, 2.43.

- Proof for Eq. 2.61 [3]

We wish to transform $\boldsymbol{V}$ into two sub-vectors $\boldsymbol{Y}_1, \boldsymbol{Y}_2$ which are independent in order to easily find their joint density $p_{\boldsymbol{Y}_1, \boldsymbol{Y}_2}(\boldsymbol{y}_1, \boldsymbol{y}_2)$. Let $\boldsymbol{Y}_1, \boldsymbol{Y}_2$ be a function of the two correlated sub-vectors $\boldsymbol{V}_1, \boldsymbol{V}_2$ from Eq. 2.55

$$\boldsymbol{Y}_1 = \boldsymbol{V}_1 , \tag{2.62}$$

$$\boldsymbol{Y}_2 = \boldsymbol{M}\boldsymbol{V}_1 + \boldsymbol{V}_2 . \tag{2.63}$$

$\boldsymbol{M}$ is chosen such that $\boldsymbol{Y}_1$ and $\boldsymbol{Y}_2$ are uncorrelated by satisfying

$$E[(\boldsymbol{Y}_1 - E[\boldsymbol{Y}_1])(\boldsymbol{Y}_2 - E[\boldsymbol{Y}_2])^T] = 0 . \tag{2.64}$$

$$E[(\boldsymbol{V}_1 - E[\boldsymbol{V}_1])(\boldsymbol{V}_2 + M\boldsymbol{V}_1 - E[\boldsymbol{V}_2 + \boldsymbol{M}\boldsymbol{V}_1])] = \quad , \tag{2.65}$$

$$E[((\boldsymbol{V}_2 - E[\boldsymbol{V}_2]) + \boldsymbol{M}(\boldsymbol{V}_1 - E[\boldsymbol{V}_1]))(\boldsymbol{V}_1 - E[\boldsymbol{V}_1])] = \quad , \tag{2.66}$$

$$\boldsymbol{M}\boldsymbol{C}_{11} + \boldsymbol{C}_{12} = 0 . \tag{2.67}$$

Thus,

$$\boldsymbol{M} = -\boldsymbol{C}_{12}\boldsymbol{C}_{11}^{-1} . \tag{2.68}$$

Substitution of Eq. 2.68 into Eq. 2.63 and taking expectations gives the mean and covariance matrices for $\boldsymbol{Y}_1, \boldsymbol{Y}_2$

$$E\begin{bmatrix} \boldsymbol{Y_1} \\ \boldsymbol{Y_2} \end{bmatrix} = E\left[\begin{bmatrix} \boldsymbol{I} & 0 \\ -\boldsymbol{C}_{12}\boldsymbol{C}_{11}^{-1} & \boldsymbol{I} \end{bmatrix}\begin{bmatrix} \boldsymbol{V}_1 \\ \boldsymbol{V}_2 \end{bmatrix}\right] , \tag{2.69}$$

$$E[(\boldsymbol{Y} - E[\boldsymbol{Y}])(\boldsymbol{Y} - E[\boldsymbol{Y}])^T] = \begin{bmatrix} \boldsymbol{C}_{11} & 0 \\ 0 & \boldsymbol{C}_{22} - \boldsymbol{C}_{12}^T\boldsymbol{C}_{11}^{-1}\boldsymbol{C}_{12} \end{bmatrix} . \tag{2.70}$$

The joint probability density function of $\boldsymbol{Y}_1 = \boldsymbol{V}_1$ and $\boldsymbol{Y}_2 = \boldsymbol{M}\boldsymbol{V}_1 + \boldsymbol{V}_2$ is $p_{\boldsymbol{Y}_1, \boldsymbol{Y}_2}(\boldsymbol{y}_1, \boldsymbol{y}_2) = p_{\boldsymbol{Y}_1}(\boldsymbol{y}_1)p_{\boldsymbol{Y}_2}(\boldsymbol{y}_2)$, expanded as

$$p_{\boldsymbol{Y}_1, \boldsymbol{Y}_2}(\boldsymbol{y}_1, \boldsymbol{y}_2) = N(\mu_1, \boldsymbol{C}_{11})N(-\boldsymbol{C}_{12}\boldsymbol{C}_{11}^{-1}\mu_1 + \mu_2, \boldsymbol{C}_{22} - \boldsymbol{C}_{12}^T\boldsymbol{C}_{11}^{-1}\boldsymbol{C}_{12}) . \tag{2.71}$$

The joint distribution of $\boldsymbol{V}_1, \boldsymbol{V}_2$ can now be transformed from $p_{\boldsymbol{Y}_1, \boldsymbol{Y}_2}(\boldsymbol{y}_1, \boldsymbol{y}_2)$ by substituting $\boldsymbol{y}_1 = \boldsymbol{v}_1$ and $\boldsymbol{y}_2 = -\boldsymbol{C}_{12}\boldsymbol{C}_{11}^{-1}\boldsymbol{v}_1 + \boldsymbol{v}_2$ and using the jacobian (the identity matrix). The conditional density in Eq. 2.61 is then the quotient of $p_{\boldsymbol{V}_1, \boldsymbol{V}_2}(\boldsymbol{v}_1, \boldsymbol{v}_2)$ and $N(\mu_1, \boldsymbol{C}_{11})$ [3].

- end proof

Using Eq. 2.61, a conditional simulation is provided by

$$(\boldsymbol{V_2}|\boldsymbol{V}_1 = \boldsymbol{v}_1) = \boldsymbol{C}_{12}^T\boldsymbol{C}_{11}^{-1}(\boldsymbol{v}_1 - \boldsymbol{V}^s{}_1) + \boldsymbol{V}^s{}_2 , \tag{2.72}$$

where $\boldsymbol{V}^s{}_1, \boldsymbol{V}^s{}_2$ are unconditionally simulated known and unknown variates, respectively, and $\boldsymbol{v}_1$ is the known measured variate [8]. Taking the appropriate expected values of Eq. 2.72 lead to the mean and covariance in Eq. 2.61.

2.5.3.1 Equivalence with Kriging

Equations 2.63 and 2.68 can be seen as parallel to Eqs. 2.32, 2.35, and 2.36, where M is analogous to the kriging weights λ_{in}. In this case, $\boldsymbol{V}_2$ in Eq. 2.63 represents the error E_n in Eq. 2.36, and has an equivalent covariance in Eq. 2.61 to that of the error in Eq. 2.43. Equation 2.72 can be seen then to be exactly Eq. 2.38 when using the original kriging simulation, i.e. equivalent to

$$v_n|v_{1\ldots n-1} = v_n^c = v_n^e + e_n^s , \tag{2.73}$$

where

$$v_n^e = \sum_{i=1}^{n-1} \lambda_{in} v_i \text{, and } e_n^s = v_n^s - \sum_{i=1}^{n-1} \lambda_{in} v_i^s \text{, and } \sum_{i=1}^{n-1} \lambda_{in} C_{ij} = C_{jn},\quad j = 1, \ldots n-1 . \tag{2.74}$$

CPDF is often preferred for its basis on probability theory rather than a best linear unbiased estimate, but it is shown above that kriging can alternatively be derived based on probability theory.

2.5.4 Frequency Domain Gaussian Conditional Simulation

Conditional simulation can be applied in either the time or frequency domain. The section addresses frequency domain simulation, which is applicable for generating realizations of time series at unmeasured locations $n \ldots N$ based measured records at other spatial locations $1 \ldots n-1$. It is assumed that the spectral density matrix between the $n-1$ known and the $N-n+1$ unknown locations is available.

Referring to the simulation Eq. 2.72, it is convenient now to change the notation to explicitly denote the known and unknown components. Equation 2.72 is replaced with

$$(\boldsymbol{V}_u | \boldsymbol{V}_k = \boldsymbol{v}_k) = \boldsymbol{C}_{ku}^T \boldsymbol{C}_{kk}^{-1} (\boldsymbol{v}_k - \boldsymbol{V}^s{}_k) + \boldsymbol{V}^s{}_u \ , \tag{2.75}$$

where now the previous subscript 1 used to denote the known variates is replaced with k, and the subscript 2 used to denote unknown variates is replaced with u.

The covariance matrices C_{kk} and C_{ku} are represented in the frequency domain as

$$C_{kk} = \begin{bmatrix} G_{11} & G_{12} & \ldots & G_{1,n-1} \\ G_{21} & G_{22} & \ldots & G_{2,n-1} \\ : & : & : & : \\ G_{n-1,1} & G_{n-1,2} & \ldots & G_{n-1,n-1} \end{bmatrix}, \tag{2.76}$$

and

$$G_{ku} = \begin{bmatrix} G_{1,n} & G_{1,n+1} & \ldots & G_{1,N} \\ G_{2,n} & G_{2,n+1} & \ldots & G_{2,N} \\ : & : & : & : \\ G_{n-1,n} & G_{n-1,n+1} & \ldots & G_{n-1,N} \end{bmatrix}. \tag{2.77}$$

Where the notation C is maintained to correspond with Eq. 2.72. The submatrices in Eqs. 2.76 and 2.77 are composed of

$$G_{ii} = \begin{bmatrix} G_{ii} & 0 \\ 0 & G_{ii} \end{bmatrix}, \tag{2.78}$$

and

$$G_{ij} = \begin{bmatrix} Re[G_{ij}] & Im[G_{ij}] \\ -Im[G_{ij}] & Re[G_{ij}] \end{bmatrix}. \tag{2.79}$$

The frequency domain components of the know and unknown variates are represented by

$$V_k = A_k + iB_k\,, \quad V_k^s = A_k^s + iB_k^s\,, \quad V_u = A_u + iB_u\,, \quad V_u^s = A_u^s + iB_u^s\,. \tag{2.80}$$

Equation 2.75 is now applied to conditionally simulate the Fourier components of the unknown processes

$$\begin{bmatrix} A_u(\omega) \\ B_u(\omega) \end{bmatrix} = C_{ku}^T(\omega) C_{kk}^{-1}(\omega) \left(\begin{bmatrix} A_k(\omega) \\ B_k(\omega) \end{bmatrix} - \begin{bmatrix} A_k^s(\omega) \\ B_k^s(\omega) \end{bmatrix} \right) + \begin{bmatrix} A_u^s(\omega) \\ B_u^s(\omega) \end{bmatrix}. \tag{2.81}$$

2.5.5 Time Domain Gaussian Conditional Simulation

The time domain application of conditional simulation is applied here to extend the length of a measured process. The measured realization $v_k(t)$ consists of discrete time components $t = 1 \ldots n-1$, and the unknown realization $v_u(t)$ is a continuation of the known realization over time $t = n \ldots N$. The simulated realizations V_k^s, V_u^s in Eq. 2.75 now consist of a simulation $v^s(1 \ldots N)$ based on the measured realization $v_k(1 \ldots n-1)$ that extends to time $t = N$. This single simulated realization is partitioned into the know and unknown portions

$$v^s(1 \ldots N) = [v_k^s(1 \ldots n-1) \quad v_u^s(n \ldots N)]\,. \tag{2.82}$$

The covariance matrix now consists of elements from the covariance vector

$$C_{kk} = \begin{bmatrix} c_0 & c_1 & c_2 & \dots & c_{n-1} \\ c_1 & c_0 & c_1 & \dots & c_{n-2} \\ c_2 & c_1 & c_0 & \dots & c_{n-3} \\ : & : & : & : & : \\ c_{n-1} & c_{n-2} & c_{n-3} & \dots & c_0 \end{bmatrix}, \tag{2.83}$$

$$C_{ku} = \begin{bmatrix} c_n & c_{n+1} & c_{n+2} & \dots & c_N \\ c_{n-1} & c_n & c_{n+1} & \dots & c_{N-1} \\ c_{n-2} & c_{n-1} & c_n & \dots & c_{N-2} \\ : & : & : & : & : \\ c_1 & c_2 & c_3 & \dots & c_{N-n+1} \end{bmatrix}. \tag{2.84}$$

where the elements c_i are from either the Fourier transform of the power spectrum measured from the known realization $v_k(t)$, or a temporal calculation of the covariance

$$c_i = \frac{1}{t-i}\sum_{t=1}^{t-i} v_k(t)v_k(t+i)\,. \tag{2.85}$$

The time domain simulation of an extension to the known record v_k is given by

$$(v_u | V_k = v_k) = C_{ku}^T C_{kk}^{-1}(v_k - v_k^s) + v_u^s\,. \tag{2.86}$$

2.6 Non-Gaussian Simulation Methods: Introduction

Non-Gaussian environmental input often may appropriately be reduced to Gaussian loads through the central limit theorem, e.g. integral loads on a building under wind loads. However, for cladding loads the Gaussian assumption is not valid and loads remain strongly non-Gaussian, especially in the regions of separated flow.

In situations where the system response is a function of loads that deviate significantly from Gaussian, techniques for the accurate simulation of these loads must be available in order to apply Monte Carlo simulation. Examples include offshore structural system response to severe non-Gaussian wave loading, cladding components of buildings subjected to wind pressure in the separation zone, and structural response to hurricane winds. This is particularly significant when considering the extreme response of these systems, which are sensitive to the tail region of the probabilistic description of input. Small deviations from the Gaussian distribution can result in significantly different failure characteristics in the system.

The simulation of Gaussian processes has been widely explored for several decades, while non-Gaussian simulation has not been as widely addressed. This section presents several existing methods for simulating non-Gaussian uni-variate and multi-variate processes.

2.7 Tailored Phase Concepts

Specific characteristics in a random signal may be generated through the manipulation of the random phase, called phase tailoring [34]. This method has been used, for example, in the generation of large steep waves in an otherwise calm sea state [27].

Methods specifically developed for the simulation of non-Gaussian wind pressure with large excursion events can be found in Seong and Peterka [78]. Central to this technique is the use of the Fourier representation of the desired signal to separate the effects of the Fourier coefficient magnitudes from those of the phase. Exponential random numbers are used to generate a signal with the desired non-Gaussian characteristics. The phase from this signal is then applied to the Fourier magnitudes described by the desired power spectrum.

The phase is extracted from a skeleton signal generated through an exponential distribution $E(t)$ combined with an autoregressive term weighted by a parameter $0 \le a \le 1$

$$y(t) = \begin{array}{l} ay(t-1) + 0, \quad \text{with probability } b \\ ay(t-1) + E(t), \quad \text{with probability } 1-b \end{array}. \tag{2.87}$$

The phase from $y(t)$ is extracted through Fourier transform, and combined with the Fourier coefficient magnitude defined by the target power spectrum to simulate the final process $x(t)$. The parameter $0 \le b \le 1$ is chosen by iteratively simulating $x(t)$, and minimizing the mean square error between the skewness and kurtosis of $x(t)$ and the target values. A uniform phase shift $-\pi \le d \le \pi$ is added to the retrieved phase before implementation with the desired spectral magnitude.

The range of skewness and kurtosis values that can be matched simultaneously is limited, which restricts the method to a small class of processes representative of pressure in certain regions of flow separation.

2.8 Adaptive Correlation Methods

Typically, the goal of non-Gaussian simulation is to generate realizations with specified spectral and probabilistic characteristics. Various methods have been developed which are design to produce realizations which match a target power spectrum and a target PDF.

A common form of non-Gaussian simulation applies a nonlinear static transformation to a Gaussian signal generated using standard techniques including spectral methods, ARMA models, and filtered white noise. The nonlinear transformation is selected to match a target PDF, and distorts the auto-correlation and the associated power spectrum used to define the Gaussian process. In order to match a target PSD, the methods presented below account for the transformation distortion using both numerical and analytical means.

Additional methods are available [34], and are presented with several new simulation techniques in chapter 3.

2.8.1 Numerical Adjustment of Gaussian Auto-correlation

Yamazaki and Shinozuka [101] developed an iterative method which operates on a Gaussian simulation with a nonlinear static transformation. After each iteration, the distortion of the power spectrum is numerically compensated by updating the spectrum used to create the Gaussian process, referred to as the design spectrum. Sufficient iteration produces a non-Gaussian signal x, which matches both the target spectrum G^T and the target PDF. A more detailed account of the method follows.

A Gaussian signal $y(t)$ is generated based on an initial design spectrum G^D. This signal is then transformed to a signal x with the desired PDF using

$$x = F^{-1}(\Phi(y)),\tag{2.88}$$

where Φ is the Gaussian cumulative distribution function, and $F(x)$ is the target non-Gaussian cumulative distribution function. The resulting transformed process x no longer matches the design spectrum G^D. The iteration numerically updates the design spectrum used to generate y, reusing the same random phase, such that the final design spectrum produces a signal y that transforms to a realization of x which matches the target power spectrum G^T. At each iteration, i, an updated design spectrum $G^D(i+1)$ is determined based on the target spectrum G^T, the previous iteration’s design spectrum, $G^D(i)$, and the resulting spectrum G^x of the transformed process x by

$$G^D(i+1) = \frac{G^D(i)}{G^x(i)}G^T. \tag{2.89}$$

A schematic of this algorithm is shown in Fig. 2.1. It can be seen in the diagram that the iterations continue until the error between the target and output spectra is acceptable.

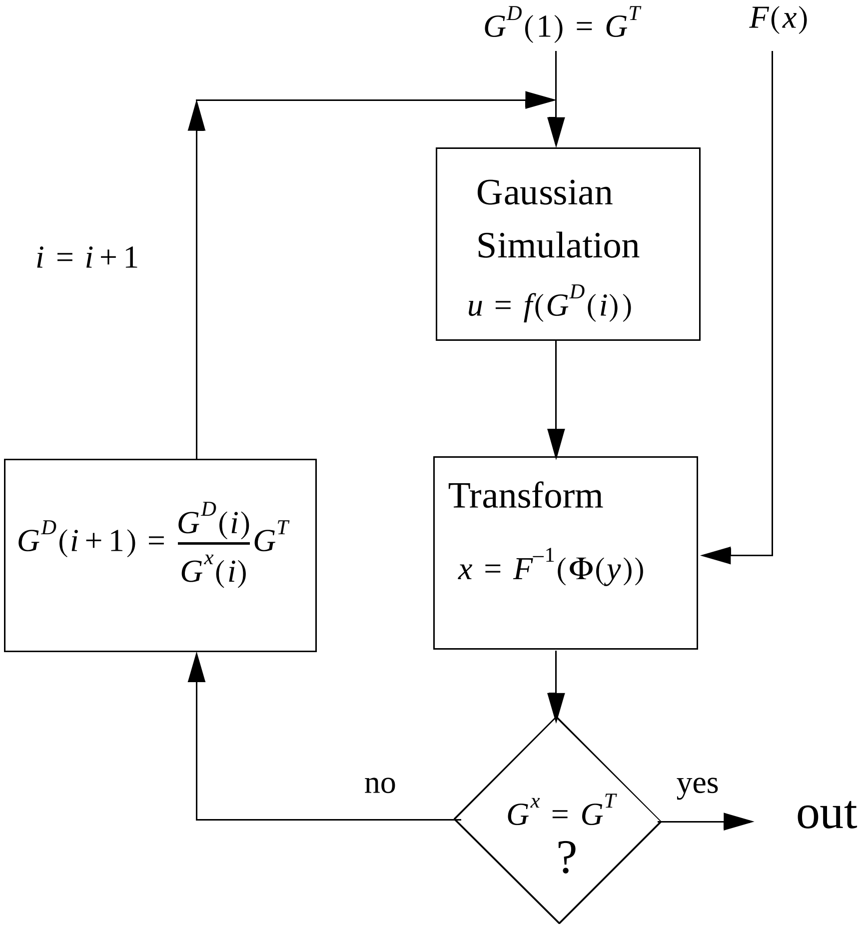


FIGURE 2.1 Schematic of the Yamazaki and Shinozuka method for iterative generation of non-Gaussian fields [101].

This technique works well under circumstances where there exists a design spectrum capable of producing the target spectrum after transformation through Eq. 2.88. However, cases exist where the required transformation cannot produce the target spectrum, regardless of the design spectrum.

Consider the normalized auto-correlation function for the Gaussian process $\rho(\tau)$ and non-Gaussian process $\zeta(\tau)$, defined by

$$\rho(\tau) = \frac{R_y(\tau)}{R_y(0)},\ R_y(\tau) = \frac{1}{2\pi}\int_{-\infty}^{\infty} G^D(\omega)\exp(i\omega\tau)d\omega, \tag{2.90}$$

$$\xi(\tau) = \frac{R_x(\tau)}{R_x(0)},\quad R_x(\tau) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}(g(y_1)-\mu_x)(g(y_2)-\mu_x)\phi(y_1, y_2, \rho(\tau))dy_1dy_2\ . \tag{2.91}$$

The Gaussian correlation coefficient ranges from -1 to 1, while the correlation coefficient for the transformed process may not cover that range. If the target spectrum has a corresponding auto-correlation function $\rho(\tau)$ that spans a restricted range for $\zeta(\tau)$, then no suitable design spectrum exists [23].

Figure 2.2 below demonstrates the relationship between $\rho(\tau)$ and $\zeta(\tau)$ for several examples of non-Gaussian static transformation. The examples include an exponential, a third order polynomial, a hermite polynomial, and a cubic transformation. Note that only symmetric transforms have a corresponding auto-correlation function $\zeta(\tau)$ which spans the entire possible range from -1 to 1.

2.8.1.1 Example Simulation: Wave Elevation

The Yamazaki and Shinozuka technique is illustrated in Fig. 2.3. It is desired to generate a non-Gaussian ocean wave process with a target spectrum G^T described by a Jonswap spectrum shown in Fig. 2.4 along with its corresponding normalized auto-correlation function. The first two plots in Fig. 2.3 are views of the target spectrum in black laid over the measured target spectrum in gray. The view on a linear scale shows an acceptable fit in the peak frequency range. The semi-log scale demonstrates that the method is unable to match the low frequency range of the target spectrum, instead producing a power spec-

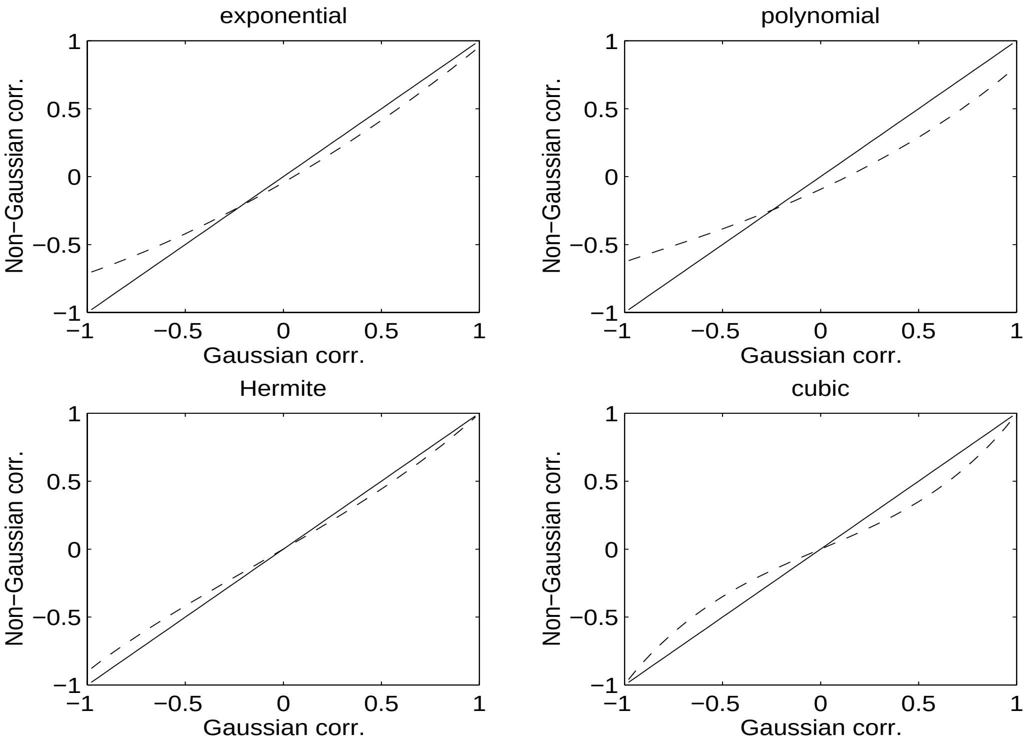


FIGURE 2.2 Samples of Gaussian correlation and non-Gaussian correlation after nonlinear static transformation.

trum with more energy in the low frequency range. The third plot in Fig. 2.3 is the target pdf and histogram of the simulation.

The normalized correlation function $\rho(\tau)$ corresponding to G^T contains values close to -1, as seen in Fig. 2.4. This is characteristic of a process with low damping, since the exponential decay of $\rho(\tau)$ is very gradual. The nonlinear transformation cannot produce a process with this level of damping, leading to the inflated low frequency region of the power spectrum, in a sense increasing the bandwidth of the power spectrum.

2.8.1.2 Restrictions

Yamazaki and Shinozuka's method [101] is intended for use with a user supplied, analytical, and invertible target non-Gaussian PDF. The paper includes an example with a tar-

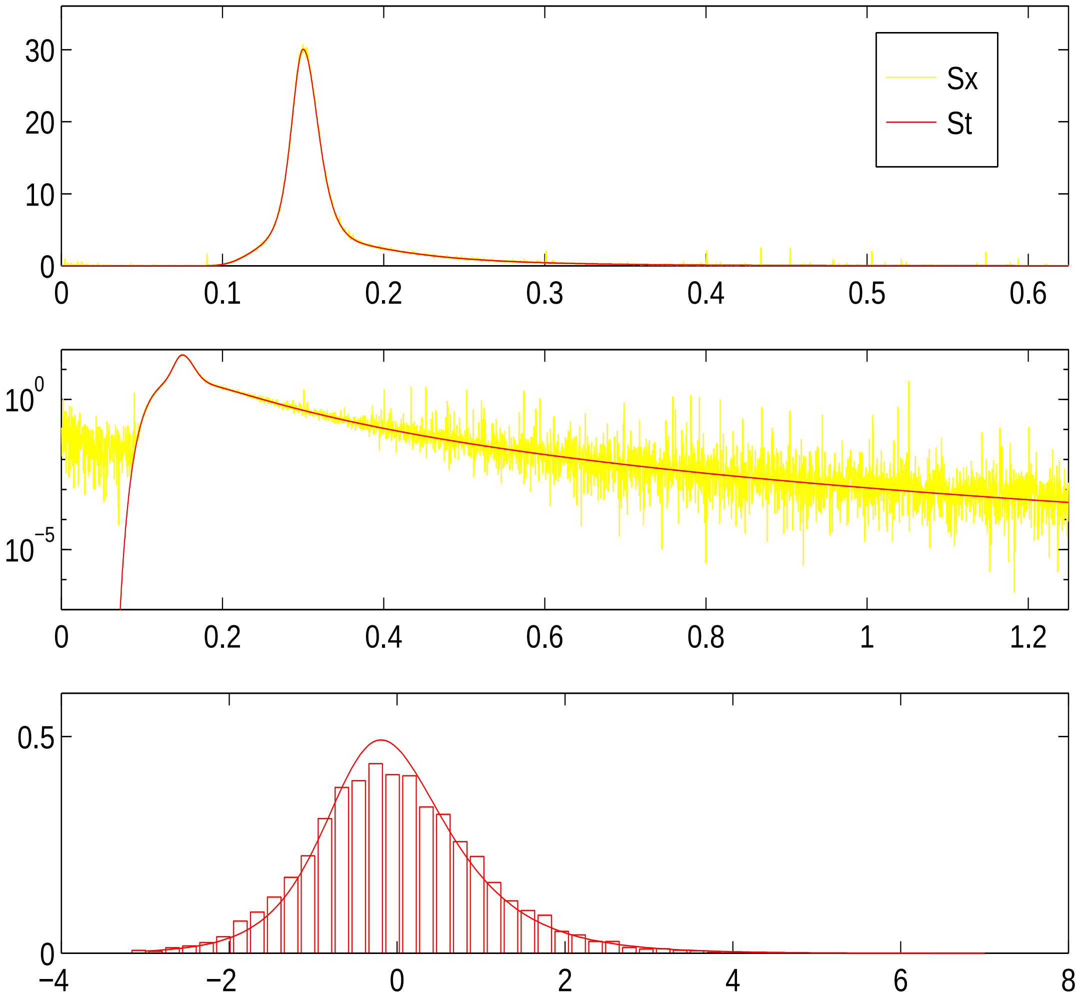


FIGURE 2.3 Power spectrum and PDF of a simulated non-Gaussian sea state using Yamazaki and Shinozuka's method.

get auto-correlation $\zeta(\tau)$ described by an exponential function. The range of values for the exponential function avoids problems associated with possibly unobtainable values of $\zeta(\tau)$ through transformation. The example in their paper illustrates that the method works well under these conditions. As shown in section 2.8.1.1, this may be interpreted as stating that the Yamazaki and Shinozuka method is appropriate for broad banded target spectra representing well damped processes.

In cases where the desired non-Gaussian simulation is based on observations of mea-

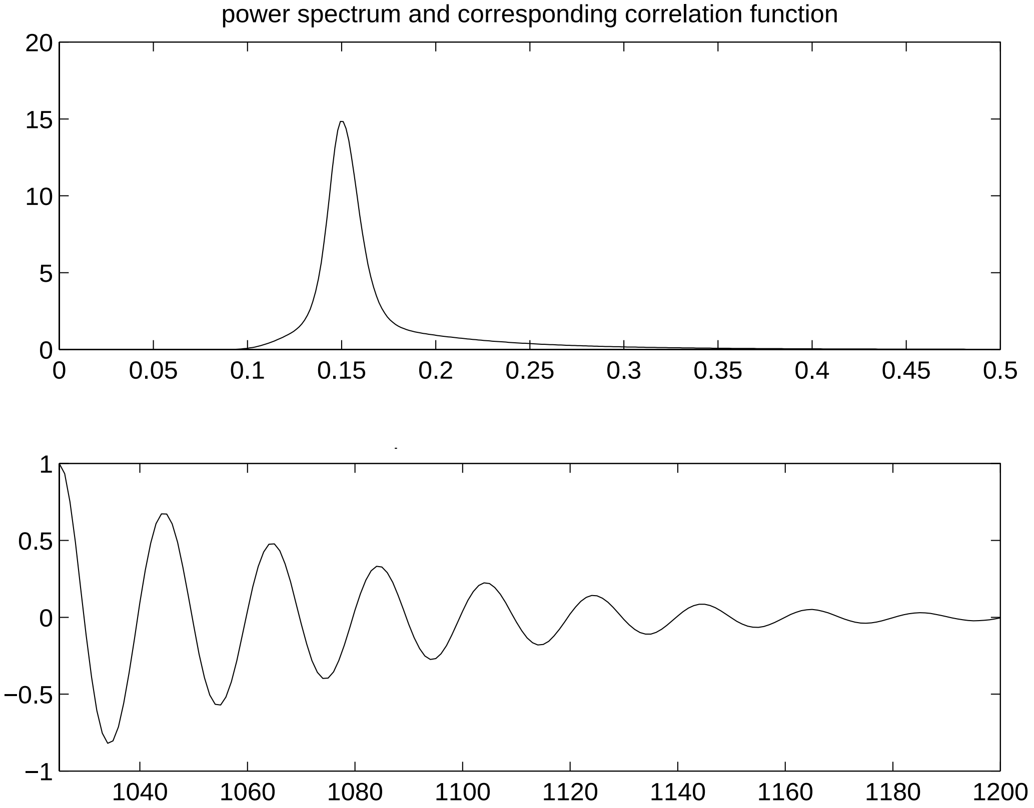


FIGURE 2.4 Target spectrum and auto-correlation function for the example corresponding to Fig. 2.3

sured data, or cases where the known analytical PDF is not invertible, numerical mapping can be employed as in the previous example. However, this may lead to inaccurate representation of the extremes. Thus, there is considerable advantage to developing an approximate analytical PDF model based on a polynomial transformation, which is easily applied to the Gaussian simulation.

Polynomial transformations also provide an analytical expression for $\zeta(\tau)$ in terms of $\rho(\tau)$ [2, 23, 32, 92, 98]. This can be determined through the properties of the expected values of Gaussian processes and the characteristic function, and may be used to determine analytically an appropriate initial design spectrum G^D to simulate the Gaussian process before transformation. This approach is used in the next section.

2.8.2 Analytical Adaptation of Gaussian Auto-correlation Function

Ammon [2] also applies a nonlinear static transformation to a simulated Gaussian realization to achieve a non-Gaussian realization. In this work the Gaussian process is simulated based on an appropriate design spectrum using an ARMA representation, and a polynomial transformation is then applied. The appropriate design spectrum is analytically derived in terms of the target spectrum for the output process. The method is divided into three sections.

First the coefficients of the polynomial transformation are determined such that the resulting non-Gaussian PDF has the desired characteristics. The coefficients c_k are sought from the representation of the non-Gaussian variate x in terms of the Gaussian process y by

$$x = \sum_{k=0}^{m} c_k y^k, \tag{2.92}$$

where m is the desired order of the polynomial. An objective function expressed in terms of x and the desired non-Gaussian characteristics is minimized to solve for the coefficients.

These coefficients c_k of the polynomial transformation define the relationship between the non-Gaussian auto-correlation function $R_x(\tau)$ and the Gaussian auto-correlation function $R_y(\tau)$ through Eq. 2.91. The Fourier transform of this relation provides an expression for the non-Gaussian PSD $G^x(\omega)$ in terms of the Gaussian design spectrum $G^D(\omega)$. This relation is inverted to express the design spectrum in terms of the non-Gaussian spectrum. Setting the non-Gaussian PSD $G^x(\omega)$ to the target spectrum $G^T(\omega)$ then provides the design spectrum $G^D(\omega)$ that is used to generate a realization of the Gaussian process y, such that transformation of y using Eq. 2.92 produces a realization of x whose PSD matches the target spectrum $G^T(\omega)$.

The third and final stage of the algorithm is to select the appropriate ARMA model order and coefficients to produce a Gaussian realization y with the design PSD determined in the previous step. The ARMA model may easily be replaced with other Gaussian simulation methods, as long as the design spectrum is matched in the simulation.

This method is designed such that the target spectrum is matched through an ensemble of realizations of x, rather than matching individual realizations as is the case for Yamazaki and Shinozuka. The limitations of its application are the same as that of Yamazaki and Shinozuka’s method. The design spectrum is derived analytically rather than numerically, but is still restrained by the range of auto-correlation values permissible by the static transformation. Certain combinations of target non-Gaussian PDFs and target spectra cannot be realized [23].

2.9 Summary of Other Methods

A variety of other methods exist for the simulation of non-Gaussian realizations. These include filtered poisson processes, ARMA models with non-white noise input, and alpha-stable processes. Each of these are briefly addressed for completeness. Chapter 3 will present several new methods developed during this study.

2.9.1 Filtered Poisson Processes

A filtered poisson process can be viewed as the response of a linear filter subjected to a train of random pulses arriving at random times [23]. Poirion [74] presents a model which achieves the desired power spectrum through the shape function describing the linear filter and the intensity of the poisson arrival times. Non-Gaussian characteristics are controlled by replacing the Gaussian pulse magnitude suggested by Grigoriu [23] with a non-Gaussian magnitude. The generation of realizations of the non-Gaussian magnitude variate can be complex. Thus, filtered poisson process simulation can be a great deal more cumbersome and less intuitive than spectral based methods.

2.9.2 Time Series Methods

Auto regressive moving average models have also been adapted to the simulation of non-Gaussian processes. A straightforward application was demonstrated in section 2.8.2, where the ARMA model is employed to generate a Gaussian realization matching a design power spectrum. In this case the ARMA model is not directly involved in the generation of the non-Gaussian signal, which depends entirely on the nonlinear static transformation applied to the Gaussian realization.

ARMA models which produce a non-Gaussian signal directly without transformation of the output have also been addressed by replacing the driver Gaussian white noise with non-Gaussian noise [e.g. 23, 62]. The ARMA model is rewritten now as

$$x(t) = -\sum_{k=1}^{p} a_k x(t-k) + \sum_{l=0}^{q} b_l u_{ng}(t-l) , \tag{2.93}$$

where u_{ng} represents the non-Gaussian driver noise, and a_k and b_l are the AR and MA coefficients, respectively.

The non-Gaussian PDF and associated higher order statistics of the simulation will be a function of the distribution of the non-Gaussian driver noise, the AR coefficients, and the MA coefficients. For low order ARMA models, recursion formulas may be used to derive relatively simple expressions relating the higher order output statistics with the model parameters and statistics of the driver noise. However, the problem of finding the probabilistic characteristics of the input driver noise corresponding to the target higher moments of the output is not guaranteed to have a solution, or be unique if a solution exists [23]. Additionally, the output PSD will also be a function of the driver noise distribution, complicating the solution procedure for identifying the AR and MA parameters.

The selection of model order, determination of the an appropriate non-Gaussian driver noise, and the solution for the ARMA coefficients to satisfy a target PSD can be rather complex. Although non-Gaussian ARMA models have been developed for specific problems of interest, its applicability as a general simulation method for non-Gaussian time series has yet to be demonstrated.

2.9.3 Alpha Stable Processes

Alpha stable processes are defined by a family of distributions described by the characteristic function

$$\varphi(\theta) = \exp\{i\mu\theta - \sigma^{\alpha}|\theta|^{\alpha}[1 + i\beta\,\mathrm{sign}(\theta)\varpi(|\theta|, \alpha)]\}, \tag{2.94}$$

$$\varpi(|\theta|, \alpha) = \begin{cases} -\tan(\pi\alpha/2), & \alpha \neq 1 \\ (2/\pi)\log|\theta|, & \alpha = 1 \end{cases}, \tag{2.95}$$

where $\alpha, \sigma, \beta, \mu$ are the stability index, scale, skewness, and shift parameters that describe the distribution type, range of likely values, departure from a symmetric distribution, and shift from zero, respectively. These constants are real, and restricted by, $\sigma \geq 0$, $|\beta| \leq 1$, $0 < \alpha \leq 2$. Closed form probability density functions are only available for a few special cases, for example, $\alpha = 2$ gives the Gaussian distribution.

The distinct characteristic of alpha stable processes is that, except when $\alpha = 2$, moments of order two and higher are infinite. Moments can be defined in terms of derivatives of the characteristic function by

$$E[X^q] = i^{-q}\frac{d^q}{d\theta^q}\varphi(\theta)\bigg|_{\theta = 0}. \tag{2.96}$$

It can be shown that the derivatives of order 2 and higher of the characteristic function in Eq. 2.94 are infinite at $\theta = 0$, leading directly to the infinite moments of order 2 and higher. This is the result of the restriction on alpha $0 < \alpha \leq 2$.

Figure 2.5 below shows several examples of the PDFs of alpha stable processes created using the Fourier transform relationship between the PDF and characteristic function. Each plot represents a collection PDFs with α values from 0.2 to 2 and fixed skewness parameter β. This parameter is varied with each graph clockwise, from a symmetric PDF $\beta = 0$ in the top left, to the limiting skewness value $\beta = 1$ in the lower left. Each figure contains a Gaussian distribution in black for comparison with the alpha stable PDFs in gray. The tails of this family of distributions are higher than Gaussian, increase as α decreases, and approach zero asymptotically. Significant weight remains in the tails even at a range far removed from the mean. Numerical integration of higher moments in the examples provided in Fig. 2.5 do not converge as the range of integration is extended.

It is relatively simple to generate sample realizations of an alpha stable process using a transformation of variables of both uniform and exponential distributions [see 23]. A time series realization can then be simulated as a filter response to these independent alpha stable realizations. The higher order moments are of course measurable from a finite time

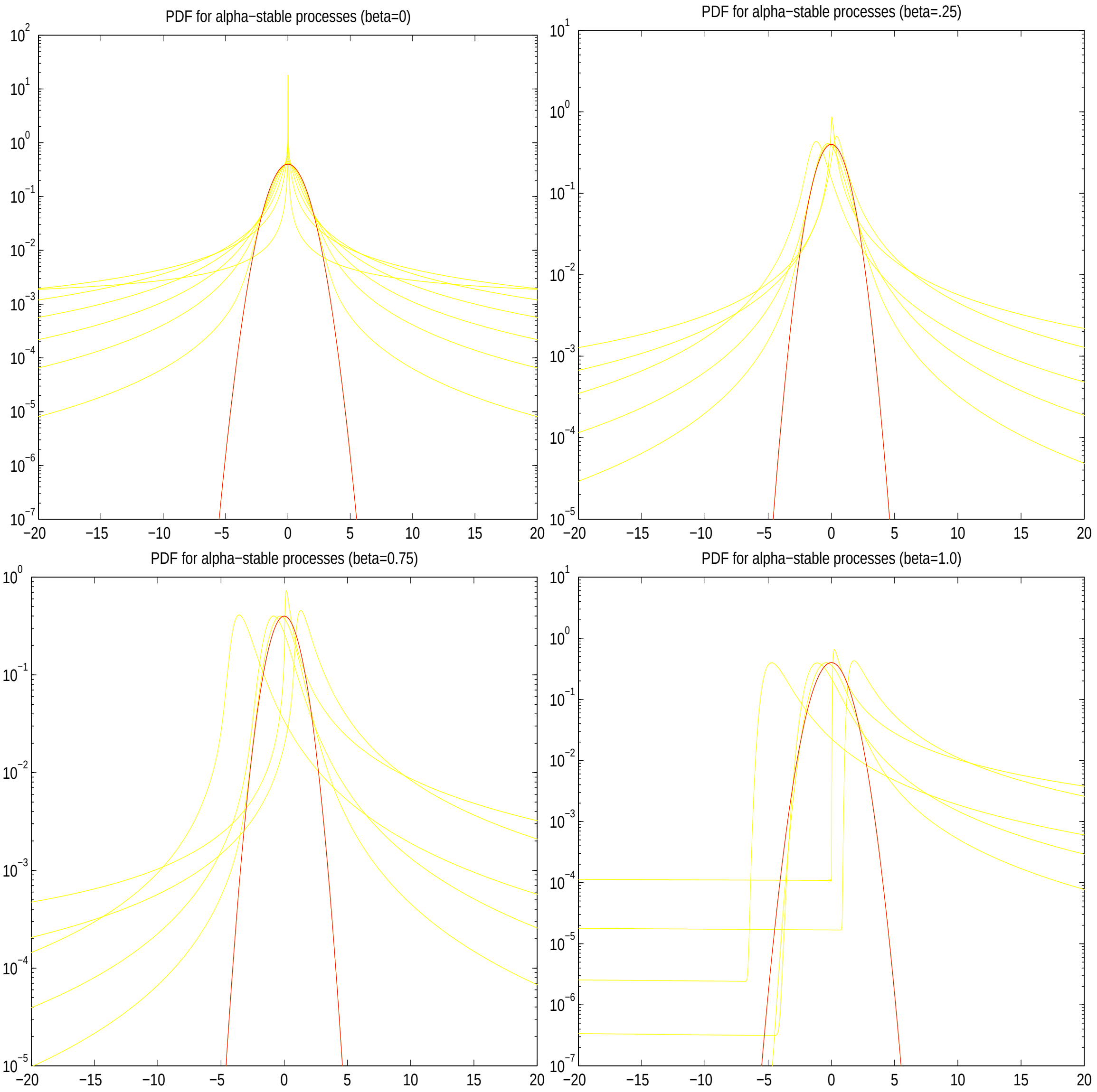


FIGURE 2.5 Examples of alpha stable distributions with varying alpha within each figure, and varying skewness between figures.

series realizations. However, as the length of the realization increases, estimated moments higher than order one continually increase, and do not converge with realization length.

Alpha stable processes have found application in the simulation of short duration processes with very heavy tails, typical of processes with small random deviations from the mean and rare very large excursions. While alpha stable processes are suitable for a class of non-Gaussian processes, they are not generally applicable to a wide range of non-Gaussian processes characterized by well defined higher moments over long time durations.

2.10 Non-Gaussian Conditional Simulation

Conditional simulation methods for multivariate Gaussian processes were presented in section 2.5. A natural extension of conditional simulation is to include non-Gaussian applications, which to date has received little attention.

Yamazaki and Shinozuka's method [101] was extended to the simulation of conditional non-Gaussian processes by Elishakoff et al. [18]. This iterative technique combines the unconditional multivariate non-Gaussian simulation technique in series with the Gaussian kriging conditional simulation method (section 2.5.1).

A schematic of this procedure is shown in Fig. 2.6. The method begins with the transformation of the known non-Gaussian field to a Gaussian field using CDF transformation (an inverse application of Eq. 2.88). The unmeasured locations are simulated conditioned on these known Gaussian realizations based on a design correlation function which is initially set to the final target correlation $R^D = R^{xT}$.

After Gaussian conditional simulation, the simulated locations are transformed back to the non-Gaussian domain, again using CDF transformation (Eq. 2.88). The error between the target correlation and the measured correlation distorted by the transformation is used to alter the design correlation for the Gaussian conditional simulation on the next iteration by

$$R^D(i+1) = R^D(i) + R^D(i)(R^{xT} - R^x) , \tag{2.97}$$

where i indicates the current iteration, and $i + 1$ is the next iteration. This continues until

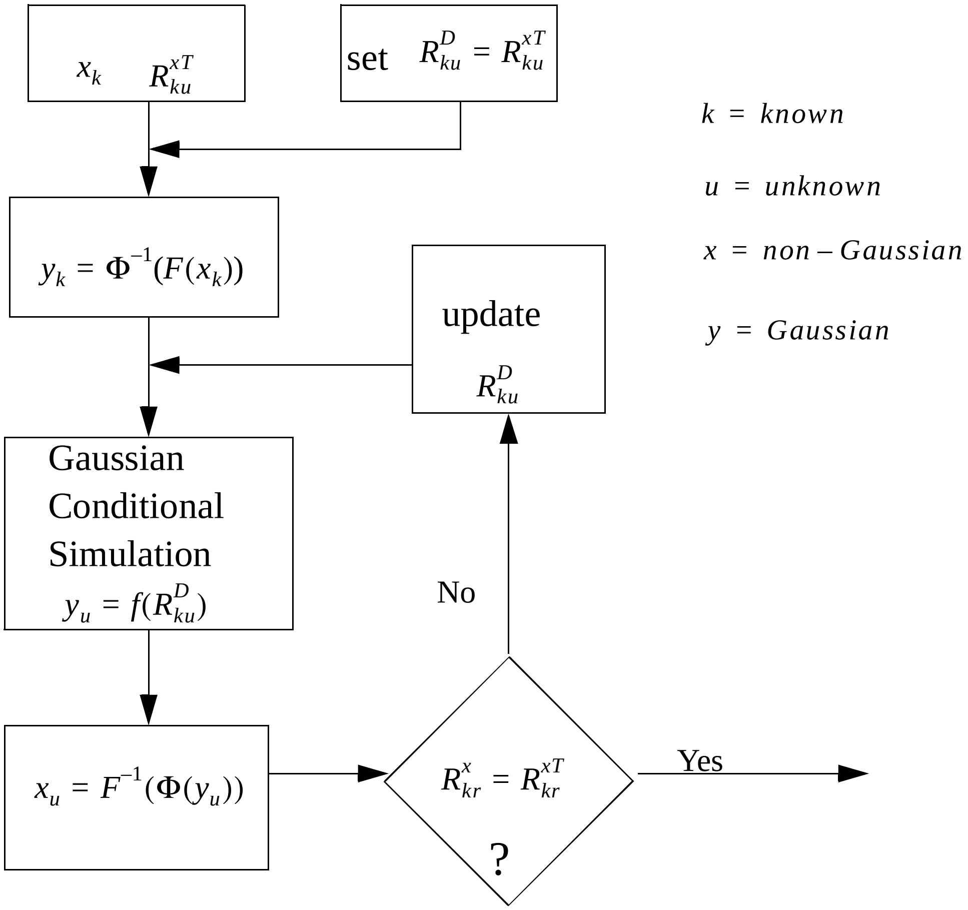


FIGURE 2.6 Schematic of Elishakoff's conditional simulation of non-Gaussian processes [18].

the error between desired and resulting non-Gaussian correlation is acceptable.

As is the case in the Yamazaki and Shinozuka paper [101], the authors demonstrate the applicability of the method through an example process characterized by an exponential auto-correlation, which avoids the difficulties associated with lightly damped (narrow banded) processes. The shortcomings of this method are identical to those of the Yamazaki and Shinozuka method, outlined in section 2.8.1.

2.11 Concluding Remarks

This chapter presents the necessary background on the simulation of Gaussian and non-Gaussian processes, and includes extensions to multi-variate simulation and conditional simulation. Many of the non-Gaussian simulation methods are restricted for use in special applications, and the more general methods have their own set of limitations.

The original contributions presented in later chapters are concerned with new, more robust techniques for simulating non-Gaussian multivariate processes, and the extension of these techniques to non-Gaussian conditional simulation. These new methods are intended for a wide variety of applications, and are designed to overcome the limitations of the existing general methods.

The development of these new methods occurred in several stages. The next chapter is concerned with some of the earlier work that complements work presented in later chapters.

CHAPTER 3

SIMULATION OF A CLASS OF NON-GAUSSIAN PROCESSES

3.1 Introduction

This chapter presents several non-Gaussian simulation techniques developed during the course of this research. The majority of the work herein represents the groundwork for the simulation methods presented in later chapters.

These simulation methods are based on measured samples and known characteristics of the system input and output [32]. First, static transformation techniques are developed to simulate non-Gaussian realizations by transforming the underlying Gaussian time or auto-correlation sample. An optimization procedure is introduced to overcome errors associated with a Hermite polynomial based transformation. This method is able to produce simulations which closely match the sample process PDF, and PSD. The restrictions of the methods are discussed in terms of motivation for improved techniques to be addressed in later chapters.

Memory-based simulation methods are also developed in order to better retain the higher-order spectral information not addressed by the static transformations. A second-order Volterra series is employed for simulation of non-Gaussian sea states, and a technique is introduced to allow Volterra-based simulation of very long realizations. Some simplifications to the Volterra model are discussed which increase the applicability of Volterra-based system identification under certain conditions. A neural network system identification model is employed for simulation of output when measured system input is available, and demonstrates the ability of memory-based simulation models to match higher order spectral characteristics.

3.2 Static Transformation Methods

3.2.1 Probability Transformation

An iterative non-Gaussian simulation method which utilizes the probability transformation $x = F^{-1}(\Phi(u))$ was discussed in the previous chapter in section 2.8.1. In this method, the spectrum used to produce a Gaussian simulation is iteratively corrected until the PSD of x matches the target spectrum after application of the probability transformation from u to x.

3.2.2 Correlation Distortion

The necessity for an iterative procedure may be eliminated if one begins with the target spectrum or auto-correlation of the non-Gaussian process and transforms it to the underlying correlation of the Gaussian process. This approach is referred to as the correlation-distortion method in stochastic systems literature (e.g., [11], [16], [46]). For a given static single-valued nonlinearity $x = g(u)$,where u is a standard normal Gaussian process, the desired auto-correlation of x in terms of u can be expressed as

$$R_{uu}(\tau) = \sum_{k=0}^{\infty} a_k^2 \rho_{xx}^k(\tau) ; \; a_k = \frac{1}{\sqrt{2\pi k!}} \int_{-\infty}^{\infty} g(\sigma u) exp\left(-\frac{u^2}{2}\right) H_k(u) du , \tag{3.1}$$

where ρ_{xx} is the normalized autocorrelation of the non-Gaussian process, and $H_k(u)$ is the k^{th} Hermite polynomial given by

$$H_k(u) = (-1)^k exp\left(\frac{u^2}{2}\right) \frac{d^k}{du^k}\left[exp\left(-\frac{u^2}{2}\right)\right] . \tag{3.2}$$

A simulation based on the schematic shown in Fig. 3.1 would eliminate the spectral distortion caused by the nonlinear transformation, since its inverse is employed to reverse the distortion.

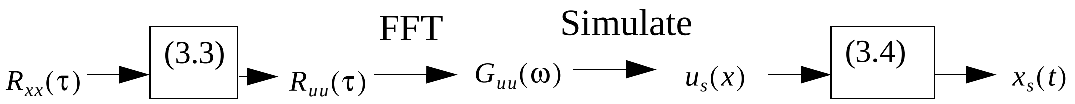


FIGURE 3.1 Schematic of the correlation distortion method

The simulation algorithm is as follows:

(i) Estimate the auto-correlation of the mean-removed normalized sample non-Gaussian process to be simulated $(R_{xx}(\tau))$;

(ii) transform to the auto-correlation of the underlying Gaussian process, $(R_{uu}(\tau))$, by solving for $R_{uu}(\tau)$ in the following equation [98],

$$R_{xx}(\tau) = \alpha^2[R_{uu}(\tau) + 2\hat{h}_3^2 R_{uu}^2(\tau) + 6\hat{h}_4^2 R_{uu}^3(\tau)] , \tag{3.3}$$

where

$\hat{h}_3 = \frac{\gamma_3}{4 + 2\sqrt{1 + 1.5\gamma_4}}$, $\hat{h}_4 = \frac{\sqrt{1 + 1.5\gamma_4} - 1}{18}$, $\alpha = \frac{1}{\sqrt{1 + 2\hat{h}_3^2 + 6\hat{h}_4^2}}$, and γ_3 and γ_4 are the skewness and kurtosis of the fluctuating process.

(iii) simulate a Gaussian process, u_s, using the spectrum, $G_{uu}(\omega)$, associated with $R_{uu}(\tau)$;

(iv) transform this simulated process u_s back to a non-Gaussian process using

$$x = \alpha[u + \hat{h}_3(u^2 - 1) + \hat{h}_4(u^3 - 3u)]; \tag{3.4}$$

(v) replace the mean and variance of the original parent process to produce a simulation, x_s of the original non-Gaussian process x.

Figure 3.2 compares a measured wind pressure signal with a single realization of a correlation distortion simulation of that signal in the top left and right figures, respectively. The bottom figures compare the power spectral density and pdf of the measured data with that from the ensemble average of 100 correlation distortion simulations.

TABLE 3.1 STATISTICS OF MEASURED WIND PRESSURE DATA AND ENSEMBLE AVERAGED SIMULATED DATA USING CORRELATION DISTORTION AND MODIFIED DIRECT SIMULATION.

	Std	Skewness	Kurtosis
Measured pressure Data	1.0	-0.8309	4.9940
Ensemble of 100 Correlation Distortion Simulations	.9927	-0.7960	5.6711
Ensemble of 100 Modified Direct Transformation Simulations	.9960	-0.8120	4.7676

The statistical moments of the normalized parent non-Gaussian signal are compared with the average moment statistics of the 100 realizations in table 3.1. The higher kurtosis in the Correlation Distortion simulations can be observed in both the time history and the pdf comparison. This distortion in the higher moments is a function of the departure of the parent non-Gaussian signal from Gaussian. The coefficients in the Hermite transformation provided in Eq. 3.3 have been derived based on the assumption of small departures from

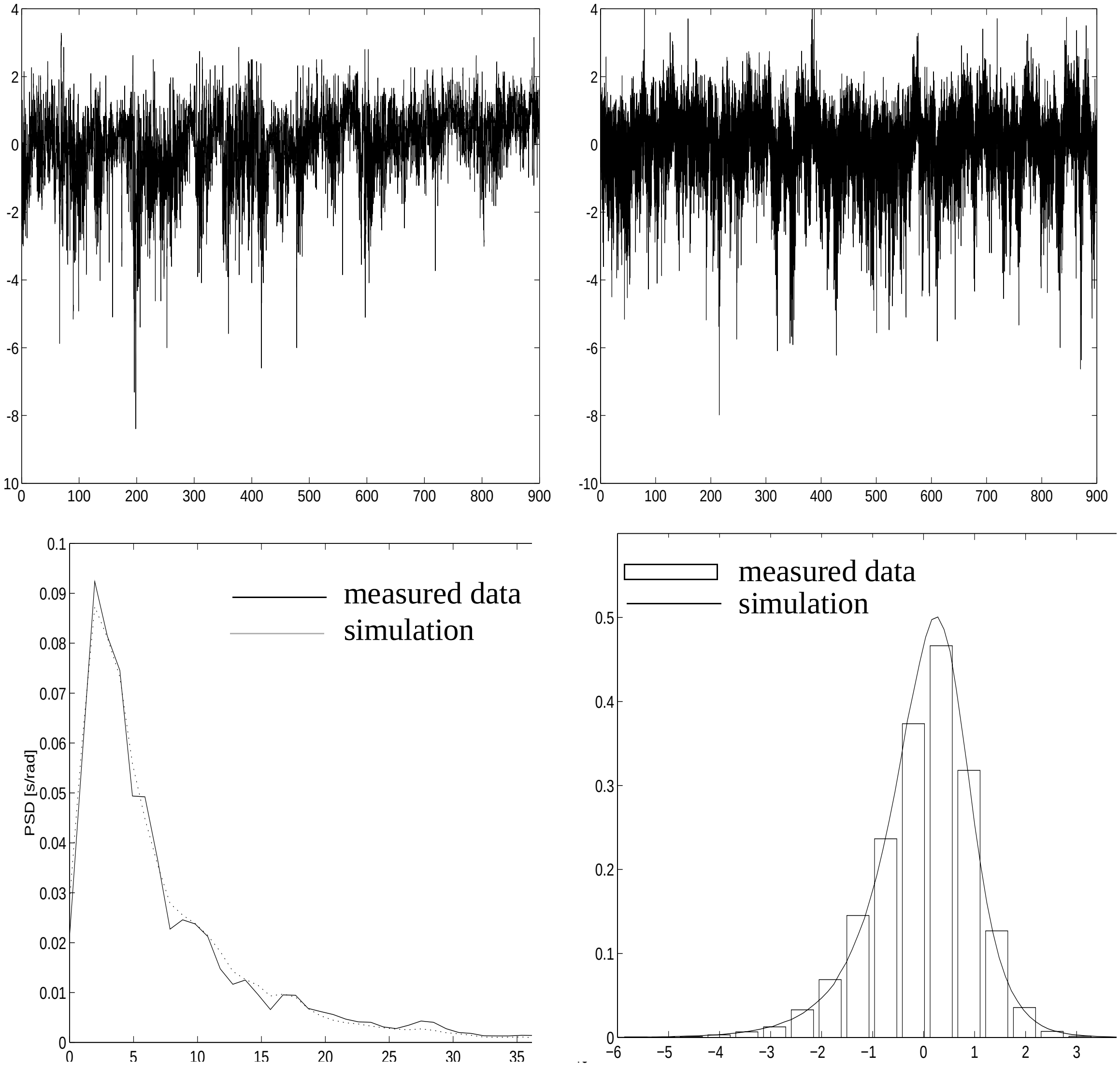


FIGURE 3.2 Measured wind pressure signal (top left), a correlation distortion simulation (top right), and power spectral density and pdf of the measured data and ensemble of 100 simulations.

Gaussian. The parent non-Gaussian signal in this example is highly non-Gaussian, making the application of Eq. 3.3 inappropriate. An alternative transformation method is now presented to overcome this restriction.

3.2.3 Modified Direct Transformation

An alternative to the correlation-based approach is to begin with a sample of a non-Gaussian time history rather than its autocorrelation. The non-Gaussian sample process, $x(t)$, is transformed to its Gaussian underlying form, $u(x)$, through

$$u(x) = [\sqrt{\xi^2(x) + c} + \xi(x)]^{1/3} - [\sqrt{\xi^2(x) + c} - \xi(x)]^{1/3} - a\,, \tag{3.5}$$

where $\xi(x) = 1.5b\left(a + \frac{x}{\alpha}\right) - a^3$, $a = \frac{\hat{h}_3}{3\hat{h}_4}$, $b = \frac{1}{3\hat{h}_4}$, $c = (b - 1 - a^2)^3$,

and the other parameters are defined after Eq. 3.3. The operation of Eq. 3.5 on the time history is referred to as direct transformation.

Direct application of Eq. 3.5 to the parent non-Gaussian pressure history will not provide the desired Gaussian underlying process. This is again due to the restriction of the coefficients of Eq. 3.5 to mild non-Gaussian cases.

A modification is now suggested to remove this distortion in direct transformation using Eq. 3.5. The governing parameters $\hat{h}_3, \hat{h}_4, a, b, c$, and thus $u(x)$, are dependent on the parameters representing skewness and kurtosis, γ_3 and γ_4. γ_3 and γ_4 may be treated as adjustable input parameters in order to force the transformed process, $u(x)$, to be Gaussian in terms of the third and fourth moments. Optimization of these two parameters is based on the minimization of the function

$$min(\gamma_{4_u}^2 + \gamma_{3_u}^2)\,, \tag{3.6}$$

where $\gamma_{3_u}, \gamma_{4_u}$ are the skewness and kurtosis of the transformed process $u(x)$. The optimized input parameters γ_3 and γ_4 now provide a Gaussian process in terms of third and fourth moments. The same parameters are used to transform back to a non-Gaussian simulation whose pdf and power spectral density closely match those of the sample process.

This technique is called the Modified Direct Transformation, and is diagramed in Fig. 3.3.

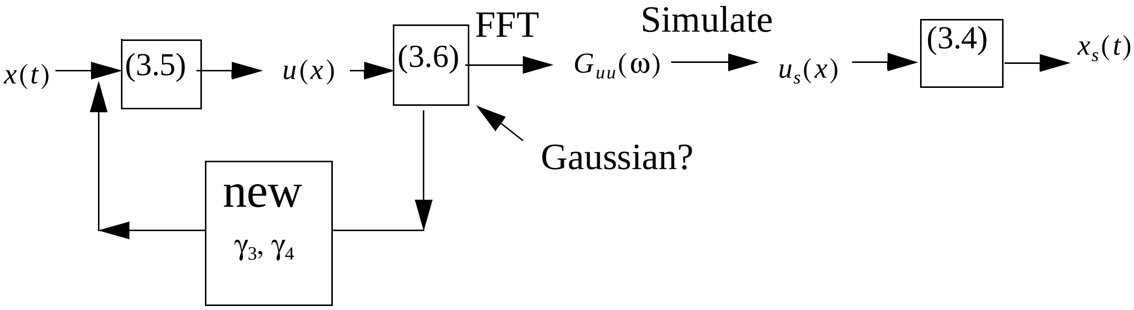


FIGURE 3.3 Schematic of the modified direct transformation method

The optimization of Eq. 3.5 is referred to as a backward Modified Hermite Transformation, and will be used in later chapters. The direct transformation from Gaussian to non-Gaussian using Eq. 3.4 can be optimized in the same way, and is referred to as forward Modified Hermite Transformation.

An example of the improvement afforded by the modified direct transformation method is demonstrated in Fig. 3.4. Again the measured pressure trace in the top left is simulated and displayed in the top right. The power spectral density and pdf of the data and simulations are shown in the bottom figures. Table 3.1 shows an improved ability to match higher order statistics compared with the correlation distortion method. By observing the positive and negative extreme behavior, as well as the fluctuation amplitude close to the mean, the modified direct simulation can be seen to emulate the characteristics of the measured process better than correlation distortion. This behavior is quantified by the kurtosis and standard deviation, which match well with the data (table 3.1).

A second example further demonstrates the performance of the modified direct transformation. The sample process to be simulated is the measured response of a model ten-

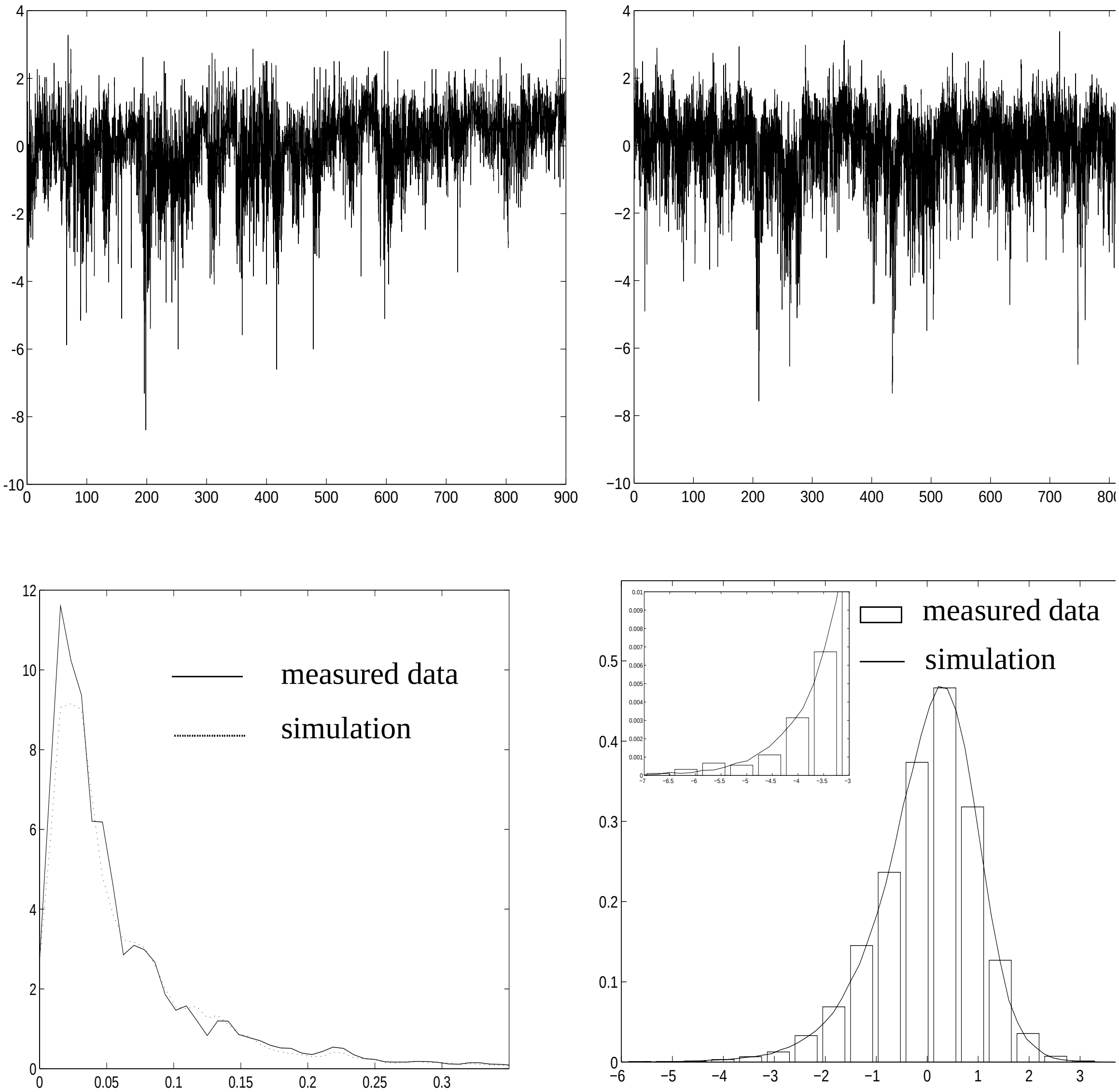


FIGURE 3.4 Measured wind pressure signal (top left), a direct transformation simulation (top right), and power spectral density and pdf of the measured data and ensemble of 100 simulations.

sion leg platform (TLP) under a random wind and wave field in a test facility. The response is highly non-Gaussian and has two dominant frequencies. Figure 3.5 shows a portion of the measured data and a realization using modified direct transformation simulation in the top and bottom plots, respectively. Figure 3.6 is a comparison of the PDF and PSD of the sample 100 realizations of the simulations. Table 3.2 lists statistics from the data and an ensemble of 100 realizations. The modified direct transformation provides realizations which match the sample PDF well, particularly in terms of positive and negative extreme behavior.

TABLE 3.2 STATISTICS OF MEASURED TLP RESPONSE DATA AND ENSEMBLE AVERAGED SIMULATED DATA USING 100 REALIZATIONS.

	Std	Skewness	Kurtosis
Measured TLP Data	1.0	0.8165	3.7455
Ensemble of 100 Modified Direct Simulations	0.9720	0.8187	4.2127

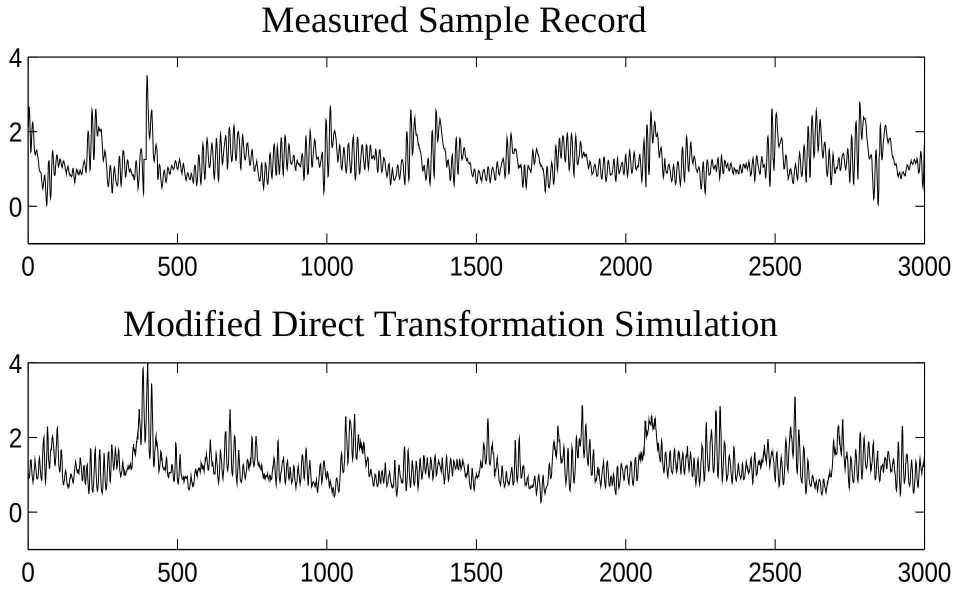


FIGURE 3.5 Measured TLP response, modified direct transformation and direct transformation simulations.

3.2.4 Restrictions of the Modified Direct Transformation Method

3.2.4.1 Restrictive Applicability

The Modified Direct Transformation method is restricted to the simulation of records for which a sample exists. This is not the case in many practical cases where a non-Gaussian simulation is desired. A more general approach has been taken by a previously dis-

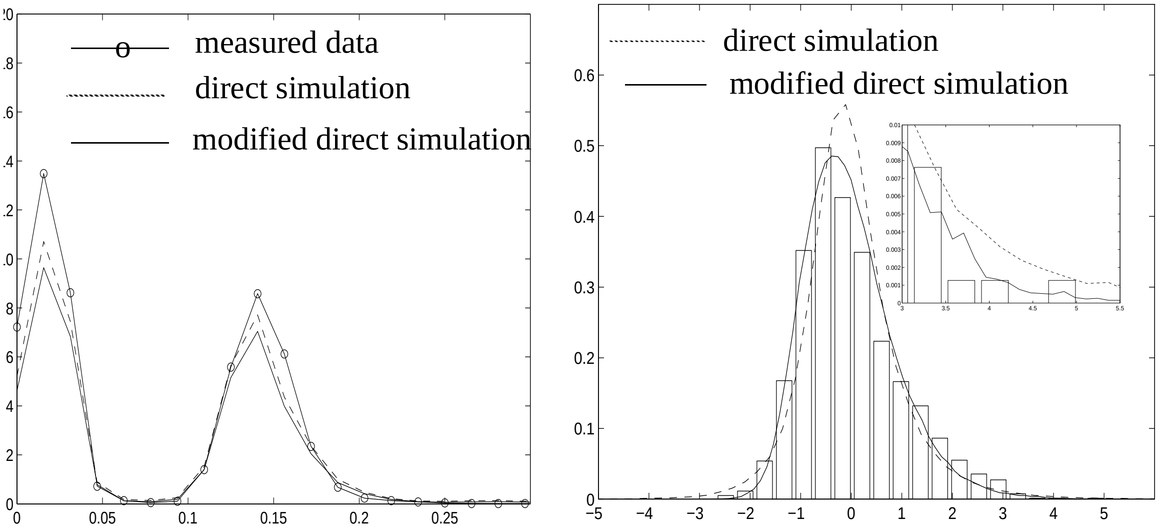


FIGURE 3.6 Power spectral density and pdf of measured TLP response signal and ensemble of 2000 simulations

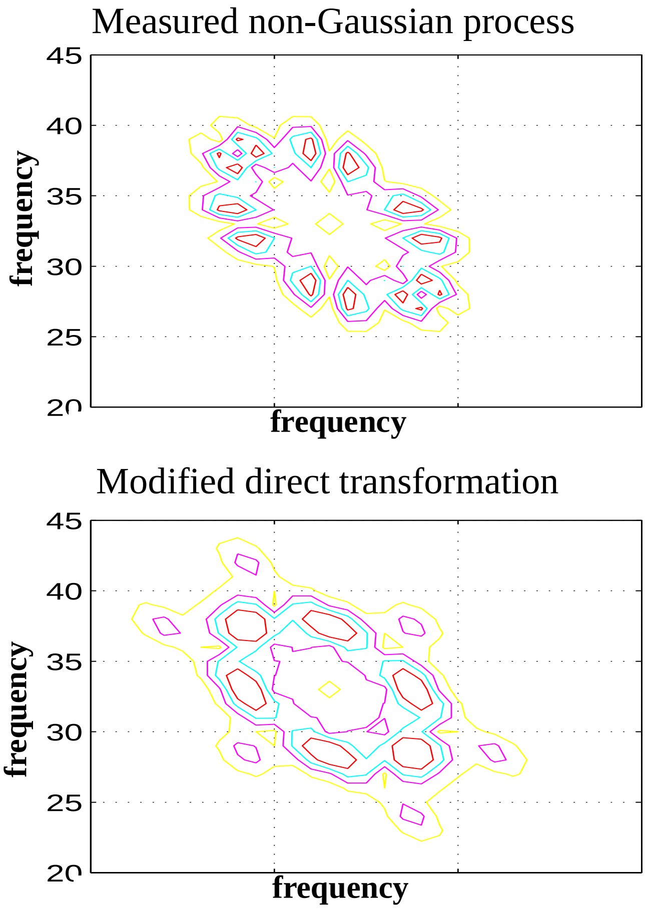
cussed non-Gaussian simulation techniques (see 2.8.1), which requires only the target spectrum and probabilistic characteristics of the desired process as input.

Portions of the Modified Direct Transformation method are used in chapter 5 in the development of a non-Gaussian simulation method which requires only the target spectrum and desired first four moments as input.

3.2.4.2 Matching Higher Order Statistics

The shortcoming of any static transformation is its inability to properly reflect the nonlinear dependence between different times. This memory relationship is necessary to retain the full nonlinear phase interaction among related frequency components, quantified by higher order spectra, e.g. the bispectrum. The bispectrum is a representation of the quadratically coupled frequency components of a realization. Just as the power spectral density is the distribution of the variance of a signal with respect to frequency, the bispectrum is the distribution of skewness with respect to frequency pairs.

Although static transformation methods are able to replicate the volume under the bispectrum, i.e. the skewness, they are not suited to correctly match the distribution of skewness with respect to frequency. Figure 3.7 compares the bispectrum contour of the sample TLP response process with that of the modified direct transformation simulations in the previous example. The simulations are unable to reproduce the shape of the measured bispectrum.


Measured non-Gaussian process

Modified direct transformation

FIGURE 3.7 **Contours of the bispectrum of the measured TLP response and the bispectrum of an ensemble of 2000 simulations using the modified direct, and the direct transformation methods**

3.3 Simulation with Memory Models

When the only available information is a sample of the final process to be simulated (e.g., wind pressure on a building face), this static transformation method is quick and promising. However, if more information is available (e.g., the upwind wind velocity), it is possible to better simulate the desired process by establishing a system identification model between, in this case, velocity and pressure. The limitation of static transformation techniques is overcome by the application of memory-based system identification models.

A memory transformation is described by an operator function $g(y(t))$ that produces a process $x(t)$ which is a function of both current and previous values of another process $y(t)$. No general method for determining the probabilistic characteristics of $x(t)$ as the output of a memory model is yet available. Further, the selection of a form of the memory transformation g to produce a specified distribution in $x(t)$ has only been solved for special cases. This renders memory transformation methods of non-Gaussian simulation impractical as a general tool for simulation non-Gaussian processes of specified but nonrestrictive frequency content and probabilistic characteristics.

Memory models are often used to identify a nonlinear system where measured input and output are available. The estimated model is then used to simulate system output using simulated input. This can be preferable to direct simulation of the output using static transformation methods, as the memory model may capture important characteristics in the output. However, when only a small amount of data is available, the model estimate may not reflect the true system accurately enough to offer this advantage.

The applicability of static or memory transformations may then be differentiated as follows. Static models addressed in this study are intended for the simulation of non-Gaussian input to systems whose mathematical model is available. Transformation of the simulated input through the known model then produces the system output for analysis. Memory transformation is applicable where samples of system input and output are available, but the system mathematical model is not. A memory model of appropriate form is selected, and the system is identified by numerically estimating parameters for the model. System output is then simulated by applying simulated input to the estimated model.

3.3.1 Volterra Series Models

In the Volterra series formulation, the input-output relationship may be expressed in terms of a hierarchy of linear, quadratic and higher-order transfer functions or impulse response functions [e.g., 55, 93, 77]. These transfer functions can be determined from experimental data or from theoretical considerations. For example, a nonlinear system modeled by a Volterra series expansion is described by

$$\begin{aligned} x(t) \ = \ & \int h_1(\tau)y(t-\tau)d\tau + \int\int h_2(\tau_1, \tau_2)y(t-\tau_1)y(t-\tau_2)d\tau_1 d\tau_2 + \\ & \int\int\int h_3(\tau_1, \tau_2, \tau_3)y(t-\tau_1)y(t-\tau_2)y(t-\tau_3)d\tau_1 d\tau_2 d\tau_3 + \ldots \end{aligned} \quad , \qquad (3.7)$$

where $h_1(\tau)$, $h_2(\tau_1, \tau_2)$ and $h_3(\tau_1, \tau_2, \tau_3)$ are the first, second and third-order impulse response functions, and $y(t)$ is some known input to be transformed to $x(t)$. The Fourier transform of the Volterra series expansion in Eq. 3.7 gives the response in the frequency domain as

$$\begin{aligned} X(f_i) \ = \ & H_1(f_i)Y(f_i) + \sum_{f_1 + f_2 = f_i} H_2(f_1, f_2)Y(f_1)Y(f_2) + \\ & \sum_{f + f + f = f} H_3(f_1, f_2, f_3)Y(f_1)Y(f_2)Y(f_3) + \ldots \end{aligned} \quad . \qquad (3.8)$$

The Volterra series model in Eq. 3.8 lends itself to the simulation of nonlinear processes for which the transfer functions $H_1(f_1), H_2(f_1, f_2), H_3(f_1, f_2, f_3)$ are available or may be estimated.

When the input $y(t)$ and output $x(t)$ of a system is available, the information can be used to estimate the Volterra kernels in Eq. 3.8 directly. The first and second order transfer functions are given by

$$H_1(f_i) = \frac{E[X(f_i)Y^*(f_i)]}{E[|Y(f_i)|^2]} \quad . \tag{3.9}$$

and

$$H_2(f_1, f_2) = \frac{1}{2}\frac{E[Y^*(f_1)Y^*(f_2)X(f_1+f_2)]}{E[|Y(f_1)Y(f_2)|^2]} \quad , \tag{3.10}$$

Just as the linear transfer function $H_1(f_i)$ is a function of the cross spectrum $E[X(f_i)Y^*(f_i)]$, the quadratic transfer function (QTF) is a function of the cross bispectrum $E[Y^*(f_1)Y^*(f_2)X(f_1+f_2)]$, which describes the level of quadratic interaction between $y(t)$ and $x(t)$.

The memory retained by convolution with the higher-order transfer functions (HOTF) facilitates the simulation of processes that are able to match not only the power spectrum of the target process, but the higher-order spectra as well. These higher order spectra are the multiple-Fourier transform counterparts to higher order correlations functions

$$R(\tau_1, \tau_2, ..., \tau_n) = E[y(t)y(t+\tau_1)...y(t+\tau_n)] . \tag{3.11}$$

For $n = 2$, Eq. 3.11 is the well known auto-correlation function, whose value at $\tau = 0$ is the second moment of $y(t)$. The Fourier transform of the auto-correlation function provides the power spectral density of $y(t)$, which describes the distribution of the second moment with respect to frequency. For $n = 3$, the third-order auto-correlation provides the third moment for $\tau_1 = \tau_2 = 0$. Its two-fold Fourier transform over all τ_1, τ_2 provides the auto bispectrum, which describes the distribution of the third moment with respect to frequency pairs.

3.3.1.1 Example: Non-Gaussian Sea State

A sample of a nonlinear simulation using a Volterra series model is shown in Fig. 3.8. This realization is the surface elevation of gravity waves, with the non-Gaussian train showing the characteristic high peaks and shallow troughs. In this case the second-order Volterra kernel is analytically derived (e.g., [36, 42, 53, 95]) and referred to as a nonlinear interaction matrix (NIM). The NIM relates a quadratic non-Gaussian process to its underlying Gaussian process. In terms of Eq. 3.8, $Y(f)$ is the desired non-Gaussian wave elevation, $X(f)$ is the underlying linear sea state, and $H_1(f)$ is unity. $X(f)$ is first simulated, then used to generate the second-order contributions.

3.3.1.2 Size Restriction

The computational cost of Volterra simulation increases rapidly with the length of the desired signal due to the required frequency domain convolution. Additionally, the storage of the required QTF can become prohibitive for long realizations. The simulation of a realization of length N requires the storage of a QTF matrix of size on the order of NxN. Using symmetry properties offers some reduction in size, which in certain cases can be reduced to $\frac{N}{2}x\frac{N}{2}$. These factors effectively limit either the total time length of the desired realization, or the frequency resolution.

This limitation can restrict analysis methods which require very long continuous time histories generated using Volterra series simulation. This has recently occurred during the

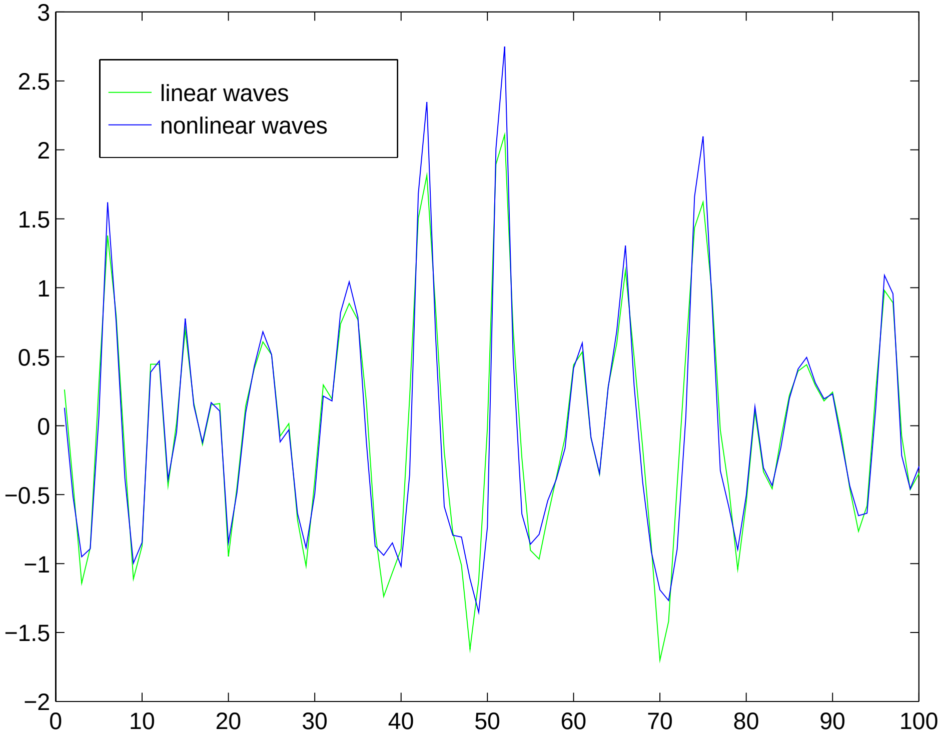


FIGURE 3.8 Realization of a Gaussian and non-Gaussian wave height generated by Volterra series using a nonlinear interaction matrix

development of probabilistic models of a special form of offshore structural response known as ringing [35].

Ringing is characterized by infrequent transient bursts of response due to nonlinear wave-structure interactions, and is known to occur almost exclusively as a result of non-Gaussian sea states. The probabilistic model requires very long duration simulations of the sea states most associated with ringing, which are generated using the Volterra series through second order. The length of the necessary simulations required some approach to overcome the computational and storage restrictions. A segmented Volterra simulation method has been developed, and is presented in the next section.

3.3.1.3 Segmented Volterra Simulation

The simulation of non-Gaussian waves as in Fig. 3.8 is a two step process:

1) A Gaussian wave elevation record, $y(t)$, is generated using an appropriate target PSD and standard Gaussian simulation techniques.

2) $y(t)$ is passed through Eq. 3.8 truncated after the second-order term to produce the non-Gaussian wave profile $x(t)$. The linear transfer function is unity, and the QTF is the nonlinear interaction matrix from section 3.3.1.1.

Equation 3.8 transforms the entire Gaussian time history to non-Gaussian in the frequency domain, and requires a square matrix to represent the QTF of dimensions NxN, where N is the length of the input signal $y(t)$ and output non-Gaussian signal $x(t)$.

If Eq. 3.8 is applied sequentially to overlapping segments of length M of the total Gaussian wave elevation $y(t)$, then the size of the QTF matrix may be fixed to some dimension MxM, where $M = N/(2^i)$, and i is an integer. Each individual segment of the total transformed non-Gaussian wave elevation is of a length determined by the percent overlap of the Gaussian segments.

The generation of $x(t)$ from $y(t)$ is a memory transformation. Thus the beginning of any output segment requires some information about the preceding segment’s input to accurately account for the relationship between $x(t)$ and $y(t-\tau)$. The use of overlapping segments of input provides this information to the output.

An example of a segmented Volterra simulation is shown in Fig. 3.9. The goal is to simulate a non-Gaussian wave elevation record of length 512 seconds. The sampling rate is 1.0 Hz, and the peak frequency of the wave spectrum is 0.1 Hz. The non-Gaussian wave record is simulated using both the full length of a Gaussian wave record, and using segments of the Gaussian wave record sequentially. The procedure and a description of Fig. 3.9 follows:

1) The size of the desired wave record is 512 seconds sample at 1.0 Hz, which makes the size of the QTF for full length simulation $512x512$.

2) For segmented simulation, the size of the segments is selected to be $M = 128$, which sets the size of the QTF for segmented simulation to $128x128$.

3) The segmented simulations will use overlapping segments of the Gaussian simulation. The overlap is chosen to be 25% of the segment length, or 32 seconds. This means the first segment will be 128 seconds long, and the overlap of 32 seconds leads to subsequent segments which are 96 seconds long when the overlapped portion is truncated when assembling the total non-Gaussian record.

4) The number of segments needed to generate the 512 second non-Gaussian segmented simulation is determined from

$$N \le M + ns(1-P)M, \tag{3.12}$$

where $N = 512$, $M = 128$, $P = 0.25$ and represents the selected percent overlap, and ns is the integer number of segments to solve for such that the right and left hand sides of Eq. 3.12 are as close to equal as possible without violating the inequality. ns provides the number of segments necessary, excluding the first segment, to build the full non-Gaussian record of at least length N after truncating the overlapping portion. For this example, $ns = 4$ equates the right and left hand sides of Eq. 3.12 exactly.

5) A Gaussian simulation of the wave elevation, $y(t)$, is generated with total time length equal to the right hand side of Eq. 3.12, in this example it is 512 seconds.

6) A full length non-Gaussian simulation, $x(t)$ is generated using the $512x512$ QTF oper-

ating on the Gaussian record, $y(t)$ from $t = 1...512$. This is the full simulation to be compared with the segmented simulation.

7) The first segment of $y(t)$ from $t = 1...128$ is operated on using a QTF of size $128x128$, where the cutoff frequency is the same as that of the $512x512$ QTF. This produces the first $t = 1...128$ seconds of the segmented realization.

8) The next segment is produced by operating on $y(t)$ from $t = 96...224$, thus overlapping the last 32 seconds of the previous Gaussian segment. The last 96 seconds of the 128 second segment is joined with the endpoint of the previous iteration to produce $x(t)$ from $t = 1...224$.

9) Remaining three segments continue to use the Gaussian segments of length 128 seconds, where the time frame overlaps the last 32 seconds of the previously generated segment. The last three segments lead to $t = 1...320$, $t = 1...416$, and finally $t = 1...512$ seconds of the non-Gaussian wave elevation record $x(t)$.

The resulting non-Gaussian full and segmented simulations are plotted in Fig. 3.9, with the segmented components separated by vertical lines. The top plot is the entire length of 512 seconds. The middle and bottom plots are close up views of two spots in the record where overlapping segments connect. It is difficult to distinguish between the two.

The quality of the segmented Volterra simulation is a function of the amount of overlap used, the frequency resolution of the segmented QTF, and the level of memory in the full second order impulse response function. The frequency resolution in the MxM QTF should be fine enough to adequately represent the full NxN QTF, and adequately represent the linear power spectrum as well.

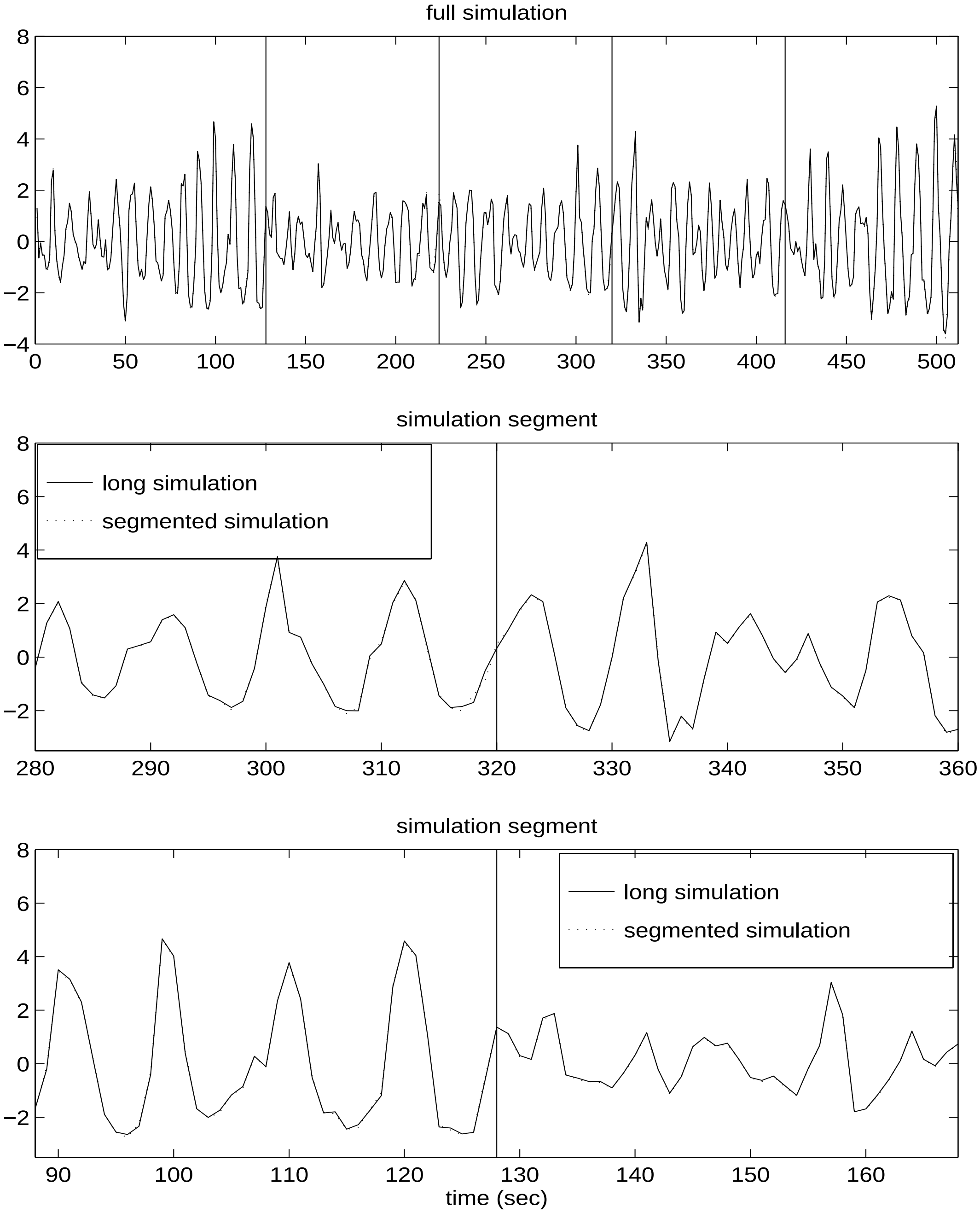


FIGURE 3.9 Simulated non-Gaussian wave train using Volterra on full length of Gaussian waves, and using Volterra on five overlapping segments of Gaussian waves.

3.3.2 Bendat’s Simplified Volterra-type System Identification

Conceptually the Volterra series may easily be extended to the simulation of nonlinear processes beyond second-order, although considerable computation time is added by convolution of the Fourier components with higher-order transfer functions. Also, the acquisition of higher-order transfer functions from measured data becomes difficult and the number of parameters necessary to describe them becomes prohibitive. This has lead to the development of alternative models given in Bendat [5], which reduce the computational difficulties by simplifying the higher order transfer functions to functions of a single frequency.

Bendat has proposed several models which lead to a significant reduction in the above restrictions. The two and three fold convolution integrals in the time domain are replaced with either single fold integrals of squared and cubed input, or squared and cubed single fold integrals of the input. The general third order Volterra model in Eq. 3.7 is replaced respectively by either

$$x(t) = \int h_1(\tau)y(t-\tau)d\tau + \int h_2(\tau)y^2(t-\tau)d\tau + \int h_3(\tau)y^3(t-\tau)d\tau\ , \qquad (3.13)$$

or

$$x(t) = \int h_1(\tau)y(t-\tau)d\tau + \left[\int h_2(\tau)y(t-\tau)d\tau\right]^2 + \left[\int h_3(\tau)y(t-\tau)d\tau\right]^3 . \qquad (3.14)$$

Equations 3.13 and 3.14 are referred to as Case 1 and Case 2 simplifications of the general third order Volterra model. In each case the two and three fold convolution terms of the general Volterra model are replaced with a static transformation in series with a first order impulse response function. Cases 1 and 2 are delineated from each other and the general third order model in the schematic in Fig. 3.10.

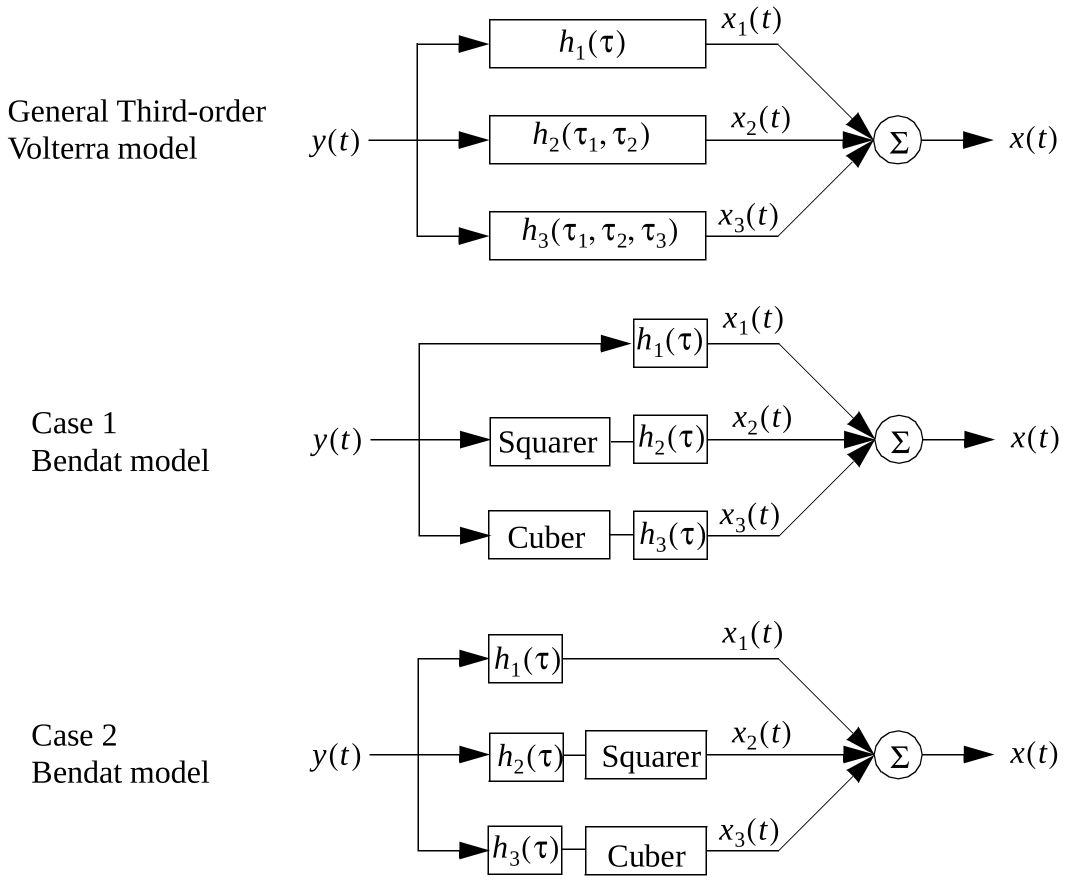


FIGURE 3.10 Schematic of the general third order Volterra model, and the Case 1 and 2 Bendat models

Case 1 can be interpreted as a general Volterra model where the off-diagonal terms in the second and third order impulse response functions are zero-valued. Eq. 3.13 can now be rewritten as a special case of the general Volterra model

$$\begin{aligned} x(t) = \int h_1(\tau)y(t-\tau)d\tau + \iint h_2(\tau_1, \tau_2)\delta(\tau_1, \tau_2)y(t-\tau_1)y(t-\tau_2)d\tau_1 d\tau_2 + \\ \iiint h_3(\tau_1, \tau_2, \tau_3)\delta(\tau_1, \tau_2, \tau_3)y(t-\tau_1)y(t-\tau_2)y(t-\tau_3)d\tau_1 d\tau_2 d\tau_3 + \ldots \end{aligned} \quad . \qquad (3.15)$$

Case 2 is also a special case of the general Volterra model, where in this case the higher-order impulse response functions are the product of single variate functions. Eq.

3.14 is rewritten as

$$
\begin{aligned}
x(t) = \int h_1(\tau)y(t-\tau)d\tau + \iint h_2(\tau_1)h_2(\tau_2)y(t-\tau_1)y(t-\tau_2)d\tau_1 d\tau_2 + \\
\iiint h_3(\tau_1)h_3(\tau_2)h_3(\tau_3)y(t-\tau_1)y(t-\tau_2)y(t-\tau_3)d\tau_1 d\tau_2 d\tau_3 + \ldots
\end{aligned} \quad . \tag{3.16}
$$

The transfer functions corresponding to the impulse response functions $h_2(\tau), h_3(\tau)$ are functions of the cross bispectrum S_{YYX} and cross trispectrum S_{YYYX} between input and output, and the auto bispectrum S_{YYY} and trispectrum S_{YYYY} of the input. Bendat shows (1990) that these higher order cross- and auto-spectra are functions of a single frequency variable for the case 1 and case 2 simplified models. The transfer functions describing the nonlinear terms are then a function of a single frequency variable as well. This greatly reduces the number of parameters necessary to develop the system model, and also requires far less data to accurately estimate these transfer functions. Other models are available which fall in between complete Volterra and Bendat-type models [e.g. 56].

When they are appropriate, these models can greatly enhance the efficiency and applicability of a nonlinear frequency domain analysis and simulation.

3.3.3 Neural Network System I.D. Approach with Input & Output

Another recently developed approach to nonlinear system identification is the application of neural networks. A neural network consists of multi-layered set of processing elements that receive input information and iteratively adjusts a weighting factor between each of the elements by minimizing the error between the resulting and measured target output.

Figure 3.11 shows an example network with three weighting layers $W_{ij}(m), m = 1...3$, where $i = 1...N_m$, $j = 1...N_{m-1}$, and N_m and N_{m-1} are the number of elements in the m^{th}

and the $m^{th}-1$ layers, respectively. The network in Fig. 3.11 has two hidden element layers $a_i(1)$ and $a_i(2)$ between the input and output layers $a_i(0)$ and $a_i(3)$. In this example the input layer consists of the input occurring at the same time as the current output from $a_1(3)$, and the two inputs preceding this lead input (a two delay input system). $W_{ij}(m)$ then represents the weighting of the output from the element $a_j(m-1)$ before its input to element $a_i(m)$. The output of each element is a nonlinear function of the weighted linear sum of the output from each of the elements in the previous layer as in (Kung [61])

$$b_i(m) = \sum_{i=1}^{N_{m-1}} W_{ij}(m)a_j(m-1) + \theta_i(m) , \tag{3.17}$$

$$a_i(m) = f(b_i(m)) , \quad 1 \le i \le N_m , \quad 1 \le m \le 3 , \tag{3.18}$$

where $\theta_i(m)$ is a threshold value fixed for each $a_i(m)$. Various nonlinear functions may be applied at the elements, under the restriction that the output must be limited to $0 \le f(b_i) \le 1$. One commonly applied function is the sigmoid function

$$f(b_i) = \frac{1}{1+e^{-b_i/\sigma}} , \tag{3.19}$$

where σ is a parameter to control the shape of $f(b_i)$.

The element weights in the neural network are adjusted iteratively, commonly with a back propagation scheme, which minimizes the error between the resulting and desired final output. This is known as the training phase, in which the optimum model parameters $W_{ij}(m), m = 1...M$ are identified, where M is the number of network layers, and $M = 3$ for the example in Fig. 3.11 [61].

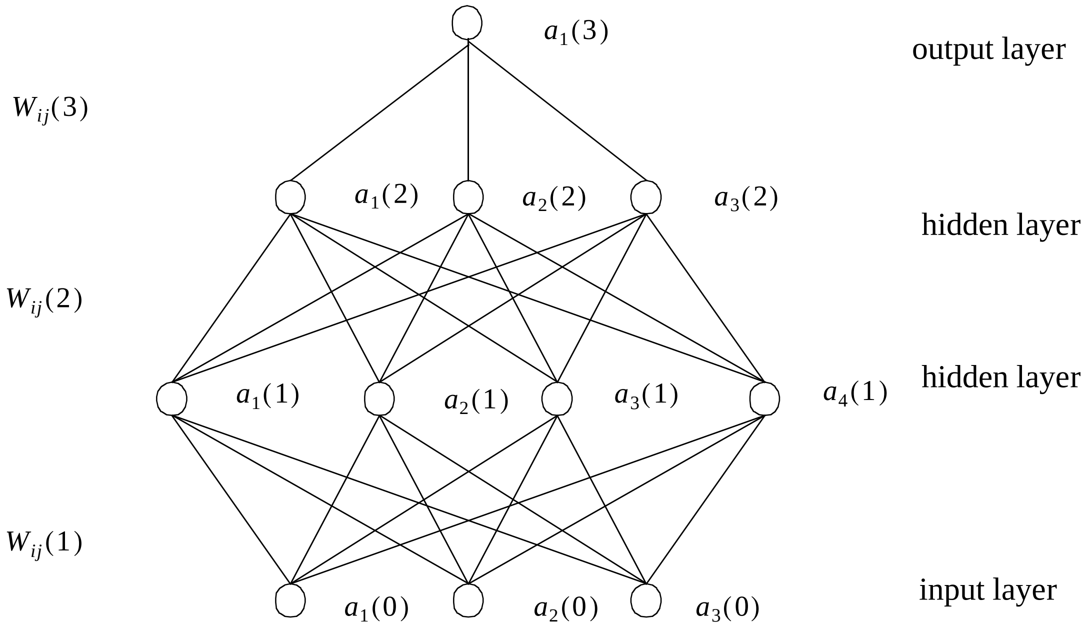


FIGURE 3.11 Multilayer neural network with three weighting layers and two hidden layers (adapted from Kung [61]).

3.3.3.1 Neural Network Simulation Example

A nonlinear transformation of Gaussian wave elevation is used for a neural network simulation example. The system input is a linear wave train simulated based on a JONSWAP spectrum with a peakedness of 5, and a peak frequency of 0.05 Hz. The nonlinear output, $F(n)$, is generated from the linear wave train, $\eta(n)$, by a hypothetical nonlinear function

$$F(n) = 0.1\eta(n-2) + 0.4\eta^2(n-2) + 0.1\eta(n-1) + 0.5\eta^2(n-1) + 0.2\eta(n) + 0.6\eta^2(n)\,. \quad (3.20)$$

A neural network with two delays is trained to model the input / output from 4096 data points. This model is then used to simulate realizations of the output in Eq. 3.20 by passing Gaussian simulations of the input, η, through it. The modified direct transformation is also used to simulate the output directly, without knowledge of the input. A comparison of statistical results is presented in table 3.3, where it can be seen that the modified direct transformation does not match the statistics as well as in previous examples.

TABLE 3.3 STATISTICS OF MEASURED NONLINEAR WAVE PROCESS AND ENSEMBLE AVERAGED SIMULATED DATA.

	Std	Skewness	Kurtosis
Measured Output Data	0.4950	2.2800	9.8329
Ensemble of 10 Modified Direct Transformation Simulations	0.3948	1.8911	8.5284
Ensemble of 10 Neural Network Simulations	0.4692	2.1256	8.6640

Figure 3.12 presents the original sample output from Eq. 3.20 in the top figure, and a neural network and modified direct transformation realization in the next two, respectively. Figure 3.13 shows a contour representation of the bispectrum of the sample output process, and of an ensemble average of 10 realizations using the neural network and modified direct transformation models. At first glance, the modified direct transformation simulation bispectrum contour appears only slightly different from that of the neural network simulation and the sample, which are almost identical. However, significant differences can be seen in an isometric view of the bispectra in Fig. 3.14. This figure also shows the neural network bispectrum to closely resemble that of the original sample, due to the memory retention.

3.3.4 Associated Difficulties with Memory Based Simulation

Each of the memory simulation methods discussed have some restrictions. These are system identification based simulation techniques. Thus if the input process is not known, the method breaks down, as the resulting models are not easily adaptable for simulation

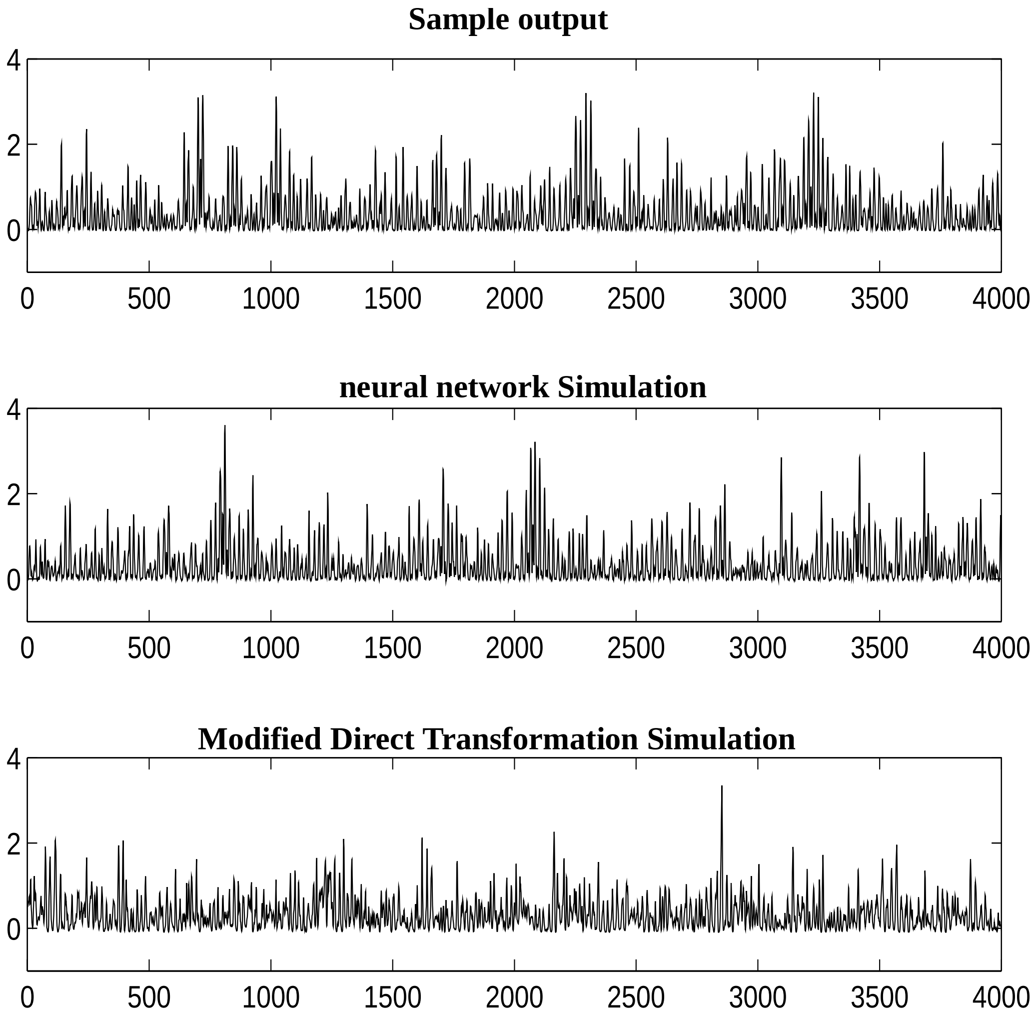
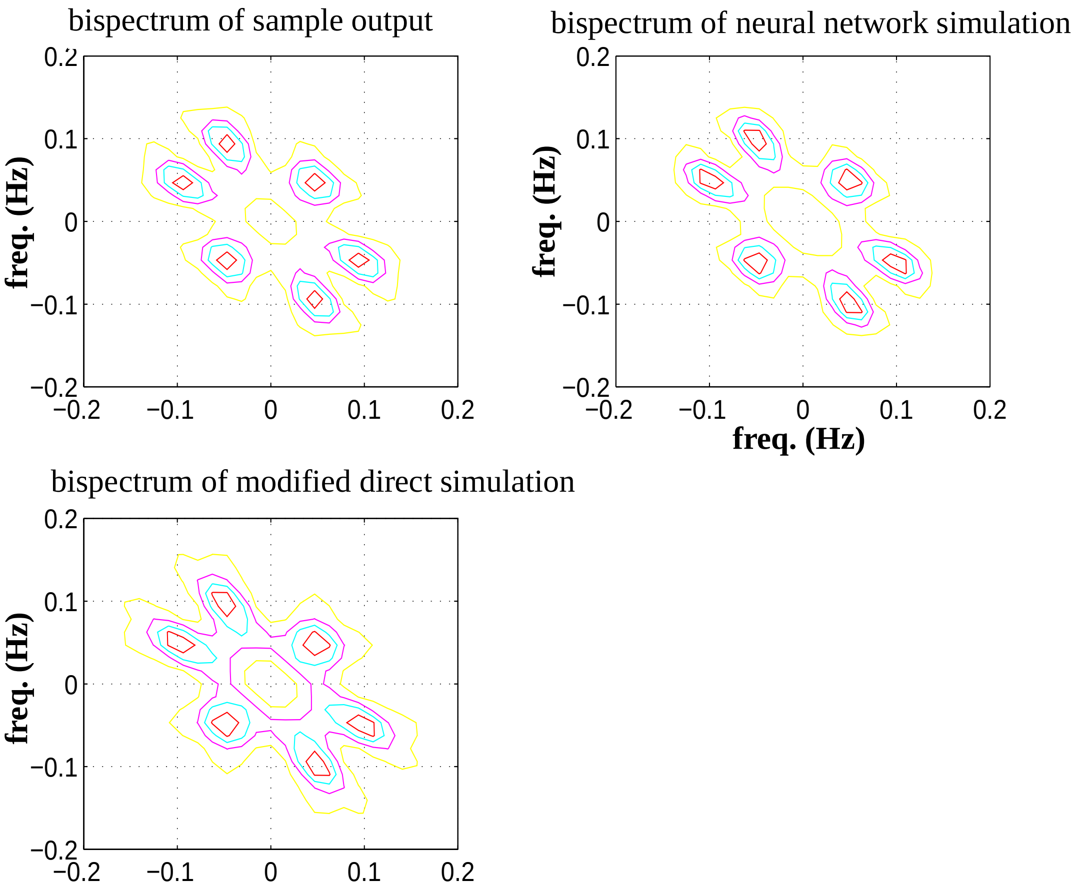


FIGURE 3.12 Sample output from Gaussian sea state input using Eq. 3.20 (top), a simulation using a neural network trained on the sample input / output (middle), and a simulation using modified direct transformation (bottom).

purposes. If the measured input is non-Gaussian, it must be of a class that is well described by static polynomial transformations of a Gaussian process. If this is not the case, the memory model identified is not strictly applicable to a simulated non-Gaussian input based on static transformation.

General Volterra type memory models require either abundant data to accurately estimate the higher order transfer functions (HOTF), or require HOTFs known apriori. The amount of data required for system identification becomes restrictive, particularly for


bispectrum of sample output

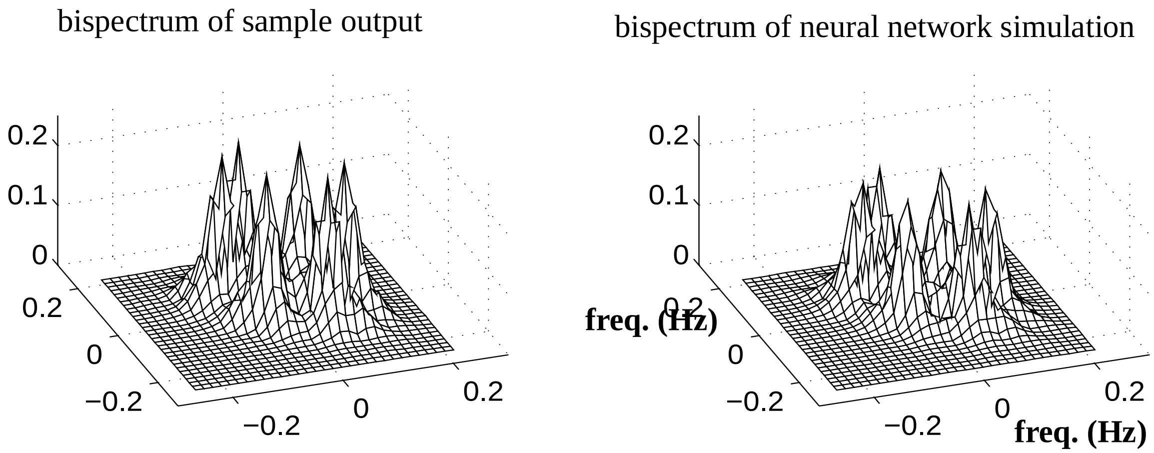
bispectrum of neural network simulation

bispectrum of modified direct simulation

FIGURE 3.13 Bispectrum contour of Eq. 3.20 output (top left), bispectrum contour of 10 neural network realizations (top right), and bispectrum contour of 10 modified direct transformation realizations (bottom left).

higher than quadratic order systems, and analytical HOTFs are only available for a few special cases. Thus, general Volterra models are not readily adaptable as a universal memory based method of simulating non-Gaussian processes.

Bendat presents a class of finite memory nonlinear models which reduce the general Volterra HOTFs dependence to a single frequency variable. These models greatly reduce the computational effort, but they are not appropriate when the static nonlinear transformation in series with a linear transfer function is insufficient to describe the relationship between input and output.



bispectrum of sample output

bispectrum of neural network simulation

bispectrum of modified direct simulation

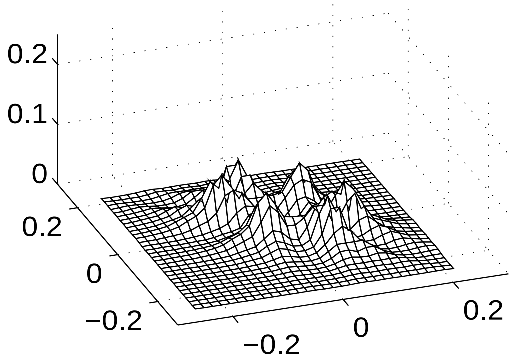


FIGURE 3.14 Isometric view of Fig. 3.13. Bispectrum of Eq. 3.20 output (top left), bispectrum of 10 neural network realizations (top right), and bispectrum of 10 modified direct transformation realizations (bottom right).

3.4 Concluding Remarks

Several non-Gaussian simulation techniques are presented in this chapter. The applications may divided into two cases. The first concerns the simulation of a non-Gaussian process as system input, where the output may be generated through a known system model. This case uses static transformation of a Gaussian process to generate the non-Gaussian input.

The second concerns the simulation of system output, where system identification models are discussed. The application of memory models is presented in the context of system identification, where the unknown system equation is estimated based on available samples of the system input and output, or analytical system models. A specialized application of Volterra-based simulation is developed for the generation of very long realizations, and applied in the development of probabilistic models of offshore transient structural response modelling. Neural networks are applied as a system i.d and simulation tool. An example application demonstrates the potential to retain better higher order spectral information when memory models are used for simulation.

The methods in this chapter are not applicable for general non-Gaussian simulation, where only a PDF and non-Gaussian characteristics are available to describe the desired process. Portions of the Modified Direct Transformation method are used later in the development of a non-Gaussian simulation method which requires only the target spectrum and desired first four moments as input. This forthcoming method will be shown to perform well for cases where current methods in the literature fall short, and will be extended to both multivariate and conditional non-Gaussian simulation.

CHAPTER 4

MODELLING OF PDFs FOR NON-GAUSSIAN PROCESSES

4.1 Introduction

Approximate nonlinear system response methods often result in statistical descriptions of response in the form of higher order moments. Since exact response distributions of nonlinear systems are only available in a few special cases, approximate methods using these estimated moments are employed. Series distribution methods, including Gram-Charlier, Edgeworth, and Longuet-Higgins series methods, all based on Hermite polynomials, have been commonly used (e.g., [71]). These series methods all tend to exhibit oscillating and negative tail behavior. For extreme response analysis, alternative means must be considered. Models based on maximum entropy and a Hermite transformation approach are considered here.

The principle of maximum entropy is employed under three different types of constraint conditions based on moment information from either the system equations or estimated from measured data. The advantages of the various formulations are examined, with some emphasis on a new form of constraint condition which shows improvement over the more traditional forms. The Hermite moment model is also employed in this study to model the probability density function, and is shown to provide accurate estimates of extreme system response. Two sets of constraints are added to the standard closed form Hermite moment model to improve performance and enhance applicability.

The maximum entropy- and Hermite moment-based PDF models, and their modified versions discussed herein, are employed in later chapters to accurately describe the probabilistic content of non-Gaussian processes for simulation purposes. Portions of the following work was published in the Journal of Wind Engineering and Industrial Aerodynamics [28].

4.2 Maximum Entropy Method (MEM)

4.2.1 Traditional Formulation

An approach to approximate the pdf of nonlinear systems is the Maximum Entropy Method (MEM), in which the entropy functional is maximized subject to constraints in the form of moment information. In the limiting case of infinite moment information, a unique PDF is defined. In reality, there is always a finite amount of moment information, for which there are an infinite number of admissible probability density functions. According to the maximum entropy principle, the PDF which maximizes the entropy functional is the least biased estimate for the given moment information. The application of entropy concepts to estimate an unknown probability distribution was first suggested by Jaynes [44]. Sobczyk and Trebicki [91], provide a clear framework for the development of the maximum entropy formulation for stochastic systems. The presentation found in [91] is given below

Consider a system of stochastic equations

$$\frac{d\boldsymbol{x}}{dt} = \boldsymbol{F}[\boldsymbol{x}(t), \boldsymbol{y}(t)] , \tag{4.1}$$

where $\boldsymbol{x}(t)$ is an unknown response process and $\boldsymbol{y}(t)$ is a given stochastic excitation. If the excitation process is represented as a filtered white noise, the excitation terms can be appended to the response process such that the new $\boldsymbol{x}(t)$ contains as many variates as the total input / output system order.

The available information for $\boldsymbol{x}(t)$ is in terms of either moments or moment equations, expressed here generically as the expected value of some polynomial function of $\boldsymbol{x}(t)$ denoted $G_{r_1 r_2 \dots r_n}(\boldsymbol{x})$. The expected value is then

$$E[G_{r_1 r_2 \dots r_n}(\boldsymbol{x})] = \int G_{r_1 r_2 \dots r_n}(\boldsymbol{x}) p(\boldsymbol{x}) d\boldsymbol{x} = \beta_{r_1 \dots r_n} , \tag{4.2}$$

where n is the number of degrees of freedom in the system, and $r_i = 0, 1, 2, ..., M$ where M is the maximum order or correlation moment.

One possible pdf estimate of the process $\boldsymbol{x}(t)$ is that which maximizes the entropy functional

$$H = -\int p(\boldsymbol{x}) \ln p(\boldsymbol{x}) d\boldsymbol{x} , \tag{4.3}$$

subject to constraints in the form of system information.

After application of the Lagrange multiplier method, the resulting description of $p(\boldsymbol{x})$ is

$$p(\boldsymbol{x}) = exp(-\lambda_0 - 1)exp\left(-\sum_{r_1 + \ldots + r_n}^{M} \lambda_{r_1 \ldots r_n} G_{r_1 \ldots r_n}(\boldsymbol{x})\right) . \tag{4.4}$$

Substitution of Eq. 4.4 into the constraints in Eq. 4.2 and an additional normalization constraint $\int p(\boldsymbol{x}) d\boldsymbol{x} = 1$ gives the following system of equations [91]

$$\int G_{r_1 \ldots r_n}(\boldsymbol{x}) exp\left(-\sum_{r_1 + \ldots + r_n}^{M} \lambda_{r_1 \ldots r_n} G_{r_1 \ldots r_n}(\boldsymbol{x})\right) d\boldsymbol{x} = \beta_{r_1 \ldots r_n} , \tag{4.5}$$

$$\int exp\left(-\sum_{r_1 + \ldots + r_n}^{M} \lambda_{r_1 \ldots r_n} G_{r_1 \ldots r_n}(\boldsymbol{x})\right) d\boldsymbol{x} = exp(\lambda_0 + 1) . \tag{4.6}$$

This system of nonlinear integral equations is solved numerically and the results yield the leased biased estimate of the system joint PDF under the given moment constraints using Eq. 4.4, 4.5, 4.6.

An alternative formulation reduces the set of nonlinear integral equations to a single concave function whose minimum provides the optimal Lagrange parameters [1]. This method is employed in this work, and aids in the speed of solution by lessening restrictions on the initial values of the Lagrange parameters.

4.2.2 MEM Applied with System Equations

When the system equations are available, the system information used to constrain Eq. 4.3 is typically a set of higher order moments. For this case the general polynomial function becomes

$$G_{r_1 r_2 \ldots r_n}(\boldsymbol{x}) = x_1^{r_1} x_2^{r_2} \ldots x_n^{r_n} , \tag{4.7}$$

and includes all combinations of

$$x_1^{r_i} x_2^{r_j} \dots x_n^{r_k} \tag{4.8}$$

such that

$$r_i + r_j + \dots + r_k = 1, 2, \dots, M , \tag{4.9}$$

where again M is the desired maximum order or correlation moment.

The values $\beta_{r_1 \dots r_n}$ from Eq. 4.2, representing the set of joint moments, are necessary for solution of the Lagrange parameters. Tagliani has derived the limiting values of higher moments for which a maximum entropy solution will exist [94].

Application of Ito's formula to the system equation provides the system's moment equations, the solution of which yields the desired joint moments. For constant coefficient linear systems, the moment equations are a closed set for any desired order, and the moment values are easily found. For nonlinear systems, the moment equations are an infinite hierarchy, and some closure technique needs to be applied to approximate the joint moments to desired order [92].

Use of the full set of joint moment values leads to an estimate of the joint PDF through order M. This requires a large number of parameters to satisfy Eqs. 4.4-4.9, and the repeated numerical solution of an n fold integral. Further, if the system is nonlinear, the resulting PDF model will be based only on an approximated set of joint moments, and the intense numerical effort may lead to a poor estimate.

An alternative to applying closure to nonlinear system moment equations is to apply the moment equations through order M directly as the constraints in Eq. 4.2. In this case the constraints $G_{r_1 r_2 \dots r_n}(\boldsymbol{x})$ are the moment equations, and $\beta_{r_1 \dots r_n} = 0$. This avoids the need for closure methods, while providing accurate system information for nonlinear systems [90].

The number of parameters in the MEM formulation increase rapidly as the number of degrees of freedom and the desired order increase. Each parameter has an associated non-linear integral equation, and the solution requires multiple evaluation of each of these integral equations. The analytical evaluation of the equations is in general not available, making time consuming numerical evaluation of a large number of multi-fold integrals necessary.

If the marginal PDFs of one or more degrees of freedom are all that is needed, it is highly desirable to solve for the marginal distribution directly. This may be done by solving the moment equations for the marginal moments, or numerically estimating the marginal moments. There is considerable computational savings involved in solving Eqs. 4.5 and 4.6 as a one degree of freedom integral with 4 unknowns, when the alternative is solving the much larger full set of constraints with multi-fold integrals.

Marginal PDFs cannot be evaluated directly using the moment equations as constraints. Rather, the joint PDF is estimated, and the marginal PDFs are estimated through integration of the joint PDF. The numerical difficulties associated with solving a system with a large number of parameters prohibits the application of MEM to special cases where either constitutive relationships between moment equations can be analytically identified and used to reduce the number of parameters (e.g., [91]), or the system order is small enough to allow direct solution of the full system.

Based on the authors’ experience with MEM joint PDF estimates of general higher order systems, the correlation order M should be no less than four to produce acceptable estimates. Given this restriction, systems of order n higher than 2 cannot be practically solved using MEM without system dependent reductions in the number of parameters.

4.2.3 Examples of MEM Using System Equations

Sobczyk and Trebicki [90, 91] have shown numerous examples of the application of both moment and moment equation constraints. Further examples can be found elsewhere [e.g. 9]. Here we repeat an example of a first order system using moment constraints, and extend the example to include moment equation constraints.

The first system under consideration is given by [90]

$$\frac{dx(t)}{dt} + ax(t) - \frac{b}{x(t)} = D\xi(t), \tag{4.10}$$

where $\xi(t)$ is a white noise, and $b = 0.5, a = 1, D = 1$. The moment equations are found through application of the Ito formula to Eq. 4.10, giving constraints

$$G_i(x) = \frac{1}{2}(i-1)D^2x^{(i-2)} + bx^{(i-2)} - ax^i, \; i = 1, ..., M. \tag{4.11}$$

Figure 4.1 shows the exact PDF provided [90], and MEM estimates using 4 moments and 4 moment equations as constraints. While the approximations fit well in the mean region, the inset figure shows the equation constraint model more accurately represents the distribution at its limiting value $y = 0$.

Quantitative measures of the goodness of fit the PDF estimates will be presented in section 4.2.6 when this example is revisited using an alternative form of MEP constraints.

4.2.4 MEM Applied with Measured Data

For the later work on non-Gaussian simulation methods, we seek PDF models developed from measured data to be used for simulation purposes. When the system information is in the form of measured or simulated data, either the joint or marginal PDFs may be estimated using estimated moments. If the measured response of a system described by

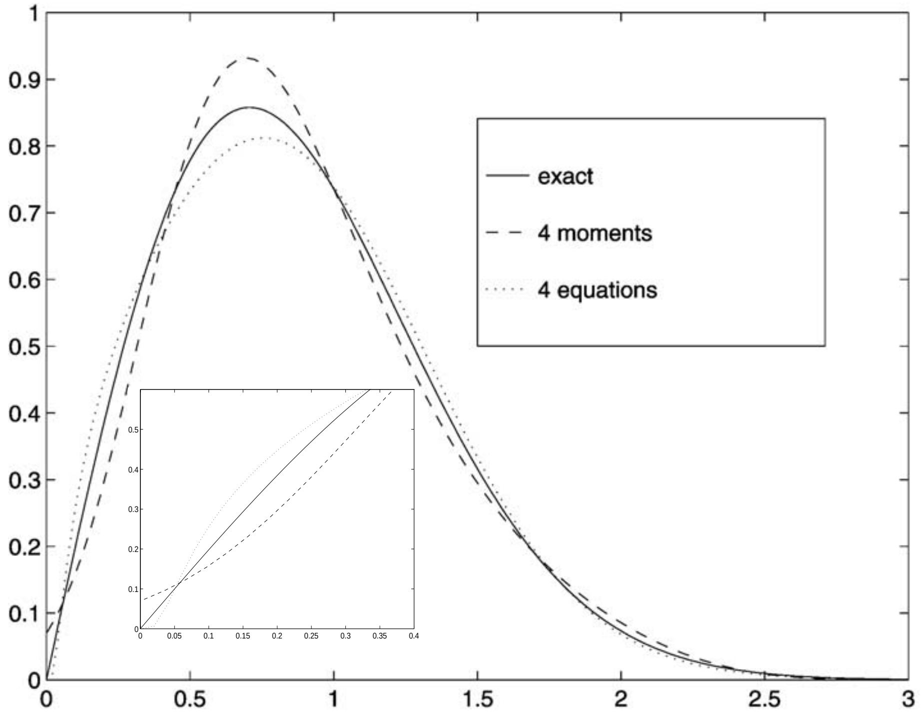


FIGURE 4.1 Exact PDF of Eq. 4.10, and MEM estimates using 4 moments and 4 equations.

Eq. 4.10 were available and sufficiently large, the MEM estimate would approach that in Fig. 4.1 where the exact moments are known. The inclusion of uncertainty in estimating the moments of limited data is addressed in Ciulli, et al. [10].

The motivation for using an MEM estimate over a generic curve fit of a PDF histogram of the data lies in the physical principles behind the MEM approximation. While MEM is a curve fitting model in a sense, it is constrained only by statistical characteristics of the system, not by fitting points in a data set. It can be argued that the MEM offers a best estimate short of making further assumptions concerning the nature of the data, as a generic curve fit would require. This is addressed in a thorough treatment of MEM in texts by Siddall [88], and Sobczyk [89], and it is stated as a motivating factor for the selection of MEM as an appropriate PDF model for reliability based applications in the literature[e.g. 72].

4.2.5 Restriction of MEM in the Extreme Regions

MEM has been used to describe the PDFs of multi-degree of freedom systems using either moments or moment equations as constraints. For these examples the fit of the MEM solution to the exact PDF is good in the mean region of the distributions when the moment order M is sufficiently high. However, the form of the MEM pdf model restricts its applicability in the tail region. The MEM method expresses the pdf as an exponential of a polynomial of at least fourth order. Regardless of the value of the Lagrange parameter associated with the higher order terms, they will tend to dominate in the extreme regions [99]. This results in tails that roll off from the exact distribution.

An example is shown in Fig. 4.2, which is a close up of the tail of Fig. 4.1 on both linear- and semi-log scales. It can be seen that both estimates clearly diverge from the exact solution. Further, the equation constraint estimate is not monotonic in the extreme region. This is a severe restriction, and limits the usefulness of MEM to analyses where the tail region is not of interest. An alternative formulation of MEM constraints will be proposed which overcomes this restrictive tail behavior.

4.2.6 Alternative MEM Constraints

As discussed in section 4.2.4, for the case where the system information is in the form of measured or simulated data, MEM provides an approximate PDF given estimates of the moments.

The basis of the MEM approximation is to constrain Eq. 4.3 through system information of the form in Eq. 4.2. Traditionally, the first four moments are chosen as constraints, since they are readily available either from the moment equations or from estimates using

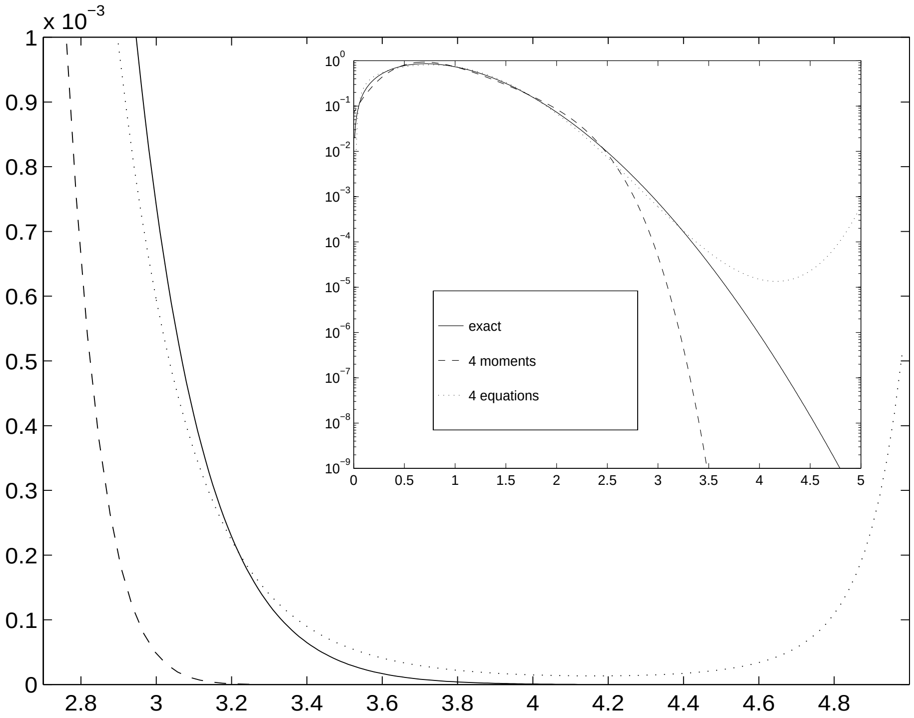


FIGURE 4.2 Linear- and semi-log scale views of the tail region in Fig. 4.1.

measured data. Higher order moment constraints are realized in Eq. 4.4 as cubic, quartic, and possibly higher functions of the process variate x, and lead to problems in the tail region as x gets large. However, the MEM is not restricted to using system information of this form. Any information which adequately represents the characteristics of the system may be applied to constrain the entropy functional.

The tail behavior may be alleviated by replacing the higher order moment constraints

$$E[x],\ E[x^2],\ E[x^3],\ E[x^4]\ , \tag{4.12}$$

with, for example, the constraints [28]

$$E[x],\ E[x^2],\ E[x^{-3}],\ E[x^{-4}]\ . \tag{4.13}$$

In this form, the MEM PDF estimate will not be dominated by higher order polynomials as x gets large. The selected constraints in Eq. 4.13 may be altered to various combinations of $E[x^n], E[x^{-m}]$ to fit the particular data being modeled.

The previous example is used again in Fig. 4.3, where a third MEM estimate has now been added using the constraints in Eq. 4.13, denoted as MEP II in the figure. The top graph is a complete view of the exact PDF with the MEM estimates, and the lower left and right graphs are log-scale views of the left and right tails, respectively. The MEM II estimate provides a better fit than the previous two MEM estimates in the mean region. More importantly, the exact PDF is almost indistinguishable from the MEM II estimate in the tail region, where the other MEM estimates diverge rapidly.

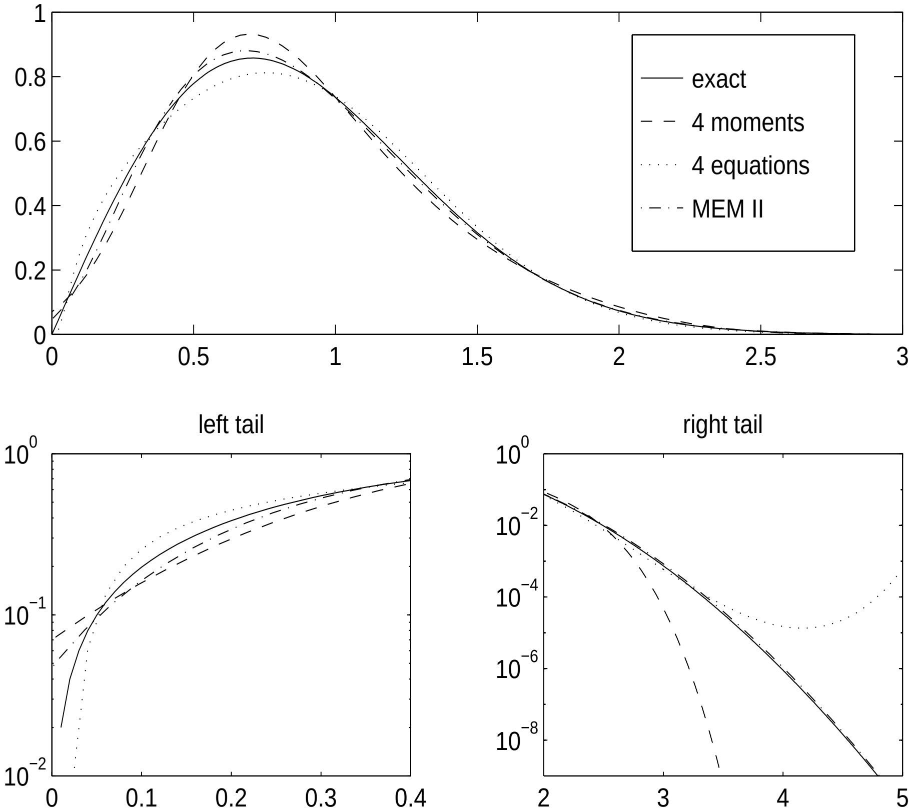


FIGURE 4.3 Exact PDF of Eq. 4.10, and 3 MEM estimates using 4 moments, 4 equations, and MEM II (Eq. 4.13).

Two measures of the goodness of fit are now applied to quantitatively differentiate between the PDF estimates. The first test is the mean square error (MSE) between the exact and fitted distributions, and the second is a K-S goodness of fit test, which is essentially the maximum error between the model and exact cumulative distribution functions (e.g., [4]). Both tests will later be used to compare the MEM and Hermite PDF estimates.

Table 4.1 shows the MSE and K-S goodness of fit test results, where lower values indicate a better fit. The quantitative results indicate that the new constraint formulation provides the best fit in this example.

TABLE 4.1 MEM FIT TESTS FOR FIG. 4.3

	MSE	K-S
4 moments	7.48e-4	0.0196
4 equations	4.12e-4	0.0120
MEM II (eq. 4.13)	1.31e-4	0.0074

A second example from Sobczyk and Trebicki [90] considers the system

$$\frac{dx(t)}{dt} + a_1 x(t) + a_2 x^2(t) + a_3 x^3(t) ln\frac{|x(t)|}{\beta} = D\xi(t) , \qquad (4.14)$$

with the parameter values $a_1 = 0.5, a_2 = -0.25$ $a_3 = 5.2, \beta = 1, D = 1$. Sobczyk and Trebicki [90] present the exact PDF and compare it with MEM estimates using both moment and moment equation constraints. In Fig. 4.4 this work is repeated, and a third estimate is included where the constraints are of the form in Eq. 4.13. In this case the standard constraints provide better MEM estimates than the new formulation, particularly in the tail region. Table 4.2 provides the goodness of fit results to confirm these observations.

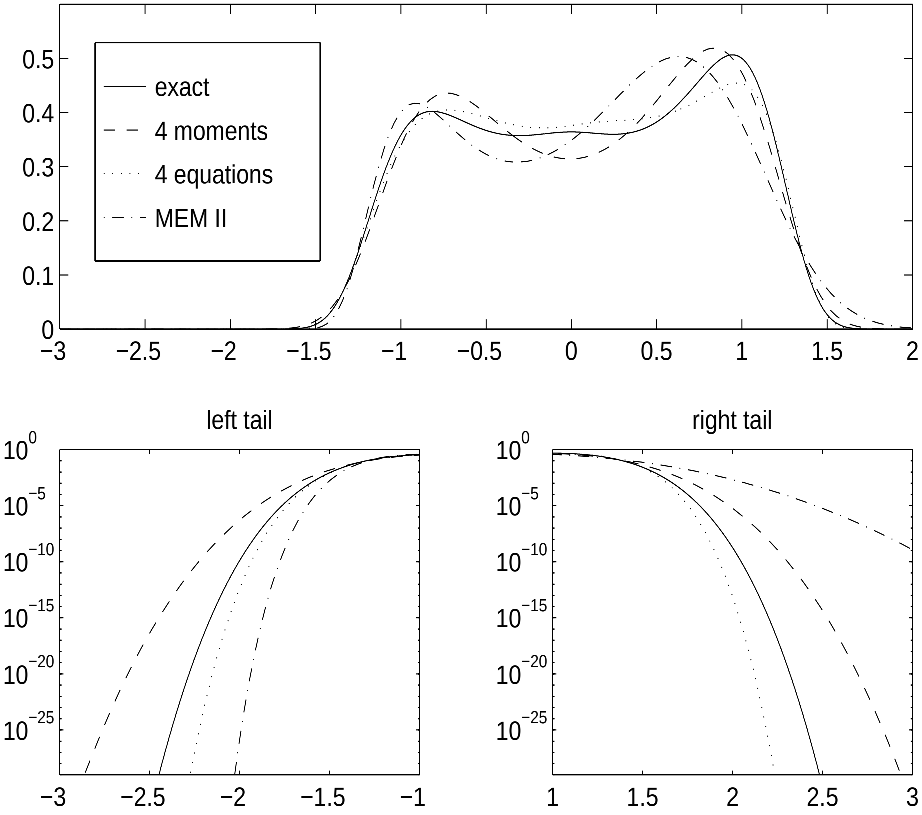


FIGURE 4.4 Exact PDF of Eq. 4.14, and 3 MEM estimates using 4 moments, 4 equations, and MEM II (Eq 4.13).

TABLE 4.2 MEM FIT TESTS FOR FIG. 4.4

	MSE	K-S
4 moments	5.14e-4	0.0113
4 equations	2.24e-4	0.0173
MEM II (eq. 4.13)	1.90e-3	0.0304

4.3 Hermite Moment Method

4.3.1 Formulation of Hermite Moment PDF Estimate

This approach is based on a functional static transformation of a standard Gaussian process, $u(t)$, to a non-Gaussian process, $x(t)$, (e.g., [21])

$$x(t) = g[u(t)]\,. \tag{4.15}$$

Several choices of g are possible to preserve only the first four moments. A cubic model of g offers a convenient and fairly accurate representation [98].

The PDF of $x(t)$ is given by [21]

$$p(x) = \frac{1}{\sqrt{2\pi}} exp\left[-\frac{u^2(x)}{2}\right]\frac{du(x)}{dx}, \tag{4.16}$$

where y is expressed in terms of u for softening processes (kurtosis > Gaussian) as [98]

$$x = u + \hat{h}_3(u^2 - 1) + \hat{h}_4(u^3 - 3u), \tag{4.17}$$

and its inverse is

$$\begin{aligned} u(x) = {} & [\sqrt{\xi^2(x) + c} + \xi(x)]^{1/3} - \\ & [\sqrt{\xi^2(x) + c} - \xi(x)]^{1/3} - a \end{aligned}, \tag{4.18}$$

where

$$\xi(x) = 1.5b\left(a + \frac{x}{\alpha}\right) - a^3, \tag{4.19}$$

$$a = \frac{\hat{h}_3}{3\hat{h}_4},\ b = \frac{1}{3\hat{h}_4},\ c = (b - 1 - a^2)^3, \tag{4.20}$$

$$\hat{h}_3 = \frac{\gamma_3}{4 + 2\sqrt{1 + 1.5\gamma_4}},\ \hat{h}_4 = \frac{\sqrt{1 + 1.5\gamma_4} - 1}{18}, \tag{4.21}$$

$$\alpha = \frac{1}{\sqrt{1 + 2\hat{h}_3^2 + 6\hat{h}_4^2}}, \tag{4.22}$$

and γ_3 and γ_4 are the skewness and kurtosis of the process x. For hardening processes (kurtosis < Gaussian), u is expressed in terms of x in similar form to Eq.4.17.

4.3.2 Limitations in the Standard Hermite Model

The various parameters in Eqs. 4.17-4.22, all dependent on γ_3, γ_4, have been derived based on the assumption of small deviations from Gaussian. When the Hermite transformation is applied in cases where the skewness and kurtosis vary significantly from Gaussian, the transformation fails to accurately capture the non-Gaussian characteristics. Examples will follow the proposed improvements.

4.3.3 Modified Hermite Models

The PDF estimate provided by Eq. 4.16 can be viewed as a two parameter model whose shape is determined by γ_3, γ_4. These define the skewness and kurtosis of the PDF estimate. The mean and variance can be accounted for through translations and dilation of the PDF model.

An improvement to this model is provided by using the expressions for $\hat{h}_3$ and $\hat{h}_4$ given previously (which are approximations) as initial conditions for solving the following pair of nonlinear algebraic equations [96]:

$$\bar{\gamma}_3 = \alpha^3(8\hat{h}_3^3 + 108\hat{h}_3\hat{h}_4^2 + 36\hat{h}_3\hat{h}_4 + 6\hat{h}_3), \tag{4.23}$$

$$\begin{aligned}\bar{\gamma}_4 + 3 = \alpha^4(&60\hat{h}_3^4 + 3348\hat{h}_4^4 + 2232\hat{h}_3^2\hat{h}_4^2 + 60\hat{h}_3^2 + \\ &252\hat{h}_4^2 + 1296\hat{h}_4^3 + 576\hat{h}_3^2\hat{h}_4 + 24\hat{h}_4 + 3)\end{aligned}, \tag{4.24}$$

where $\bar{\gamma}_3, \bar{\gamma}_4$ are the target skewness and kurtosis. Solution of these equations leads to values for $\hat{h}_3, \hat{h}_4$, which lead to the necessary values for γ_3, γ_4. These equations have been derived by setting the third-and fourth-order central moments of $g[u(t)]$ equal to the known central moments of the non-Gaussian process $x(t)$.

A more robust method of identifying the optimal parameters γ_3, γ_4 is to numerically minimize the error between the target skewness and kurtosis $\bar{\gamma}_3, \bar{\gamma}_4$ and the skewness and kurtosis measured from the estimated PDF in Eq. 4.16. The optimization minimizes the constraints on the higher order statistics as

$$err = (\bar{\gamma}_3 - \gamma_{3e})^2 + (\bar{\gamma}_4 - \gamma_{4e})^2 , \tag{4.25}$$

γ_{3e}, γ_{4e} are provided by

$$\gamma_{3e} = \frac{\int_{lb}^{ub} (y - \mu_x)^3 p(x) dx}{\sigma_x^3} , \tag{4.26}$$

$$\gamma_{4e} = \frac{\int_{lb}^{ub} (x - \mu_x)^4 p(x) dx}{\sigma_x^4} , \tag{4.27}$$

where lb, ub are the lower and upper bounds of integration, and

$$\mu_y = \int_{lb}^{ub} x p(x) dx , \quad \sigma_x = \sqrt{\int_{lb}^{ub} (x - \mu_x)^2 p(x) dx} . \tag{4.28}$$

The minimization of the error in Eq. 4.25 provides the optimal values γ_3, γ_4 to produce an estimate $p(x)$ that maintains the desired skewness and kurtosis $\bar{\gamma}_3, \bar{\gamma}_4$.

In the case where system equations are available in place of measured data, the optimization function in Eq. 4.25 may again be constrained in terms of higher order moment information. However, when the system is nonlinear, the moment equations may be used directly as constraints in order to avoid the application of closure. The expected value of any moment equation for a stationary system is zero, and the objective function becomes

$$minimize\left(\sum_{i=1}^{4} |G_i(x)| \right) . \tag{4.29}$$

The relationship between the standard normal process $u(t)$ and the non-Gaussian process $x(t)$ in Eq. 4.16 can be described by any static transformation $g(u)$ which is capable of producing a non-Gaussian process with the desired non-Gaussian skewness and kurtosis. The use of the Hermite polynomials here provides PDF estimates whose skewness γ_{3e} is independent of γ_4, and kurtosis γ_{4e} independent of γ_3 through the orthogonal property of the Hermite polynomials. This increases the efficiency of the optimization algorithm.

4.3.4 Example Application of Modified Hermite PDF Estimation Method

The previous example system in Eq. 4.10 is used to demonstrate the Hermite based PDF estimates. Figure 4.5 presents the exact PDF, the standard Hermite moment PDF model (Eqs. 4.16-4.22), denoted ‘Herm.’, the modified Hermite model using moment constraints, denoted ‘Herm. mom’, and the modified Hermite model using moment equation constraints, denoted ‘Herm. Eq.’. Both Fig. 4.5 and the fit statistics in Table 4.3 indicate no obvious superiority of one estimate over another in this case.

The second example considers the exact PDF corresponding to Eq. 4.14 and the three Hermite estimates denoted as in the previous example. Figure 4.6 shows all three models provide conservative and acceptable estimates in the tail region. The standard Hermite model is not an accurate estimate in the mean region, while both modified Hermite models (equation constraints and moment constraints) provide a more acceptable fit in the mean. The fit statistics in Table 4.4 confirm the improvements offered by both modified Hermite estimates.

Comparing these results with the previous example, it can be seen that the MSE and K-S fit statistics are capable of distinguishing between estimates that are similar, and estimates that significantly differ.

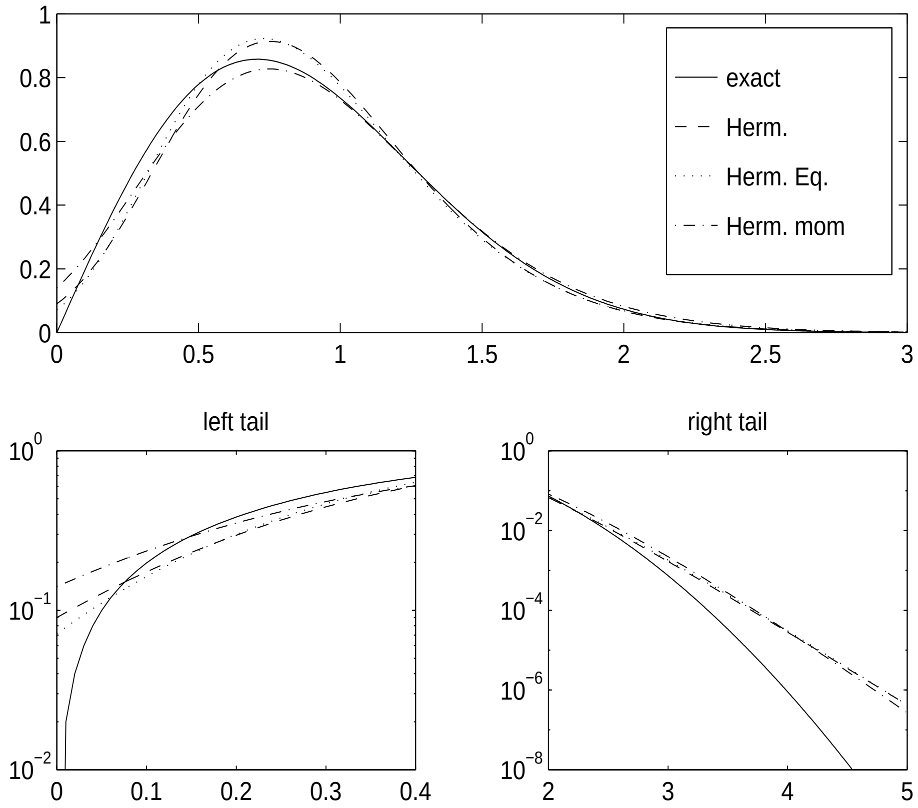


FIGURE 4.5 Exact PDF of Eq. 4.10, and 3 Hermite estimates using standard model, 4 moment equation constraints, and 4 moment constraints.

TABLE 4.3 MEM FIT TESTS FOR FIG. 4.5

	MSE	K-S
Herm.	9.02e-4	0.0278
Herm. Eq	7.25e-4	0.0219
Herm. mom	6.52e-4	0.0261

4.4 PDF Estimates of Measured Data

A final pair of examples estimate the PDF of measured data, where no system equations are available. These examples illustrate the intended purpose for the PDF models developed here, to represent the probabilistic characteristics of measured data use in simulation methods.

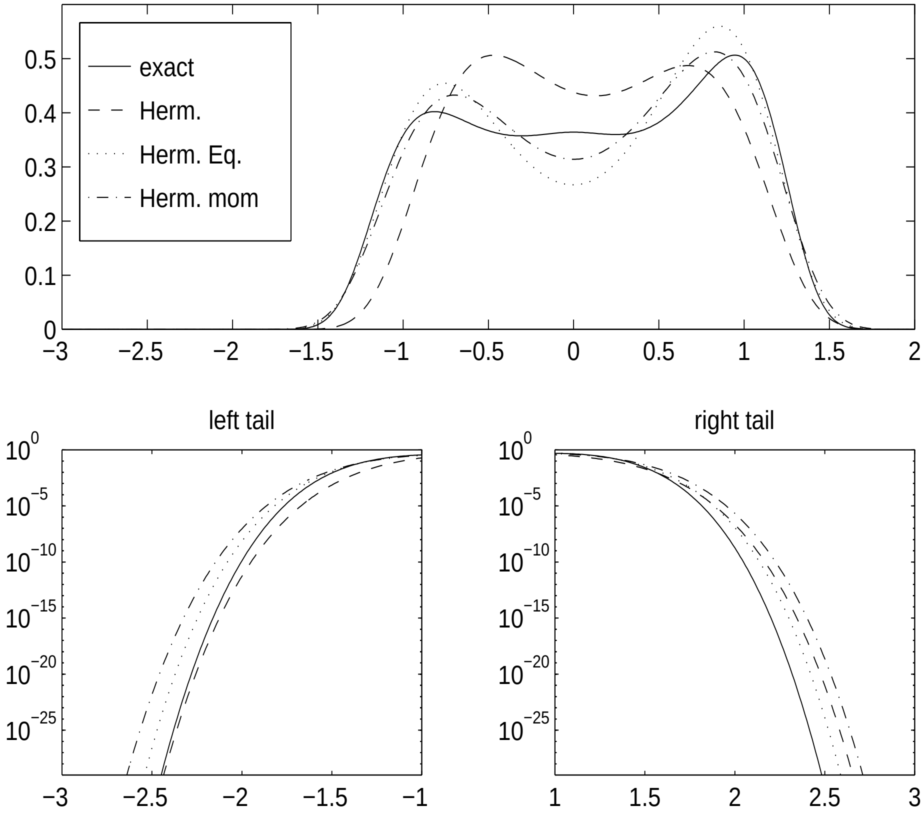


FIGURE 4.6 Exact PDF of Eq. 4.14, and 3 Hermite estimates using standard model, 4 moment equation constraints, and 4 moment constraints.

TABLE 4.4 HERMITE FITS FOR FIG. 4.6

	MSE	K-S
Herm.	5.20e-3	0.0731
Herm. Eq	1.20e-3	0.0304
Herm. mom	5.45e-4	0.0132

The first example uses a full scale non-Gaussian pressure record measured on the top of a building in a separated flow region. An accurate PDF model would be necessary, for example, to provide a statistical description of input to a reliability model of cladding components. In Fig. 4.7 the data histogram is compared with 4 PDF estimates. Two estimates are the standard 'Herm.' and modified Hermite model using moment constraints 'Herm. mom'. Two MEM estimates are also included. One uses the standard 4 moments constraints, denoted ‘MEM I’, and the other uses Eq. 4.13 for constraints, denoted ‘MEM II’. Some of the important features of Fig. 4.7 are the oscillating tail behavior of the MEM I estimate in the extreme region, and the good fit provided in this region by both Hermite models and MEP II model. In the right tail, only MEP II follows the data trend well, and in the mean region the MEM II seems to provide the best estimate. The fit statistics in Table 4.5 indicate that the standard Hermite and MEM models are less appropriate than the modified versions proposed in this study.

The second in this pair of examples concerning experimental data uses the measured non-Gaussian response of a model compliant offshore platform subjected to random wind and waves. The input sea state is non-Gaussian, and the structural system is nonlinear. Figure 4.8 presents the data histogram and the four PDF estimates. The most obvious feature is the divergent tail behavior of the MEP I estimate, and its poor fit in the mean region. Table 4.6confirms that MEP I is by far the least appropriate fit. The table suggests also that MEP II is the best fit, although all three estimates excluding MEP I fit the mean region and the important extreme right tail region very well.

4.5 Concluding Remarks

This study reviews two existing PDF models based on either system equations or measured data, and presents modifications to these models which can improve performance. Alternative constraints for the maximum entropy method are presented which overcome the oscillating tail behavior associated with the traditional MEM formulation. An optimization routine is employed to improve the standard Hermite moment PDF estimation model using either moment value or moment equation constraints. Systems with known

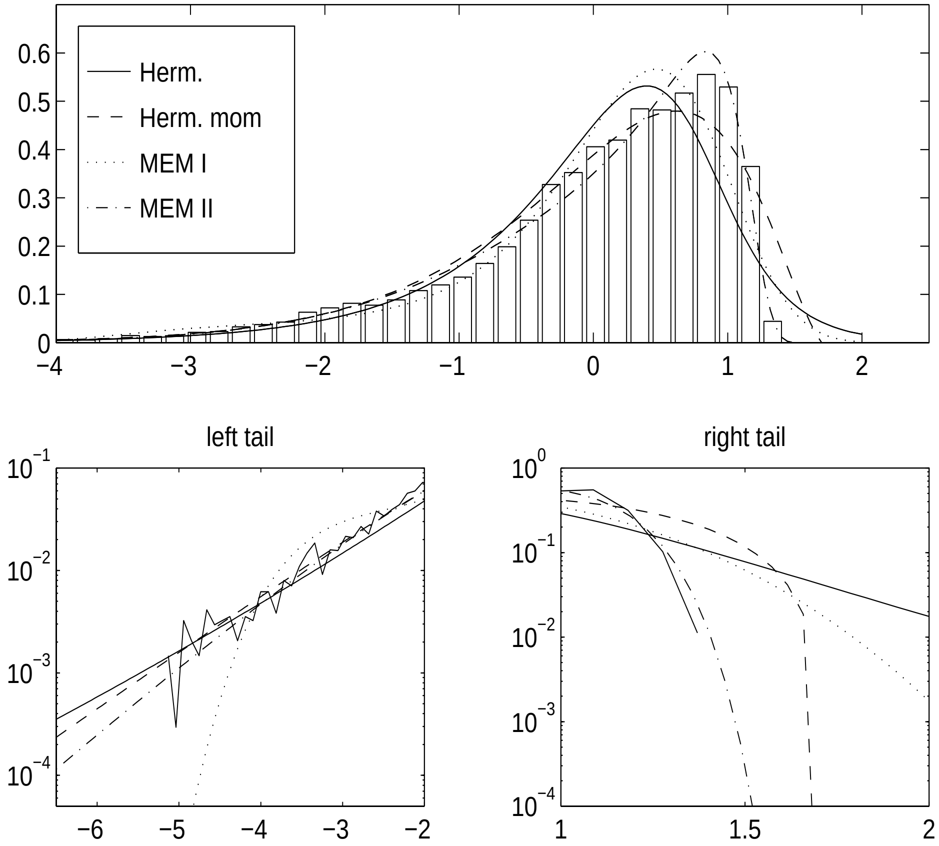


FIGURE 4.7 Pressure data PDF histogram, and four PDF estimates using two Hermite models and two MEM models.

TABLE 4.5 FIT TEST RESULTS FOR FIG. 4.7

	MSE	K-S
Herm.	3.96e-3	0.0542
Herm. mom	1.67e-3	0.0336
MEM I	2.68e-3	0.0410
MEM II	6.44e-4	0.0159

exact PDFs are used to demonstrate the potential improved performance of the proposed modifications. The models are also applied to measured data, and the performance of the various estimates are compared.

In the following chapters, the new non-Gaussian simulation methods require accurate information regarding the non-Gaussian character of the processes to be simulated. It is

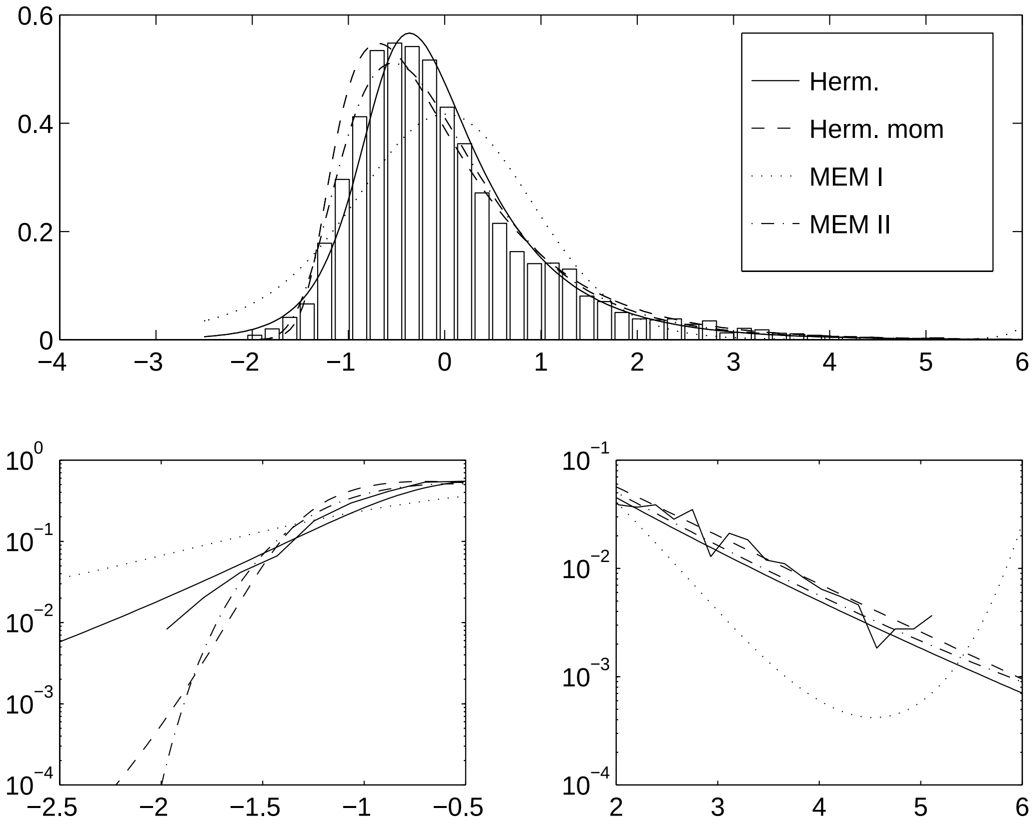


FIGURE 4.8 Offshore platform response data PDF histogram, and four PDF estimates using two Hermite models and two MEM models.

TABLE 4.6 FIT TEST RESULTS FOR FIG. 4.8

	MSE	K-S
Herm.	7.72e-4	0.0429
Herm. mom	5.36e-3	0.0563
MEM I	5.9e-3	0.1156
MEM II	5.2e-4	0.0197

not suggested that the modified models are universally more applicable than their standard counterparts. Rather, these modifications extend their versatility to situations where they may otherwise not be appropriate, and offer more alternatives for simulation purposes.

CHAPTER 5

SIMULATION OF UNI-VARIATE NON-GAUSSIAN PROCESSES: THE SPECTRAL CORRECTION METHOD (SC)

5.1 Introduction

The PDF models developed in chapter 4 were shown to effectively represent non-Gaussian characteristics measured from data. The Modified Hermite and MEM probability models are now used to represent non-Gaussian characteristics in a simulation algorithm.

An earlier study by the author [32] presented the development of the so called Modified Direct transformation method for the simulation of non-Gaussian processes, utilizing the backward Modified Hermite Transformation (see section 3.2.3). This work focussed on the simulation of processes for which a sample time history was available. It is not capable of simulation based on only a target spectrum and target PDF, and is of limited application. A far more robust and versatile method of non-Gaussian simulation makes use of the backward and forward Modified Hermite Transformations, and is capable of matching a target power spectrum.

This FFT-based iterative simulation technique generates non-Gaussian processes that match a target power spectrum and either four target moments or an analytical PDF model. This technique is referred to as the spectral correction method (SC), and will be the basis for new methods of multi-variate and conditional simulation in later chapters. Portions of the work in this chapter have recently been published [28].

The method is described schematically in a step by step procedure. Examples are then provided to demonstrate the method's effectiveness. The stationary property is discussed, and the convergence of this iterative method is shown. The various input options, and options in the SC algorithm are then addressed.

5.2 Previous Work, the Modified Hermite Transformation

It was shown in chapter 4 that the Modified Hermite PDF estimate provides an adequate representation of experimental data using the first four moments. The Modified Hermite moment-based PDF model is applied to a Gaussian normal time history such that the moments of the transformed process match a set of target values through an iterative procedure which optimizes the transformation parameters. The static transformation applied to the normal Gaussian process u is of the form introduced in chapter 4, Eq. 4.17. An optimization procedure minimizes the error between the measured third and fourth moments of x, and the target moments. This procedure is referred to as the forward Modified Hermite Transformation when applied to produce a non-Gaussian realization x.

The inverse procedure is also available, where the optimization applies the inverse of Eq. 4.17, to transform a non-Gaussian realization, x, to a process which reflects Gaussian skewness and kurtosis values. In this form, the optimization procedure is referred to as backward Modified Hermite Transformation.

5.3 Schematic: Spectral Correction Simulation

Spectral correction is a robust new simulation method which utilizes user specified non-Gaussian characteristics and frequency content in the form of target moments through fourth order and a target power spectrum. Alternatively, the user may define non-Gaussian characteristics through a PDF model.

There are two options for the exit criterion in the iterative SC algorithm. The first option is discussed in the following section, and the second thereafter.

5.3.1 Spectral Correction: Exit Option 1

Spectral correction is shown in a schematic in Fig. 5.1. A six step description of the algorithm in Fig. 5.1 follows.

1) Generate a Gaussian normal process using a random amplitude random phase (RARP) algorithm (see section 2.2.1)

$$u(t) = \sum_{n=1}^{n} (A_n \cos\omega_n t + B_n \sin\omega_n t)\,, \tag{5.1}$$

or

$$u(t) = \sum_{n=1}^{n} a_n \cos(\omega_n t + \theta_n)\,, \tag{5.2}$$

where $a_n = \sqrt{A_n^2 + B_n^2}$, and $\theta_n = \operatorname{atan}(B_n/A_n)$. A_i and B_i are independent Gaussian random variables with values at equally spaced discrete frequencies ω_i separated by $\Delta\omega$ with

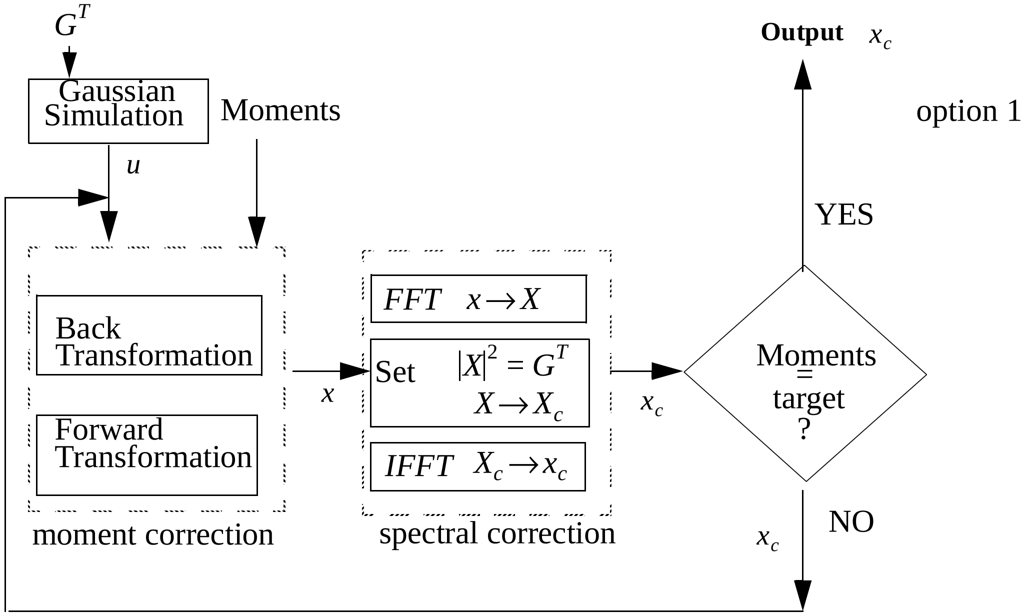


FIGURE 5.1 Schematic of the spectral correction method, exit option one

covariance

$$E[A_n^2] = E[B_n^2] = G(\omega_n)\Delta\omega\,, \quad E[A_nB_n] = 0\,. \tag{5.3}$$

u and the first four target moments, $m_k^T, k = 1...4$ are sent into the moment correction section of the algorithm.

2) Apply Forward Modified Hermite Transform to $u(t)$ to produce a non-Gaussian process $x(t)$

$$x = g_{FH}(u)\,, \tag{5.4}$$

where g_{FH} is the forward modified Hermite transformation (section 3.2.3). $x(t)$ can be expressed as the sum of cosine functions

$$x(t) = \sum_{n=1}^{N} b_n \cos(\omega_n t + \psi_n)\,, \tag{5.5}$$

where b_n, ψ_n are functions of A_n, B_n and the transformation g_{FH}. The sum frequencies resulting from the nonlinear transformation which are greater than ω_N are ignored due to the pre-assigned cutoff frequency.

The moments of x match the target moments m_k^T, and has a power spectrum G^x which no longer matches the target spectrum G^T due to distortion caused by the Modified Hermite Transformation. The non-Gaussian process, x, is then directed to the spectral correction section.

3) The spectral correction section transforms the process, x, to a process, x_c, whose spectrum again matches the target spectrum G^T. The amplitude correction is applied to the Fourier coefficients of x, represented by $|X|^2 = G^T$, while maintaining the phase ψ_n. In the time domain, $x(t)$ is transformed to

$$x_c(t) = \sum_{n=1}^{N} \sqrt{2G_n^T} \cos(\omega_n t + \psi_n) . \tag{5.6}$$

4) The spectral correction transformation from x to x_c distorts the target moments in x. A check is performed on the error between the target, γ_3^T, γ_4^T, and the resulting simulation skewness and kurtosis, γ_3, γ_4, of $x_c(t)$.

$$err = \left|\frac{\gamma_3^T - \gamma_3}{\gamma_3^T}\right| + \left|\frac{\gamma_4^T - \gamma_4}{\gamma_4^T}\right| . \tag{5.7}$$

If the error is within the specified tolerance, the algorithm stops and $x_c(t)$ is the output non-Gaussian process with target spectrum G^T. If the error is not within the specified tolerance, another iteration is begun in step 5).

5) The next iteration is begun by a backward Modified Hermite Transform of $x_c(t)$ to a process with Gaussian skewness and kurtosis $x_g(t)$

$$x_g = g_{BH}(x_c)\,, \tag{5.8}$$

where g_{BH} is the backward modified Hermite transform (section 3.2.3)

6) The algorithm now returns to step 2), where u is replaced for the second and any future iterations with x_g as

$$x = g_{FH}(x_g)\,. \tag{5.9}$$

In each iteration a check of the error is performed at step 4), and the iterative cycle is stopped if the tolerance is reached. The error measured after step 3) by Eq. 5.7 becomes smaller in each subsequent iteration, for reasons to be addressed later.

5.3.2 Spectral Correction: Exit Option 2

The algorithm in Fig. 5.1 will exit the loop after the spectral correction section if the error between the target moments and the distorted moments in x_c is small.

The second exit option is described in the schematic in Fig. 5.2. In this case, the same iterative corrections to the moments and spectrum are applied. However, rather than exiting based on the error between target and resulting moments after the spectral correction section, this algorithm exits based on an acceptable error between the target and resulting spectrum after the moment correction section. Using this approach, the moments will match the target, and the spectrum will be within a specified tolerance.

Both exit options have been applied and readily converge to solution within several

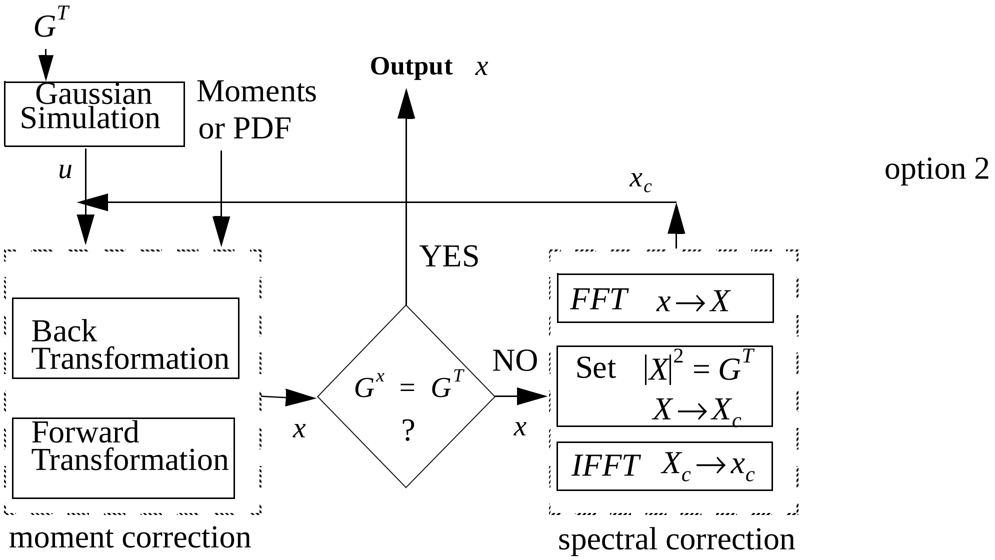


FIGURE 5.2 Schematic of the spectral correction method, exit option two.

iterations. Details of the method will be examined after the presentation of some examples.

5.3.3 Simulation Examples

5.3.3.1 Simulation of a Sea State

Recall the example given in section 3.1.1 demonstrating the Yamazaki and Shinozuka method by simulating non-Gaussian waves. The results of this example are presented again, and compared to the results of a spectral correction simulation of the same process. The target spectrum is a narrow banded JONSWAP sea state spectrum with a peak factor = 6.0, and a peak frequency = 0.1 Hz.

Figure 5.3 shows the target PSD with PSDs measured from SC and Y&S simulations. The Y&S method is unable to reproduced the narrow banded spectrum in the low frequency region, resulting in energy from 0 to 0.05 Hz that is not specified in the target spectrum. The SC simulation is able to emulate the target spectrum throughout the entire frequency range.

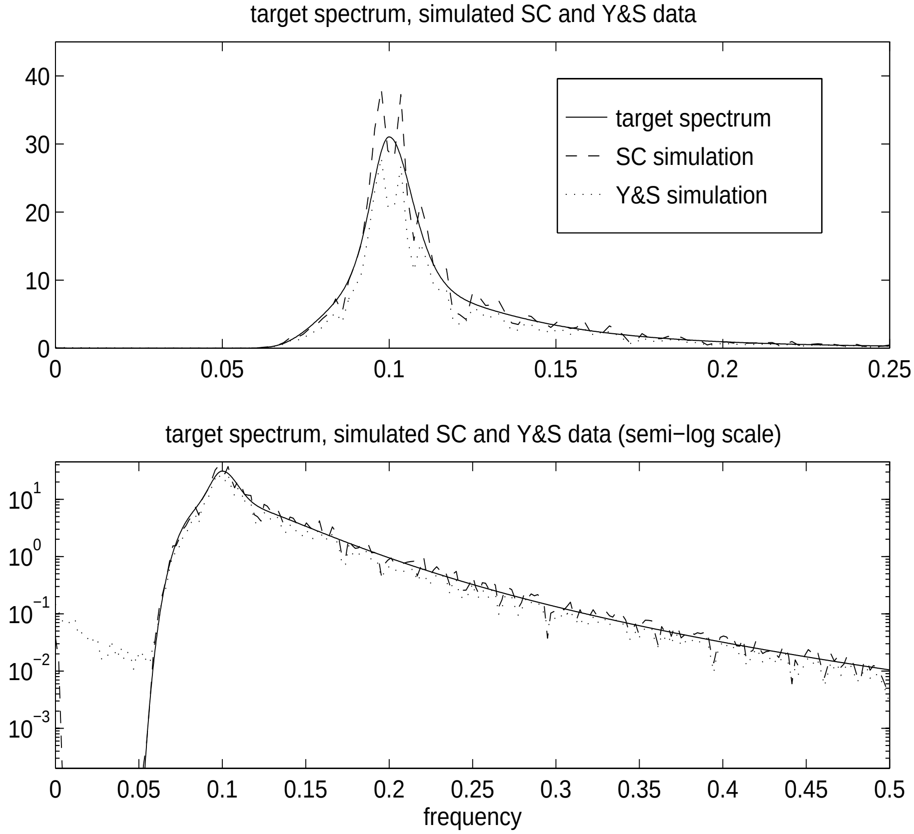


FIGURE 5.3 Wave elevation data: Comparison of target PSD and simulation PSDs using SC and Y&S.

Figure 5.4 shows the Modified Hermite PDF model used to specify the non-Gaussian characteristics of the simulations, and the estimated PDFs of the two simulations. SC uses a Hermite transformation directly on the time history, and Y&S uses a CDF mapping scheme. The bottom two figures indicate acceptable tail representation for both cases.

The target and simulation skewness and kurtosis are listed in Table 5.1 below. It is noted that the SC simulation has skewness and kurtosis closer to the target values. Figure 5.5 shows two sample realizations of wave elevation using SC and Y&S non-Gaussian simulation methods. The added low frequency content in the Y&S simulation is manifested as grouped clusters of large magnitude waves.

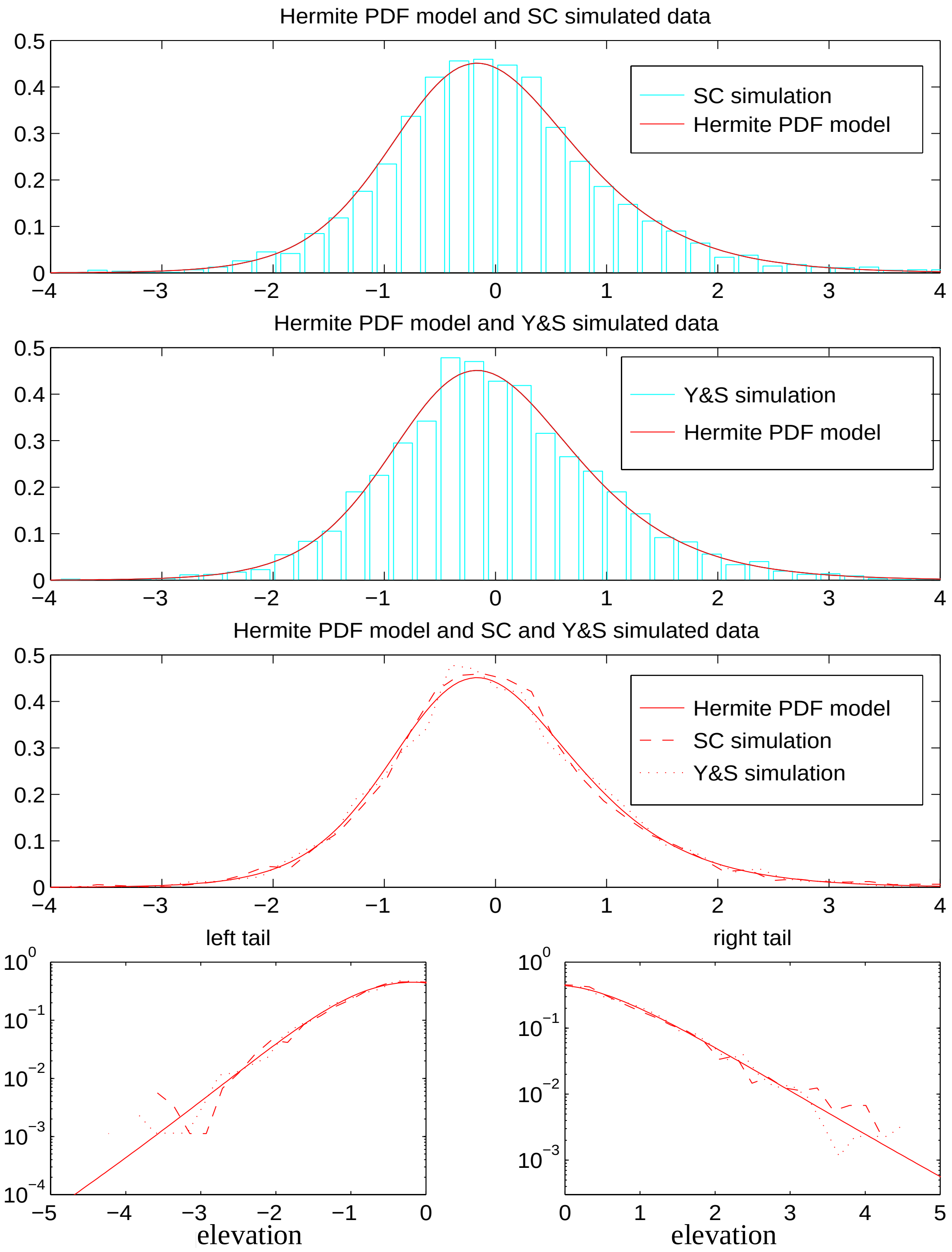
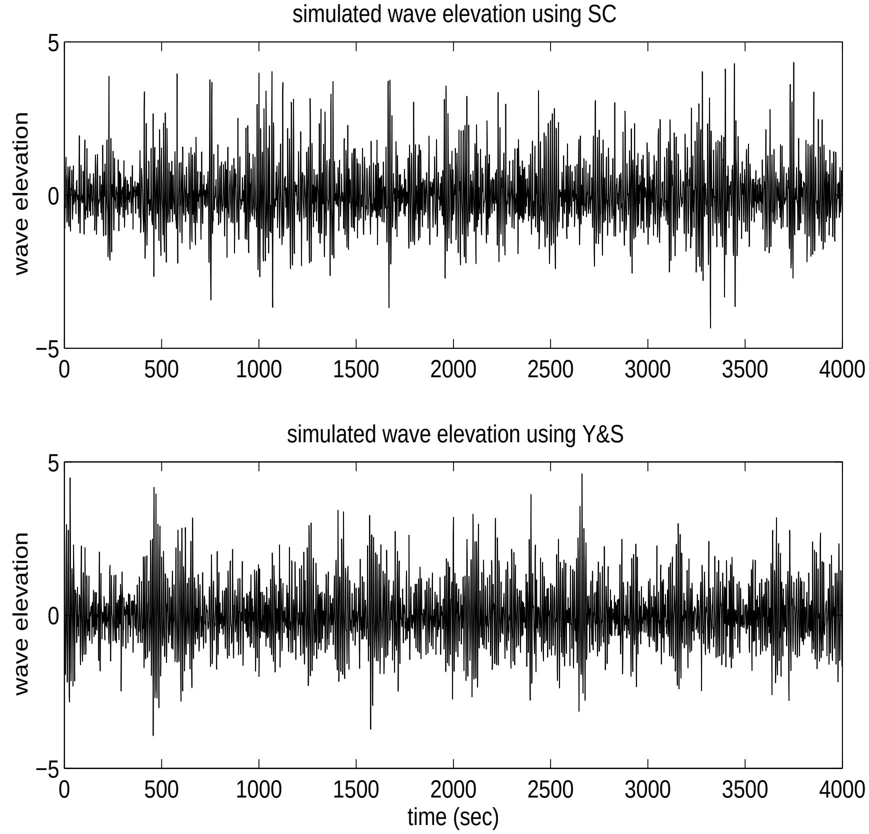


FIGURE 5.4 Wave elevation data: Comparison of target PDF with simulation PDFs using SC and Y&S non-Gaussian simulation

TABLE 5.1 COMPARISON OF TARGET SKEWNESS AND KURTOSIS WITH THOSE MEASURED FROM THE SC AND Y&S SIMULATIONS OF WAVE ELEVATION

	skewness	**kurtosis**
target	0.5	4.5
SC simulation	0.49	4.45
Y&S simulation	0.39	4.03


simulated wave elevation using SC

simulated wave elevation using Y&S

FIGURE 5.5 Simulated wave elevation using SC and Y&S

5.3.3.2 Simulated Pressure Data

A second example uses the SC and Y&S methods to simulate wind induced pressure data measured on a building rooftop in a separated flow zone. This is a practical example of a highly non-Gaussian process which is commonly encountered in wind engineering, and is the source of roof and building envelope damage in extreme wind events.

Figure 5.6 shows the target PSD measured from the data with PSDs measured from SC and Y&S simulations. Both method represent the target spectrum well in the low frequency range, and SC is able to emulate the target spectrum throughout the entire frequency range. A course frequency resolution provides better delineation of the three spectral estimates, and reduces variance error in the estimates.

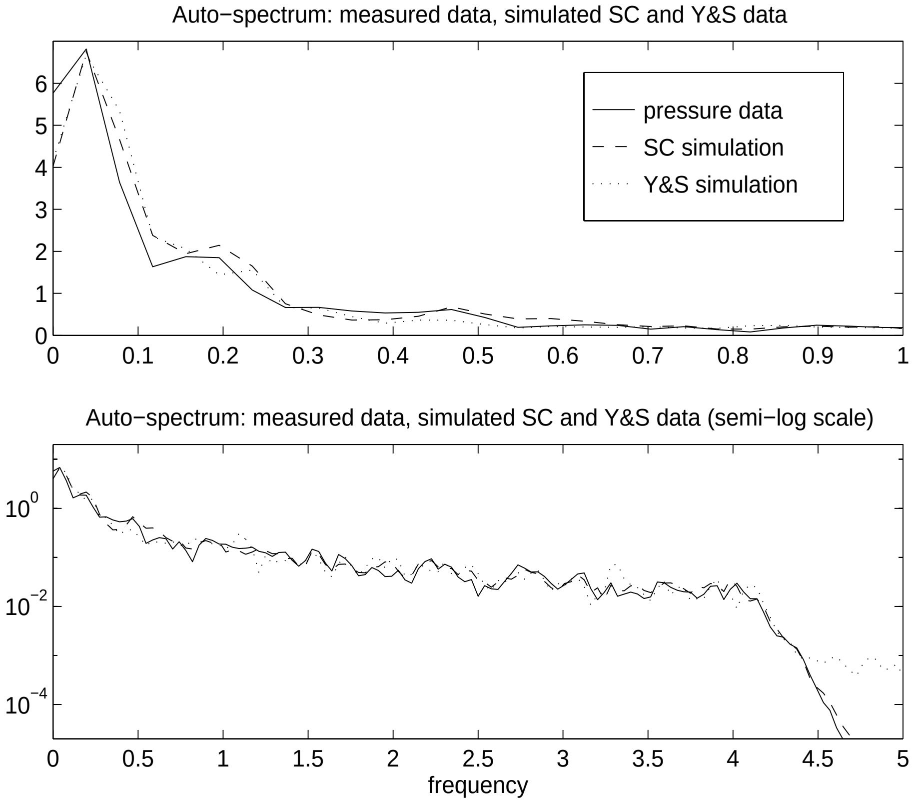


FIGURE 5.6 Pressure data: Comparison of target PSD and simulation PSDs using SC and Y&S.

Figure 5.7 shows the Modified Hermite PDF model used to specify the non-Gaussian characteristics of the pressure simulations, and the estimated PDFs of the two simulations. SC uses a Hermite transformation directly on the time history, and Y&S uses a CDF mapping scheme. The CDF mapping of Y&S leads to a non-Gaussian time history that better represents the PDF model in this case than does the SC transformation. This can be seen in the right tail region in the first and second plots of Fig. 5.7, where the measured data is not presented. The third plot shows the measured data PDF as the bars, and the PDF model and simulation PDFs as solid, dashed, and dotted lines. Now it can be seen that the SC Modified Hermite transformation operating directly on the time series better represents the actual data than does either the PDF Modified Hermite model, or its application in the Y&S simulation. This demonstrates that matching the first four moments through direct Modified Hermite transformation can in some cases better represent measured data than CDF transformation. The bottom two figures are close up views of the tails of the third plot, indicating acceptable left tail representation for both cases, and a better right tail representation using SC simulation. Note that the data and the SC simulation do not cross above zero pressure, while the Y&S simulation does. This is of course a function of the PDF model selected for the CDF transformation, where here the Modified Hermite model is used as a general method for fitting a class of measured data.

The target and simulation skewness and kurtosis are listed in Table 5.1 below. The skewness and kurtosis values for the Y&S simulation under-represent the target values, which is a function of the right tail region. Figure 5.8 shows the measured pressure data, the SC simulation, and the Y&S simulation. Close examination of the region near zero pressure shows the crossing above zero in the Y&S simulation. This will again be addressed in the demonstration of crossing rates in a later section.

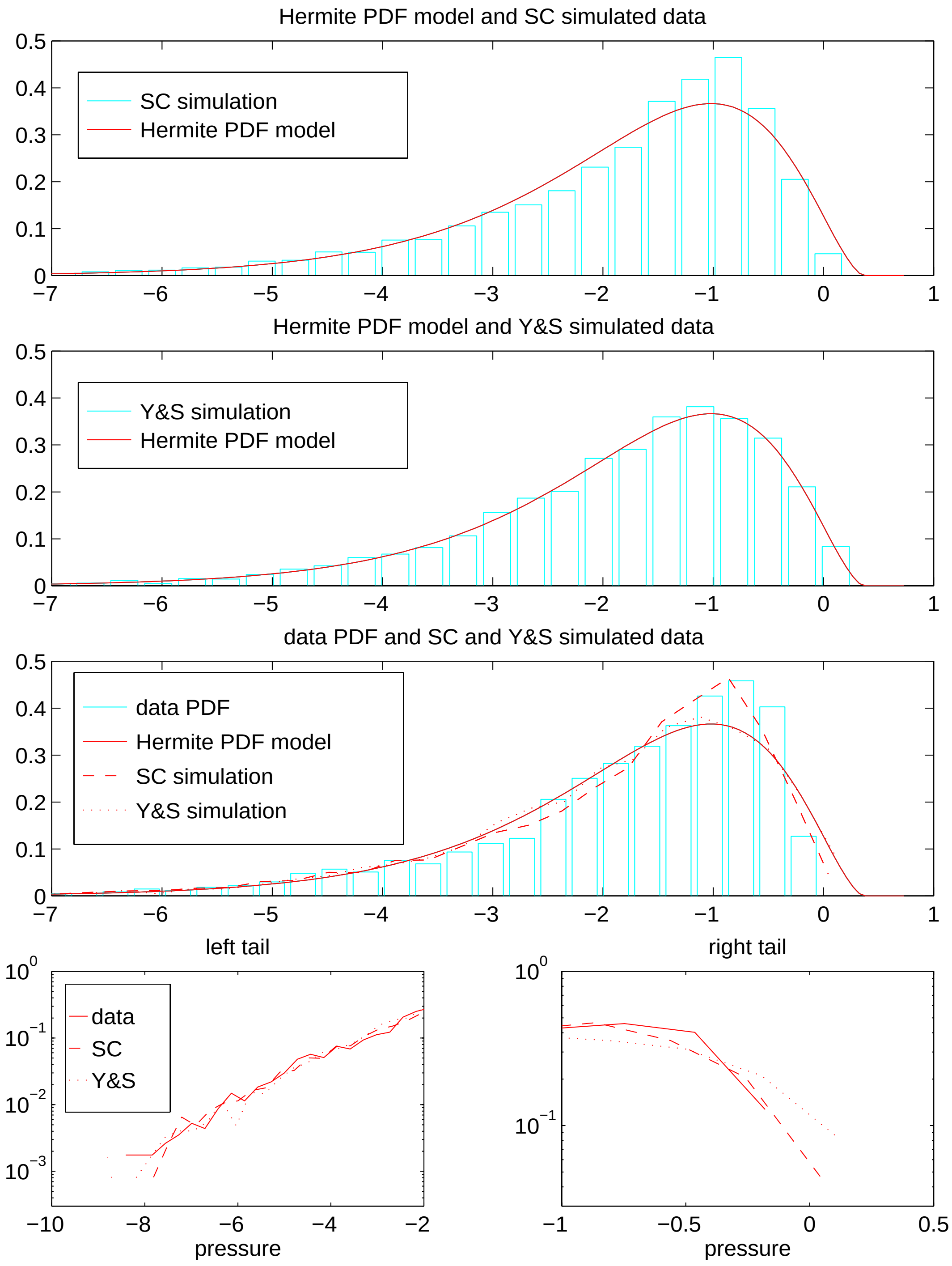


FIGURE 5.7 Pressure data: Comparison of target PDF with simulation PDFs using SC and Y&S.

TABLE 5.2 COMPARISON OF MEASURED TARGET SKEWNESS AND KURTOSIS OF THE PRESSURE DATA WITH THOSE MEASURED FROM THE SC AND Y&S SIMULATIONS OF WAVE ELEVATION.

	skewness	kurtosis
target	-1.40	5.15
SC simulation	-1.35	5.07
Y&S simulation	-0.95	4.56

5.4 Analysis of Stationarity and Convergence

This section first discusses the stationary property of the spectral correction algorithm in detail, then addresses the convergence of the SC iteration scheme.

5.4.1 Stationary Property of Spectral Correction Simulation

The stationary property of the spectral correction simulation algorithm is shown analytically, and then demonstrated numerically through an ensemble of realizations.

5.4.1.1 Analytical Derivation of the Stationary Property of Spectral Correction

The spectral correction algorithm can be shown to produce ensembles of realizations which are stationary. The proof consists of four parts, which are listed here, and then proven in sequence.

1) A process, $y(t)$, generated by the RARP spectral method in section 2.2.1, Eqs. 2.1, 2.3, is Gaussian and strict sense stationary (SSS) for any spectral discretization.

2) A nonlinear static transform of a stationary process is itself stationary.

3) Given a non-Gaussian stationary process $x(t)$ resulting from a static polynomial transform of a Gaussian stationary process $y(t)$,

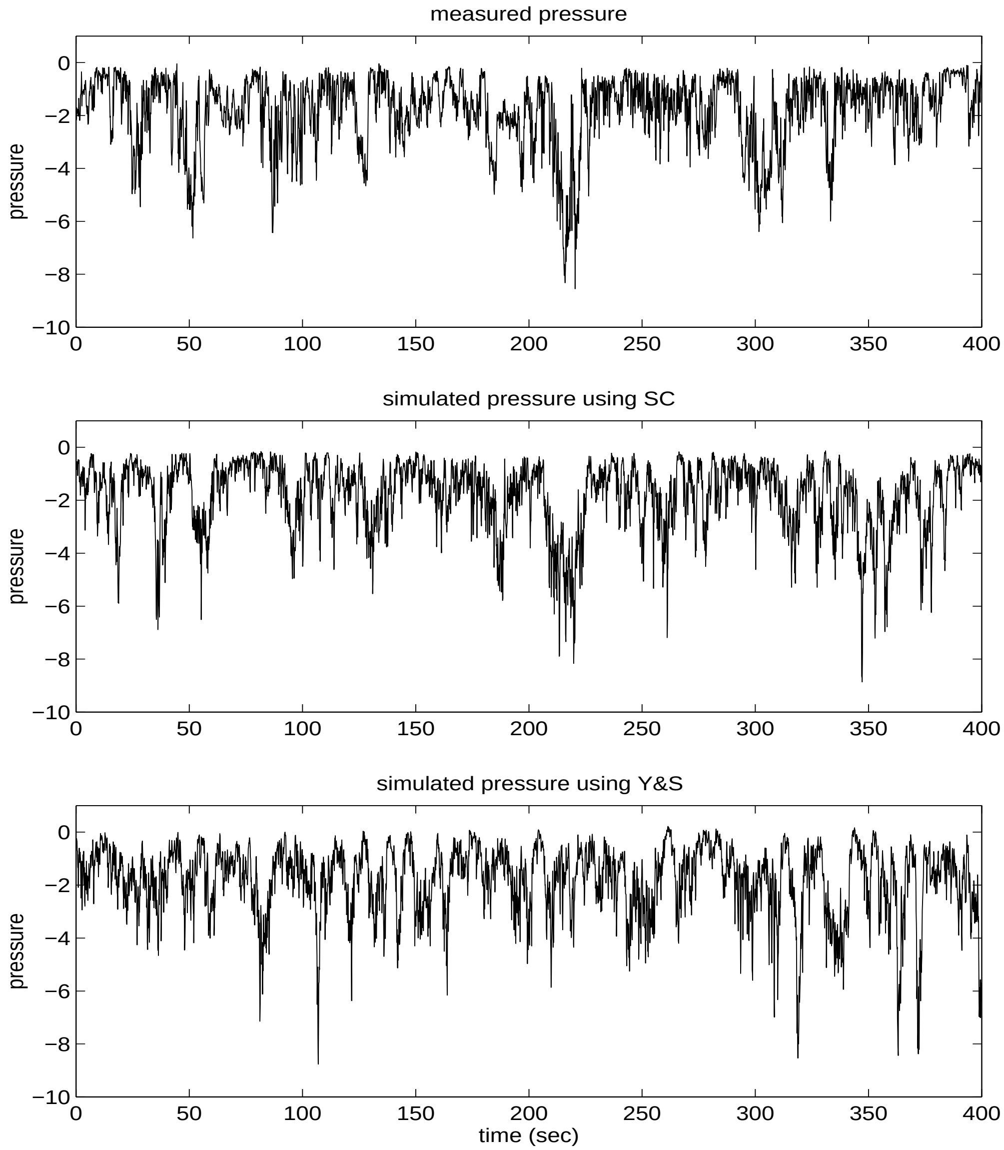


FIGURE 5.8 Measured pressure data and simulations using SC and Y&S

$$x = \sum_{n=1}^{3} c_n y^n , \tag{5.10}$$

changing the amplitude of the Fourier coefficients of $x(t)$ does not affect the stationary property.

4) A process formed by iterative static transformations and amplitude corrections is sta-

tionary if the original process being operated on is stationary.

These four parts proof the stationary property of spectral correction. They are now addressed individually.

Proof of part 1):

A process, $y(t)$, generated by the RARP spectral method in section 2.2.1, Eqs. 2.1, 2.3, is Gaussian and strict sense stationary (SSS) for any spectral discretization.

The equations for RARP simulation are rewritten here

$$y(t) = \sum_{n=1}^{N} (A_n \cos\omega_n t + B_n \sin\omega_n t), \tag{5.11}$$

where A_n and B_n are independent Gaussian random variables with values at equally spaced discrete frequencies ω_n separated by $\Delta\omega$ with covariance

$$E[A_n^2] = E[B_n^2] = G(\omega_n)\Delta\omega, \tag{5.12}$$

$$E[A_n B_m] = 0. \tag{5.13}$$

The process $y(t)$ is a linear combination of Gaussian random variables, as in Eq. 5.11, and is itself Gaussian for any N [23].

Alternatively, Eq. 5.11 can be expressed

$$y(t) = \sum_{n=1}^{N} a_n \cos(\omega_n t + \theta_n). \tag{5.14}$$

where

$$a_n = \sqrt{A_n^2 + B_n^2}, \text{ and } \theta_n = -\mathrm{atan}(B_n / A_n). \tag{5.15}$$

It can be shown [e.g. 73] that the random variables a_n and θ_n are statistically independent due to the Gaussian property of A_n, B_n and the property in Eq. 5.13. Futher, θ_n is uniformly distributed and statistically independent.

The process $y(t)$ is shown to be wide sense stationary if the expected value of the first and second order moments are independent of time. The expected value of the first order moment is shown to be time independent by expanding

$$E[y(t)] = E\left[\sum_{n=1}^{N} a_n \cos(\omega_n t + \theta_n)\right]. \tag{5.16}$$

Using the property that the operations of expectation and summation are commutative, and the independence of a_n and θ_n, Eq. 5.16 is rewritten

$$E[y(t)] = \sum_{n=1}^{N} E[a_n] E[\cos(\omega_n t + \theta_n)]. \tag{5.17}$$

The expectation on a_n is known to be time independent as a function of time independent variables (Eq. 5.15). Now expressing the second expectation as

$$E[\cos(\omega_n t + \theta_n)] = \int_{-\pi}^{\pi} \cos(\omega_n t + \theta_n) f_{\Theta}(\theta) d\theta \ , \tag{5.18}$$

and substituting the uniform distribution for $f_{\Theta}(\theta)$ gives

$$E[y(t)] = \sum_{n=1}^{N} a_n E[\cos(\omega_n t + \theta_n)] = \sum_{n=1}^{N} a_n \frac{1}{2\pi} \int_{-\pi}^{\pi} \cos(\omega_n t + \theta_n) d\theta = 0. \tag{5.19}$$

The expected value of the second order moment is also shown to be time invariant through

$$E[y(t)y(t+\tau)] = E\left[\sum_{n=1}^{N} a_n \cos(\omega_n t + \theta_n) \sum_{m=1}^{N} a_m \cos(\omega_m (t+\tau) + \theta_m)\right]. \tag{5.20}$$

Applying once again the commutative property, Eq. 5.20 can be rewritten as

$$E[y(t)y(t+\tau)] = \sum_{n=1}^{N}\sum_{m=1}^{N} E[a_n a_m]E[\cos(\omega_n t+\theta_n)\cos(\omega_m(t+\tau)+\theta_m)] \,. \tag{5.21}$$

For the case where $n \neq m$, the statistical independence of θ gives

$$E[\cos(\omega_n t+\theta_n)\cos(\omega_m(t+\tau)+\theta_m)] = E[\cos(\omega_n t+\theta_n)]E[\cos(\omega_m(t+\tau)+\theta_m)] \,, \tag{5.22}$$

and the individual expectation terms equal zero following Eq. 5.19.

For the case where $n = m$, Eq. 5.21 is

$$E[y(t)y(t+\tau)] = \sum_{n=1}^{N} E[a_n^2]E[\cos(\omega_n t+\theta_n)\cos(\omega_n(t+\tau)+\theta_n)] \,. \tag{5.23}$$

Using the trigonometric identity $\cos(A)\cos(B) = \frac{1}{2}(\cos(A+B)+\cos(A-B))$ gives

$$E[y(t)y(t+\tau)] = \sum_{n=1}^{N} E[a_n^2]E[\cos((2\omega_n t+\omega_n\tau+2\theta_n)+\cos(-\omega_n\tau))] \,, \tag{5.24}$$

or

$$E[y(t)y(t+\tau)] = \sum_{n=1}^{N} E[a_n^2]\{E[\cos(2\omega_n t+\omega_n\tau+2\theta_n)]+E[\cos(-\omega_n\tau)]\} \,. \tag{5.25}$$

The first cosine expectation term is zero following Eq. 5.19, leaving

$$E[y(t)y(t+\tau)] = R_y(\tau) = \sum_{n=1}^{N} a_n^2\cos(\omega_n\tau) \,. \tag{5.26}$$

The second order moment in Eq. 5.26 is independent of time, and only depends on the time shift τ.

It has been shown that $y(t)$ is wide sense stationary and Gaussian for any N. The Gaussian properties lead to the proof of part 1), that $y(t)$ is strict sense stationary (SSS).

Proof of 2):

A nonlinear static transform of a stationary random process is itself stationary.

Define a process $x(t)$ as the result of a static transform of a process $y(t)$ through the operation g

$$x = g(y)\,. \tag{5.27}$$

$y(t)$ can be expressed also as the inverse operation in terms of $x(t)$ as

$$y = g^{-1}(x)\,. \tag{5.28}$$

The probability of $x(t)$ at some time t is expressed in terms of the probability of $y(t)$ as

$$f_X(x)dx = f_Y(y)dy \;, \tag{5.29}$$

or in terms of the relationships in EQS. 5.27, 5.28 [5, 23]

$$f_X(x) = f_Y(y)\left|\frac{dg^{-1}(x)}{dx}\right| = \frac{f_Y(y)}{|dg(y)/dy|} \quad . \tag{5.30}$$

This describes the marginal distribution of the process $x(t)$ at some time t. The full T dimensional distribution for all discrete time values of $x = x(1\ldots T)$ can also be expressed in terms of the distribution of the individual mapping at each time [23]. Since the operators in both expressions in Eq. 5.30 are independent of time, so too is the full distribution of $x(t)$ for all times $t = 1\ldots T$. This time independence shows that $x(t)$ is stationary if $y(t)$ is stationary. Note that this proof does not require the transformed process $y(t)$ be Gaussian, only stationary.

Proof of 3):

Given a non-Gaussian stationary process $x(t)$ resulting from a static polynomial transform of a Gaussian stationary process $y(t)$, as in $x = \sum_{n=1}^{3} c_n y^n$, changing the amplitude of the Fourier coefficients of $x(t)$ does not affect the stationary property.

Define a non-Gaussian process $x(t)$ as any static transformation of a Gaussian stationary process, $y(t)$ generated using Eqs. 5.11-5.13. $x(t)$ can be expressed as

$$x(t) = \sum_{n=1}^{N} b_n \cos(\omega_n t + \psi_n) , \tag{5.31}$$

where b_n, ψ_n are functions of A_n, B_n in Eq. 5.11, and the transformation $g(y)$. The first and second order moments are expressed as

$$E[x(t)] = E\left[\sum_{n=0}^{N} b_n \cos(\omega_n t + \psi_n)\right], \tag{5.32}$$

$$E[x(t)x(t+\tau)] = E\left[\sum_{n=0}^{N}\sum_{m=0}^{N} b_n b_m \cos(\omega_n t + \psi_n)\cos(\omega_m (t+\tau) + \psi_m)\right]. \tag{5.33}$$

Application of the commutative property now gives

$$E[x(t)] = \sum_{n=0}^{N} E[b_n \cos(\omega_n t + \psi_n)] , \tag{5.34}$$

$$E[x(t)x(t+\tau)] = \sum_{n=0}^{N}\sum_{m=0}^{N} E[b_n b_m \cos(\omega_n t + \psi_n)\cos(\omega_m (t+\tau) + \psi_m)] . \tag{5.35}$$

It is not necessarily trivial to show that the expectations in Eqs. 5.34 and 5.35 yield time invariant functions, since the probabilistic characteristics of the phase ψ_n and amplitude b_n depend on the nonlinear operator g. However, it need not explicitly be shown that these functions are time invariant. $x(t)$ has been defined as the static transform of a SSS process, it is known to be stationary through the proof in part 2).

What needs to be shown to prove part 3 is that when the nonlinear operator $g(\ )$ is a polynomial static transformation, the expectations in Eqs. 5.34 and 5.35 remain time invariant if the random variables b_n are replaced with constants and pulled outside the expectation. Thus, replacing these amplitudes will not effect the stationary property of $x(t)$.

If it can be shown that b_n and ψ_n are statistically independent, the expectations in Eqs. 5.34 and 5.35 can be expressed as

$$E[b_n \cos(\omega_n t + \psi_n)] = E[b_n]E[\cos(\omega_n t + \psi_n)] = \text{time invariant}\,, \tag{5.36}$$

$$\begin{aligned} &E[b_n b_m \cos(\omega_n t + \psi_n)\cos(\omega_m(t+\tau) + \psi_m)] \\ &= E[b_n b_m]E[\cos(\omega_n t + \psi_n)\cos(\omega_m(t+\tau) + \psi_m)]\,, \\ &= \text{time invariant} \end{aligned} \tag{5.37}$$

where the time invariant property is guaranteed by the proof in part 2). Each expectation term is then time invariant, and the proof of part 3) will be complete.

It will now be shown that b_n and ψ_n are statistically independent. Consider first the case where $g(\ )$ is a square operator, giving $x(t) = y^2(t)$. The squaring operation in the time domain is equivalent to convolution in the frequency domain, such that the Fourier components of $x(t)$ can be written in terms of the Fourier components of $y(t)$ as

$$X(f) = \sum_{\alpha = -f_c}^{f_c} Y(\alpha)Y(f-\alpha) \quad . \tag{5.38}$$

It can be shown that the Fourier components of $y(t)$ can expressed in a complex form $Y(f) = A_y(f) + iB_y(f)$, where $A_y(f)$ and $B_y(f)$ are the A_n and B_n from Eq. 5.11, 5.12 with a notation change. The new notation uses a subscript to indicate the process from which the variable is taken, and the argument (f) indicates frequency place of the subscript n.

Eq. 5.38 can be rewritten as

$$X(f) = A_x(f) + iB_x(f) = \sum_{\alpha = -f_c}^{f_c} (A_y(\alpha) + iB_y(\alpha))(A_y(f-\alpha) + iB_y(f-\alpha)) \quad . \tag{5.39}$$

Expanding and expressing in terms of real and imaginary components of $X(f)$ gives

$$A_x(f) = \sum_{\alpha=-f_c}^{f_c} A_y(\alpha)A_y(f-\alpha) - B_y(\alpha)B_y(f-\alpha) \quad , \tag{5.40}$$

and

$$B_x(f) = \sum_{\alpha=-f_c}^{f_c} A_y(\alpha)B_y(f-\alpha) + B_y(\alpha)A_y(f-\alpha) \quad . \tag{5.41}$$

Both Eqs. 5.40 and 5.41 are the sum of products of independent variables for any frequency f, that is

$$E[A_y(\alpha)A_y(f-\alpha)] = E[B_y(\alpha)B_y(f-\alpha)] = 0 \quad , \tag{5.42}$$

and

$$E[A_y(\alpha)B_y(\alpha)] = E[A_y(\alpha)B_y(f-\alpha)] = 0 \quad . \tag{5.43}$$

Thus, each frequency component of $A_x(f)$ and $B_x(f)$ is Gaussian by the central limit theorem. This will be demonstrated numerically later.

Additionally, it can be shown that the real and imaginary components are uncorrelated for any frequency f

$$E[A_x(f)B_x(f)] = 0 \quad . \tag{5.44}$$

Expansion of the expectation above in terms of Eqs. 5.40 and 5.41 gives

$$\begin{aligned} &E[A_x(f)B_x(f)] \\ &E\left[\sum_{\alpha=-f_c}^{f_c} A_y(\alpha)A_y(f-\alpha) - B_y(\alpha)B_y(f-\alpha) \sum_{\gamma=-f_c}^{f_c} A_y(\gamma)B_y(f-\gamma) + B_y(\gamma)A_y(f-\gamma)\right] \\ &= \sum_{\alpha=-f_c}^{f_c}\sum_{\gamma=-f_c}^{f_c} E[(A_y(\alpha)A_y(f-\alpha) - B_y(\alpha)B_y(f-\alpha))(A_y(\gamma)B_y(f-\gamma) + B_y(\gamma)A_y(f-\gamma))] \end{aligned} \tag{5.45}$$

Further expansion inside the expectation leads to the sum of four expressions

$$\begin{aligned} &E[(A_y(\alpha)A_y(f-\alpha)-B_y(\alpha)B_y(f-\alpha))(A_y(\gamma)B_y(f-\gamma)+B_y(\gamma)A_y(f-\gamma))] = \\ &E[A_y(\alpha)A(\gamma)A_y(f-\alpha)B_y(f-\gamma)]-E[B_y(\alpha)B_y(\gamma)A_y(f-\alpha)B_y(f-\gamma)]+ \\ &E[A_y(f-\alpha)A_y(f-\gamma)A_y(\alpha)B_y(\gamma)]-E[B_y(f-\alpha)B_y(f-\gamma)A_y(\gamma)B_y(\alpha)] \end{aligned} \quad . \tag{5.46}$$

Each of the four expected value expressions consists of four statistically independent terms, each one which has an expected value of zero. For the case when $\alpha = \gamma$, each expectation is reduced to three independent terms, two of which have expected values of zero. Thus Eq. 5.44 is shown to hold.

The time domain expression for $x(t)$ in Eq. 5.31 is in terms of polar coordinates, where the amplitude b_n and phase ψ_n are

$$b_n^2 = A_x(n)^2 + B_x(n)^2 , \tag{5.47}$$

$$\psi_n = \operatorname{atan}\left(\frac{B_x(n)}{A_x(n)}\right) . \tag{5.48}$$

Using the results that $A_x(n)$ and $B_x(n)$ are Gaussian and uncorrelated, it can be shown that b_n and ψ_n are statistically independent [73].

Using the above proof of Eqs. 5.36 and 5.37, Eqs. 5.34 and 5.35 can be rewritten

$$E[x(t)] = \sum_{n=0}^{N} E[b_n]E[\cos(\omega_n t+\psi_n)] = \text{time invariant} , \tag{5.49}$$

$$\begin{aligned} E[x(t)x(t+\tau)] &= \sum_{n=0}^{N}\sum_{m=0}^{N} E[b_n b_m]E[\cos(\omega_n t+\psi_n)\cos(\omega_m(t+\tau)+\psi_m)] \\ &= \text{time invariant} \end{aligned} , \tag{5.50}$$

where the time invariant property is guaranteed by the proof in part 2). For Eqs. 5.49 and 5.50, the individual expectations are each time invariant.

It can be seen now that the results of the expectation operating on the cosine terms in Eqs. 5.49 and 5.50 is time invariant. The expectation on the amplitude variables may then be replaced with constants without affecting the stationary property of $x(t)$.

The above proof is for the case where $x = y^2$. The same results are found when the operator is a cubing function. In this case Eq. 5.38 becomes

$$X(f) = \sum_{\alpha=-f_c}^{f_c} \sum_{\gamma=-f_c}^{f_c} Y(\alpha)Y(\gamma)Y(f-\alpha-\gamma) \quad . \tag{5.51}$$

The Gaussian property of the components $A_x(f)$ and $B_x(f)$, and their statistical independence is demonstrated numerically in Fig. 5.9 and Table 5.3, where in this case $A_x(f)$ and $B_x(f)$ are the result of a third order polynomial transformation of a Gaussian process. 10,000 realizations of a Gaussian process are simulated based on a wide banded wind spectrum. Each realization is passed through a third order Hermite polynomial transformation with coefficients chosen to produce skewness and kurtosis of 0.1 and 5.0, respectively. The first four moments are estimated at each frequency through the ensemble average of the 10,000 realizations for both $A_x(f)$ and $B_x(f)$. Table 5.3 gives the statistics of the skewness and kurtosis in terms of their mean and standard deviation about their expected values of zero and three, respectively. Table 5.3 shows the same information for the components, $A_y(f)$ and $B_y(f)$, of the Gaussian simulations before transformation. These components are know to be Gaussian and orthogonal, and have the same variance about their expected values as do the components of the non-Gaussian process.

The ensemble averaged expected values $E[A_x(f)B_x(f)]$ and $E[A_y(f)B_y(f)]$ are shown at every frequency in Figure 5.9. These plots and Table 5.3 demonstrate the Gaussian property and orthogonality between $A_x(f)$ and $B_x(f)$.

Since every real and imaginary component is Gaussian and SI, the phase will be independent of the magnitude, and the proof in part 3) holds.

TABLE 5.3 MEAN AND STD OF SKEWNESS AND KURTOSIS ESTIMATES OF THE FOURIER COMPONENTS FROM A NON-GAUSSIAN AND GAUSSIAN SIMULATION

	$Re_x(f)$	$Im_x(f)$	$Re_y(f)$	$Im_y(f)$
mean skewness	0.001	.001	0.01	0.004
std skewness	0.058	0.061	0.055	0.056
mean kurtosis	2.999	3.006	3.007	3.008
std kurtosis	0.111	0.116	0.107	0.1130

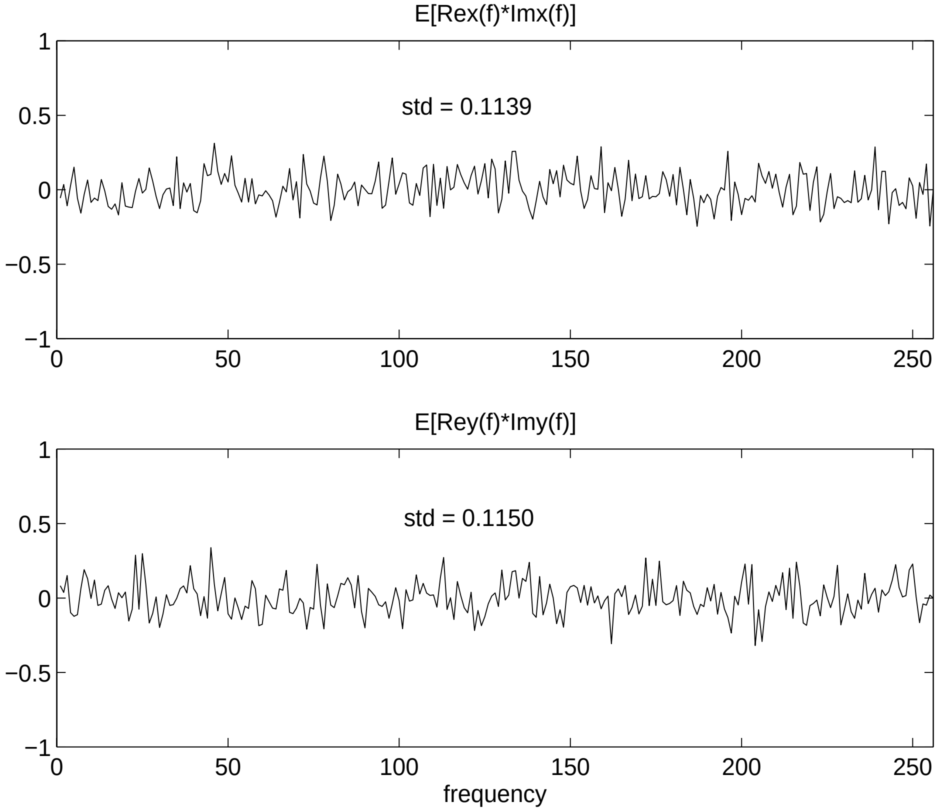


FIGURE 5.9 Expected value of the product of the real and imaginary Fourier components at every frequency through an ensemble of 10,000 realizations. top: non-Gaussian simulation. Bottom: Gaussian simulation.

It has been shown that a third order polynomial transformations of a Gaussian process is stationary when its amplitude coefficients are replaced with constants after transformation.

Proof of 4):

A process formed by iterative static transformations and amplitude corrections is stationary if the original process being operated on is stationary. Static transform and amplitude correction operations have been shown to preserve the stationary property, so a process formed by iterative static transforms and amplitude corrections is stationary if the original process being operated on is stationary. This proof follows directly from parts 1) through 3).

5.4.1.2 Numerical Demonstration of the Stationary Property

The stationary property of the spectral correction algorithm is here numerically demonstrated by showing time invariance through multiple realizations. The example simulates 10,000 zero mean unit variance realizations of a non-Gaussian sea state with 2048 seconds per realization using a unit sampling frequency. The skewness and kurtosis values are 0.5 and 6.0, respectively. 10,000 Gaussian realizations are simulated using RARP. A narrow banded JONSWAP spectrum is used as the target, with a peak factor of 3.3, a peak frequency of 0.1 Hz, and a sampling rate of 1.0 Hz.

The auto-correlation is measured through the ensemble centered at eight evenly spaced time intervals

$$E[x(t_i)x(t_i+\tau)]\,, \tag{5.52}$$

where $t_1 = 1, t_2 = 257, t_3 = 513, ..., t_8 = 1793$ seconds, and $\tau = 1...512$ seconds.

Figure 5.10 shows the superposition of resulting auto-correlation at eight times for the non-Gaussian and Gaussian simulations, and their total superposition. It is clear that no significant difference exists, demonstrating the independence of frequency content with respect to time.

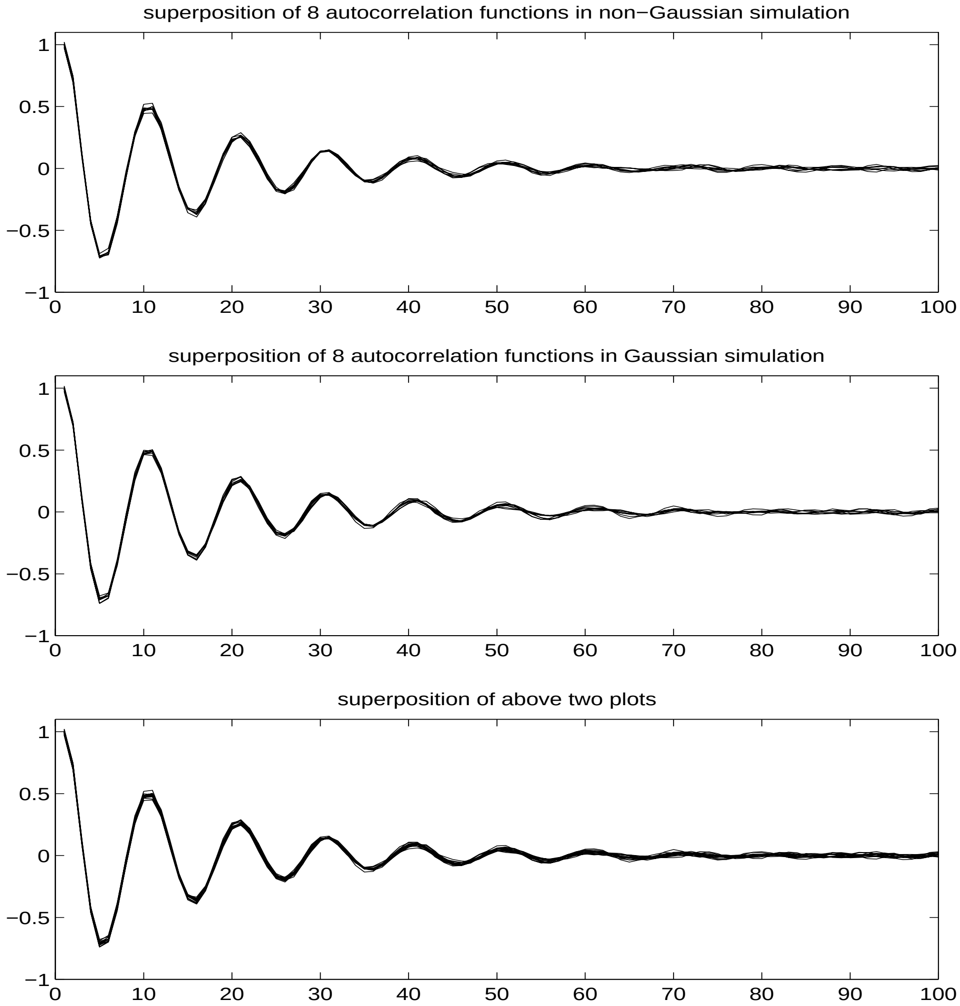


FIGURE 5.10 Superposition of auto-correlation function centered at eight evenly spaced times. Top: non-Gaussian simulation. middle: Gaussian simulation. bottom: superposition of top and middle.

The first four moments are also calculated at each time step through the ensemble. Figure 5.11 is a plot of the first four normalized moment histories, and Fig. 5.12 shows these results for Gaussian simulations. There are no obvious trends in the moment histories of the Gaussian realizations in Fig. 5.12. No trends are observed in the moment histories in Fig. 5.11 using non-Gaussian SC simulation, which supports the stationary hypothesis.

In order to quantify any trends in the moment histories that indicate nonstationary behavior, reverse arrangement tests are performed on each of the eight moment histories in Figs. 5.11 and 5.12. This test uses the number of changes in sign in a sequence of statisti-

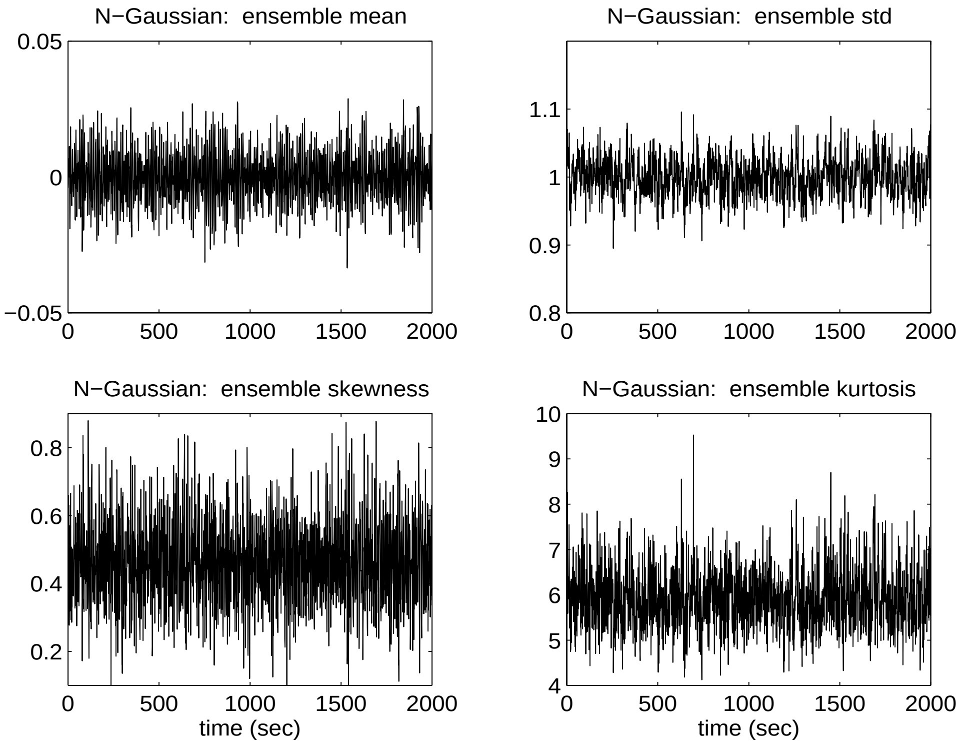


FIGURE 5.11 First four normalized moments at each time step through an ensemble of 10,000 realizations of a non-Gaussian process simulated using SC.

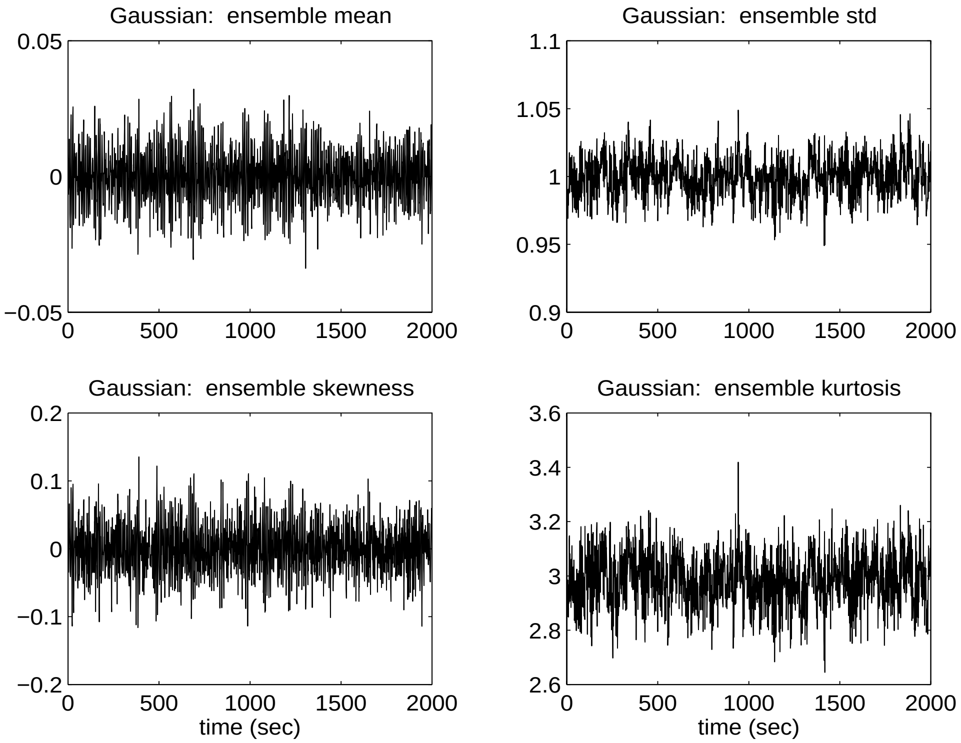


FIGURE 5.12 First four normalized moments at each time step through an ensemble of 10,000 realizations of a Gaussian simulated process.

cal measurements to estimate the likelihood of trends in data sets. For the tables that follow, a larger number of reverse arrangements indicates a greater degree of certainty in the assumed stationary hypothesis. Details of the reverse arrangement test may be found in [6].

Table 5.4 shows the results of this test for all eight histories representing the first four normalized moments through the ensemble of 10,000 realizations for the Gaussian and non-Gaussian simulations. Every 32nd point in the moment history is used in the test. There are at least as many reverse arrangements in the moment histories from the non-Gaussian simulations as there are for the Gaussian moment histories. This indicates that, with at least as much certainty as for the Gaussian moment histories, no trends can be identified in the statistics of the non-Gaussian ensemble of processes. To the degree that the Gaussian simulation method is accepted as stationary, so too can the SC method.

TABLE 5.4 REVERSE ARRANGEMENT TEST RESULTS ON MOMENTS HISTORIES IN FIGS. 5.11, 5.12.

	Reverse arrangements			
	mean	**std**	**skewness**	**kurtosis**
Gaussian simulation (Fig. 5.12)	953	926	872	969
non-Gaussian simulations (Fig. 5.11)	1201	1154	1269	1125

As a final example, consider the justification of the stationary assumption for a single realizations of measured data. A single very long realization of a non-Gaussian process is generated with target skewness and kurtosis values of 0.7 and 6.0, respectively. A corresponding Gaussian simulation of the same length and target spectrum is generated. The data is divided into 64 segments, and the first four moments are calculated in each segment. The reverse arrangement test is applied to both sets of four moments, presented in table 5.5. Comparison of the results of the reverse arrangement test on the Gaussian and non-Gaussian realization indicate at least as great a degree of certainty of the stationary hypothesis for the non-Gaussian realizations. Thus, the non-Gaussian simulation is accepted as stationary with the same confidence as the Gaussian realization.

TABLE 5.5 REVERSE ARRANGEMENT TEST RESULTS ON A SINGLE REALIZATION

reverse arrangements	**Gaussian simulation**	**non-Gaussian simulation**
mean value	1189	1254
std	1115	1160
skewness	1063	1130
kurtosis	923	1065

5.4.2 Convergence of Spectral Correction

The user specifies the exit criterion based on either the error between simulated and target moments, or the error between the simulated and target spectrum. The SC algorithm iteratively converges upon both target moments and a target spectrum, and it is shown that the convergence of both are directly related. The schematic of the SC algorithm shown in Fig. 5.1 is given in a simplified form in Fig. 5.13.

Referring to the schematic, each iteration consists of a moment correction section, producing a process, x, which has the target moments and a spectrum which is distorted from the target G^T. x is passed to the spectral correction section, where x is transformed to x_c, which has no spectral distortion, but does have moment distortion from the target moments m_k^T. After each iterative correction to the spectral contents of x, the resulting

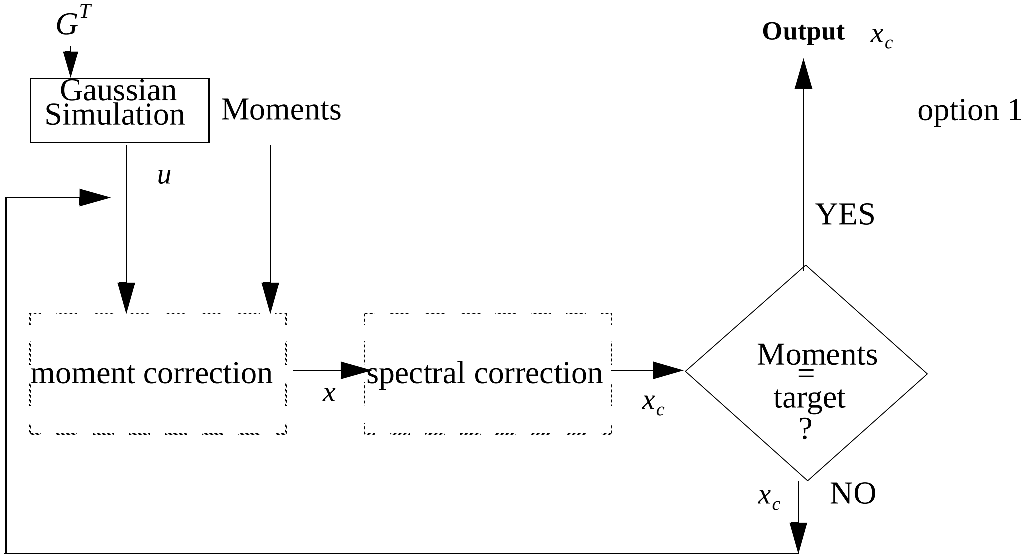


FIGURE 5.13 Schematic of SC algorithm with error in moments as the exit criterion.

moments of x_c are closer to the target than the previous iteration, thus achieving convergence.

This moment convergence in x_c is a function of the decreasing distortion in the spectrum of x with each iteration. In terms of the Fourier components, the amount of correction from x to x_c, $|X|^2 \to G^T = |X_c|^2$, becomes smaller after each iteration, until the correction $x \to x_c$ is small enough such that the moments of x_c are within acceptable error.

5.4.2.1 An Illustration of Convergence

This convergence process is shown in several steps in Fig. 5.14, in terms of a hypothetical target auto-correlation function, R^T, and the measured auto-correlation, R^x, of the process x through four iterations. The steps to convergence are as follows:

step 1) The initial Gaussian simulation, u is sent to the moment correction section to produce x_1. This results in a distortion in the auto-correlation R_1^x, shown in Fig. 5.14 in the top left.

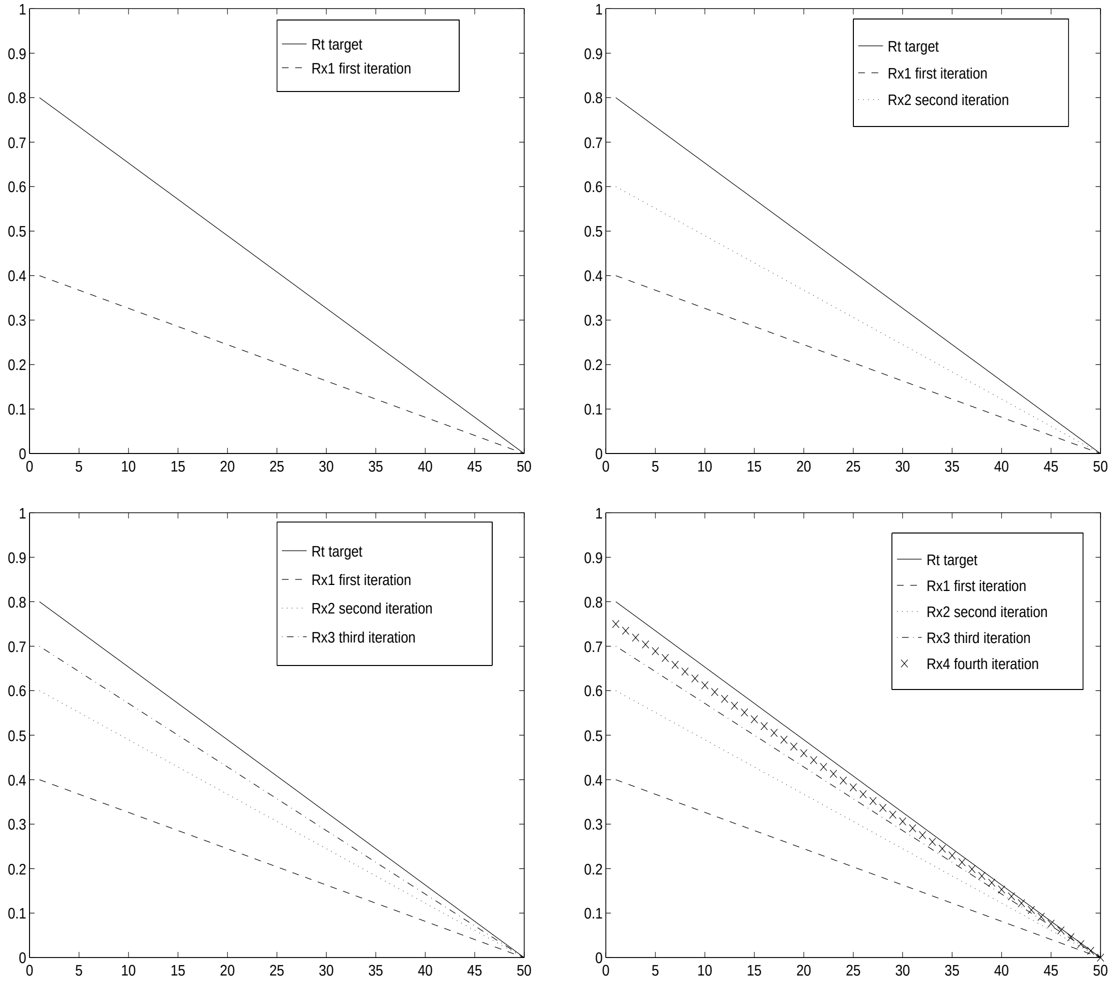


FIGURE 5.14 Illustration of the convergence of the auto-correlation function of x to its target value through four iterations (hypothetical demonstration).

step 2) x_1 is sent to spectral correction to restore R^T, producing the first iteration of x_c, denoted x_{c1}. The transformation $x_1 \rightarrow x_{c1}$ consists of small corrections to the amplitude of the Fourier coefficients of x_1. Since the change in amplitude is small, x_{c1} is still non-Gaussian, but not as severely as x_1, and it is also true that $R^T = R^{x_{c1}}$.

step 3) The second iteration begins by correcting the moments of x_{c1} to produce $x_{c1} \rightarrow x_2$ through another moment correction transformation. Now since the non-Gaussian properties of x_{c1} and x_2 are closer than in the first iteration, $u \rightarrow x$, the moment correction is less severe. The distortion between R_2^x from R^T is also less severe, as shown in the top right plot in Fig. 5.14.

step 4) The distortion in R_2^T is corrected to produce x_{c2}. Now, since the spectral distortion being corrected is less severe than in the first iteration, the moments of x_{c2} are less distorted from the target than were x_{c1}.

These same steps are repeated, with each iteration correcting smaller distortions in the moments and spectrum, until the moments of x_c are within an acceptable tolerance of the target moments m_k^T, or the auto correlation R^x is within acceptable tolerance of R^T. This demonstrates that the convergence of both the spectrum and moments are related, and happen simultaneously. Thus either exit option (sections 5.3.1 and 5.3.2) is appropriate for exiting the SC algorithm.

5.4.2.2 Example of the Convergence of an Actual Simulation

The hypothetical example in Fig. 5.14 is complemented in this section with a demonstration of the convergence of an actual realization. A wide banded target spectrum is employed to simulate a realization of non-Gaussian wind velocity. The distorted spectrum of x and the distorted moments of x_c converge with iteration as demonstrated in Fig. 5.15. Figure 5.16 shows the convergence of the skewness and kurtosis of x_c with iterations, and the sum of the percent difference between the target and simulation values. The distortion of skewness and kurtosis in x_c, seen in Fig. 5.16, is presented in Table 5.6 along with the target skewness and kurtosis through six iteration.

TABLE 5.6 THE TARGET SKEWNESS AND KURTOSIS, THE SIMULATED VALUES IN x_c, AND THE SUM OF THE SKEWNESS AND KURTOSIS PERCENT DIFFERENCE.

	skewness	kurtosis	sum of % diff. from target
target	0.4	6.0	0.0
1st iteration	0.232	4.25	71.135 %
2nd iteration	0.312	5.08	37.410 %
3rd iteration	0.353	5.52	19.621 %
4th iteration	0.375	5.76	10.312 %
5th iteration	0.387	5.87	5.436 %
6th iteration	0.393	5.94	2.867 %

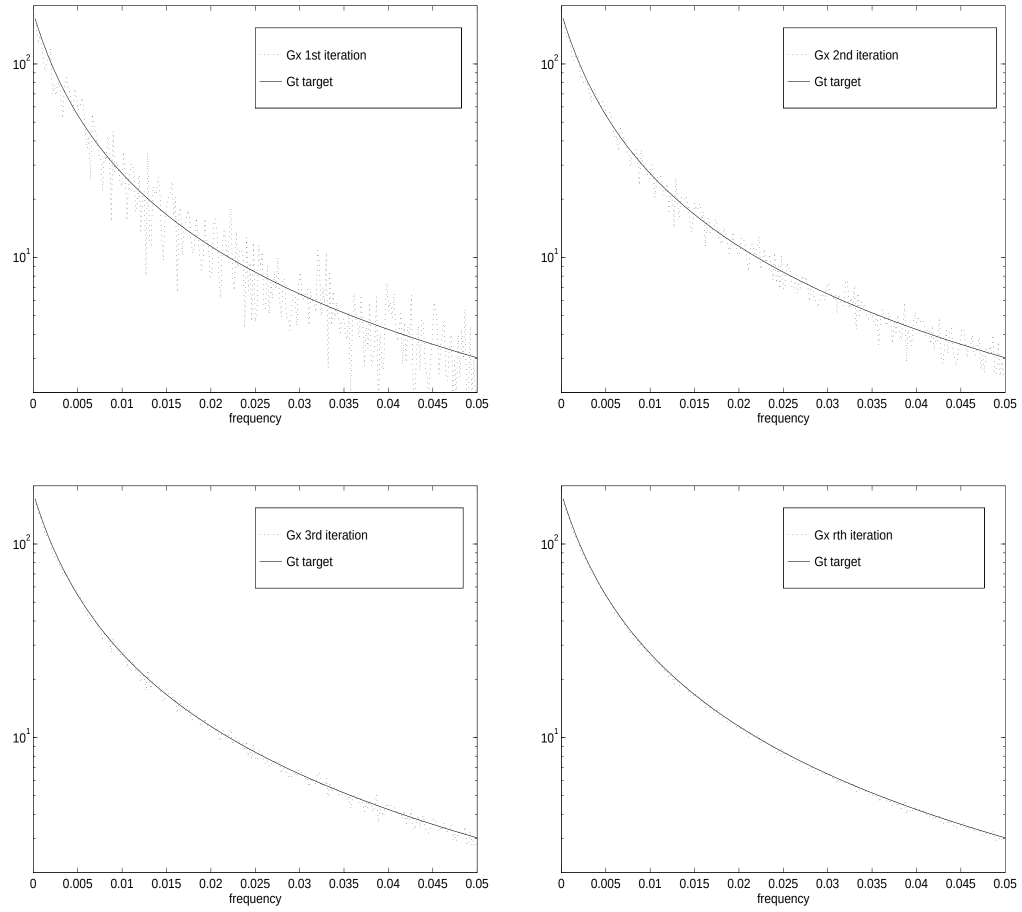


FIGURE 5.15 Demonstration of the convergence of the auto-spectrum of x to its target value through four iterations (actual realization).

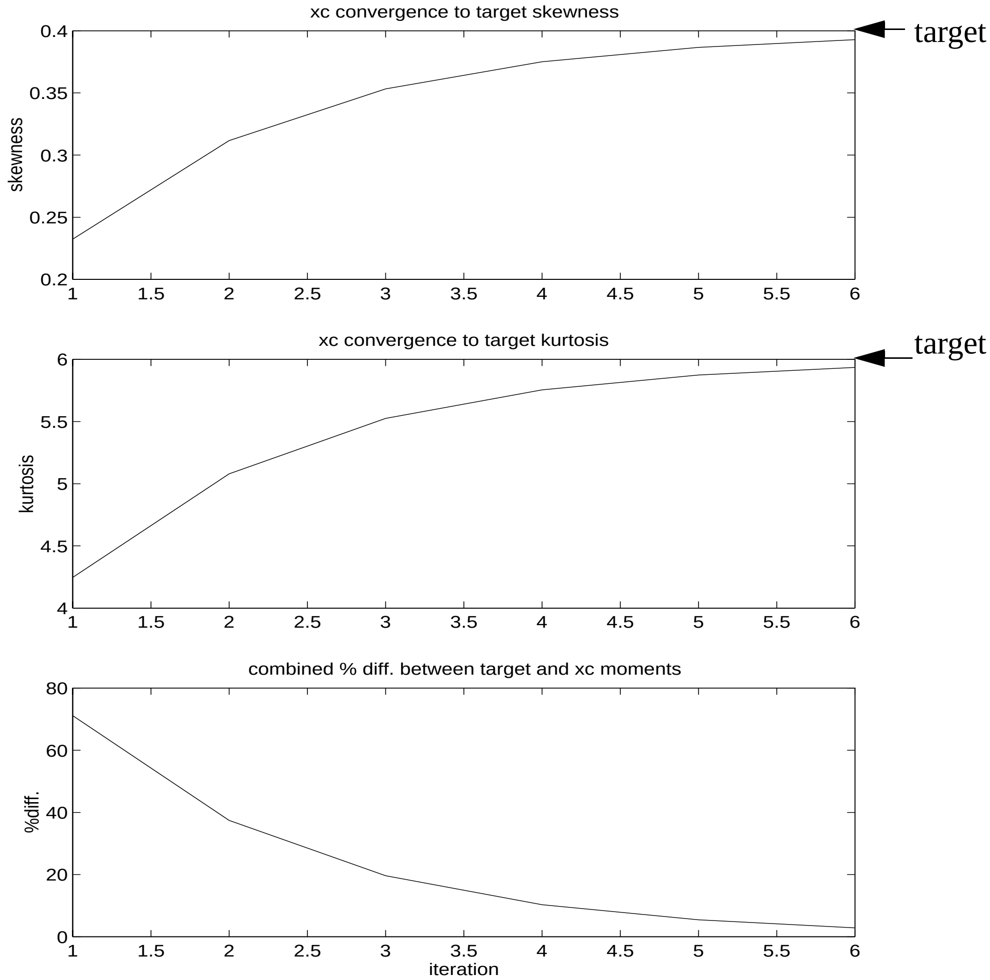


FIGURE 5.16 Convergence of the normalized moments of x_c to their target values through six iterations (actual realization corresponding to Fig. 5.15).

5.5 Target Spectral Options

It was shown in Chapter 2 on Gaussian simulation methods that both random amplitude random phase (RARP) and deterministic amplitude random phase (DARP) are commonly used spectral simulation techniques. Although both have been shown by Grigoriu [22] to be appropriate under most conditions, some preference has been given to RARP in other literature. Using RARP, the power spectrum of a particular realization matches the target spectrum only in an ensemble averaged sense, but individual realizations do not match the target spectrum. This can be argued to be a more accurate reflection of the nature of a real stochastic process, where the target spectrum is typically obtained through an ensemble averaging procedure.

It is proposed to use spectral correction mainly to generate input for Monte Carlo simulation of systems subjected to non-Gaussian loads. A Monte Carlo analysis of system response typically requires the results of a large ensemble of response simulations in order to reduce variance error in the resulting statistical output. In light of the variance of spectral contents in naturally occurring system inputs, some attempt should be made to emulate spectral variance in simulated input.

The generation of a realization using the spectral correction method relies on correcting the spectral amplitude to a deterministic target spectrum in each iteration. The power spectrum of each realization will match the target spectrum, and thus spectrum correction can be classified as a DARP method. In order to retain the desired spectral variance between individual realizations, the spectral correction algorithm includes an option to emulate this variance through pseudo RARP type realizations.

The pseudo RARP option is accomplished by generating a randomized target spectrum which is used as the deterministic target for a spectral correction realization. An individual simulated realization will match this randomized target spectrum using the standard spectral correction algorithm. The next realization uses another random realization of the original target spectrum as its deterministic target. The ensemble will then consist of realizations generated using the DARP spectral correction method, where the deterministic amplitude (DA) to be matched is determined by a random amplitude (RA) realization of the target spectrum G^T. The simulated realizations will then match the original target spectrum in an ensemble sense.

The RA realizations of the DA are generated by simulating a Gaussian realization using Eqs. 5.1 and 5.3. The resulting randomized version of the target spectrum is retained as the DA for a single realization using spectral correction.

5.6 Transformation Options

Central to SC is the static transformation of a Gaussian realization to non-Gaussian. The characteristics of the desired non-Gaussian process are often described in terms of either higher moments or an analytical PDF. While an analytical PDF contains more information than the first four moments, an analytical PDF is not always available. In these cases the first four moments can be adequate information to develop a static transform based on an assumed form of analytical PDF.

The SC algorithm is capable of using either the first four moments or a PDF model to produce a process with the associated non-Gaussian characteristics. These two options are discussed briefly below.

5.6.1 Modified Hermite Transformation (first four moment method)

It was shown in the chapter concerning PDF modelling that the modified Hermite moment based PDF model provides an adequate representation of experimental data using the first four moments. The modified Hermite transformation is adapted to operate directly on a time history such that its moments match the target values. This has been addressed previously in sections 3.2.3 and 5.2. The static transformation applied to the normal Gaussian process u is of the form

$$x = u + \hat{h}_3(u^2 - 1) + \hat{h}_4(u^3 - 3u) , \tag{5.53}$$

where the coefficients are defined through the optimization discussed in Section 5.2

This option is applicable for the simulation of experimentally measured data when it can be shown that the modified Hermite moment PDF model (section 4.3.3) is a good representation of the data histogram. This option has been employed thus far for all examples of SC, and is preferred for its flexibility.

5.6.2 Transformation Using a PDF Model

The SC algorithm can employ a PDF model as the source of information about the characteristics of the desired non-Gaussian process. This adds a great deal of flexibility to the method, since the Hermite transformation may not always be the most appropriate PDF model.

The Modified Hermite transformation in Eq. 5.53 is replaced with

$$x = F^{-1}[\Phi(u)] , \tag{5.54}$$

where Φ is the cumulative distribution function of a normal Gaussian process, and F^{-1} is the inverse cumulative distribution function derived from the analytical PDF of the target non-Gaussian process. SC applies the transform in Eq. 5.54 through a direct mapping technique, and eliminates the need for an analytical inverse function.

The error between target and simulated moments is not optimized explicitly when Eq. 5.54 is applied as the nonlinear transformation. It is more appropriate in this case to use the iteration termination option outlined in Fig. 5.2, where error between the target and simulation spectrum is measured after the moment correction section.

5.6.2.1 Example of CDF Transformation Based Spectral Correction Simulation

The chapter on PDF modelling showed cases where the histogram of measured full scale pressure data was modeled most appropriately using a modified maximum entropy model referred to as MEM II. This model is applied here as input to the spectral correction algorithm to represent the non-Gaussian characteristics of the measured pressure data. Fig. 5.17 shows the measured data and spectral correction simulations using the CDF transformation option with the MEM II model, and the four moment Modified Hermite transformation option. Figure 5.18 compares the PDFs of the simulations with the MEM II model and the measured data PDF. Although the MEM II PDF model fits the data better than the Modified Hermite PDF model, it is of little consequence in this simulation example.

5.7 Crossing Rates using SC and Y&S Simulation

An important statistical measurement of non-Gaussian processes is the crossing rate of extreme values. This study compares the crossing rate of measured data with simulated data using Spectral Correction (SC) and the Yamazaki and Shinozuka method (Y&S). SC is applied using both the raw amplitude coefficients of the Fourier transform of the data as the target spectrum, and using a ensemble averaged target spectrum measured from the data and applied using the pseudo RARP method described in section 5.5. The average crossing rate of 50 realizations of each method are compared with the measured data.

A realization of both measured pressure data and measured tension leg platform surge response data are used in this study. The measured records are shown in Fig. 5.19.

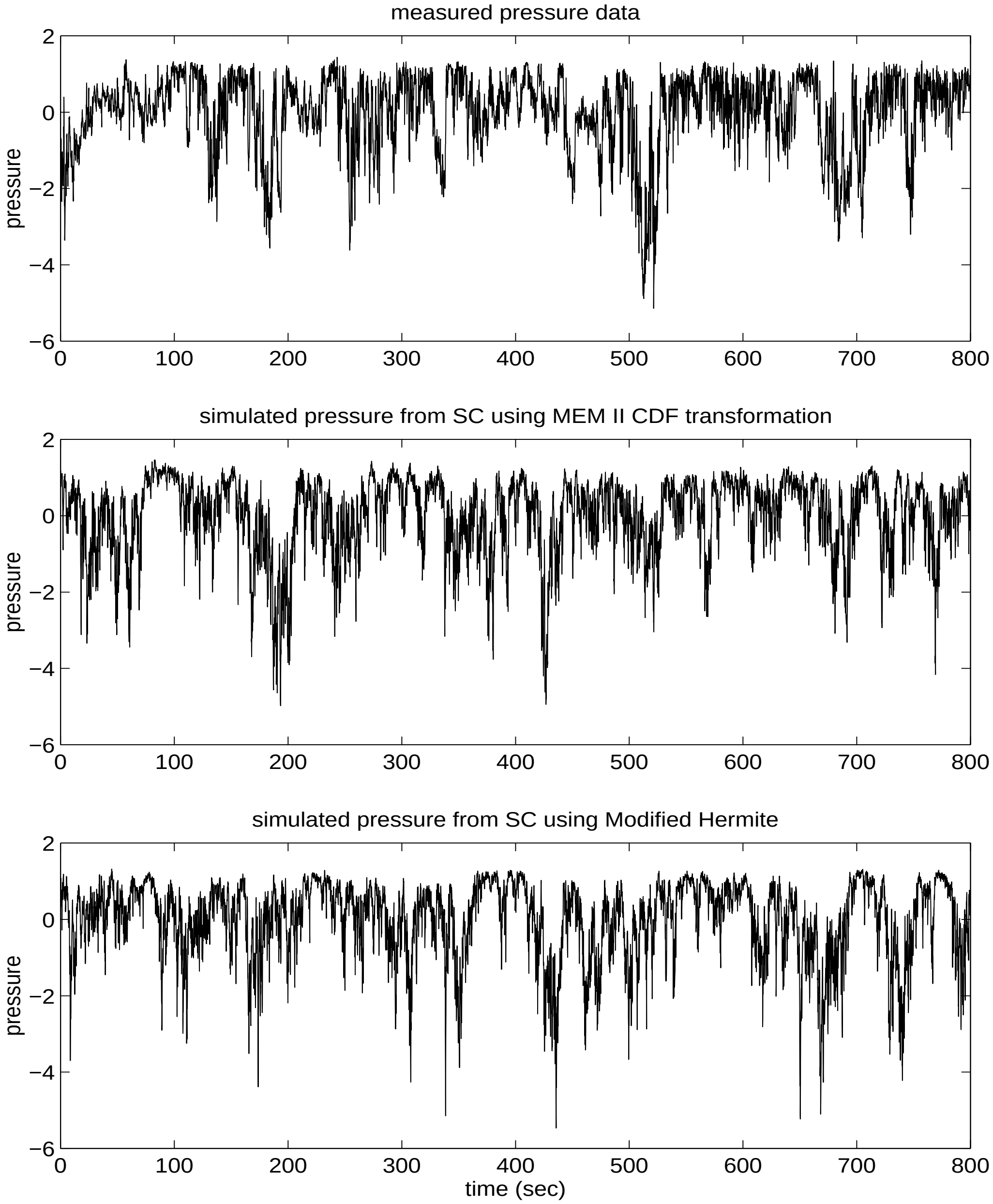


FIGURE 5.17 Measured pressure data and two simulations using : middle: Spectral Correction with CDF transformation option using MEM II PDF model, bottom: Spectral Correction with four moment Modified Hermite transformation.

5.7.1 Case 1: Pressure Data

The measured pressure record used in section 5.3.3.2 is used here to describe the target spectral and non-Gaussian characteristics of the simulations. Figure 5.20 shows the abso-

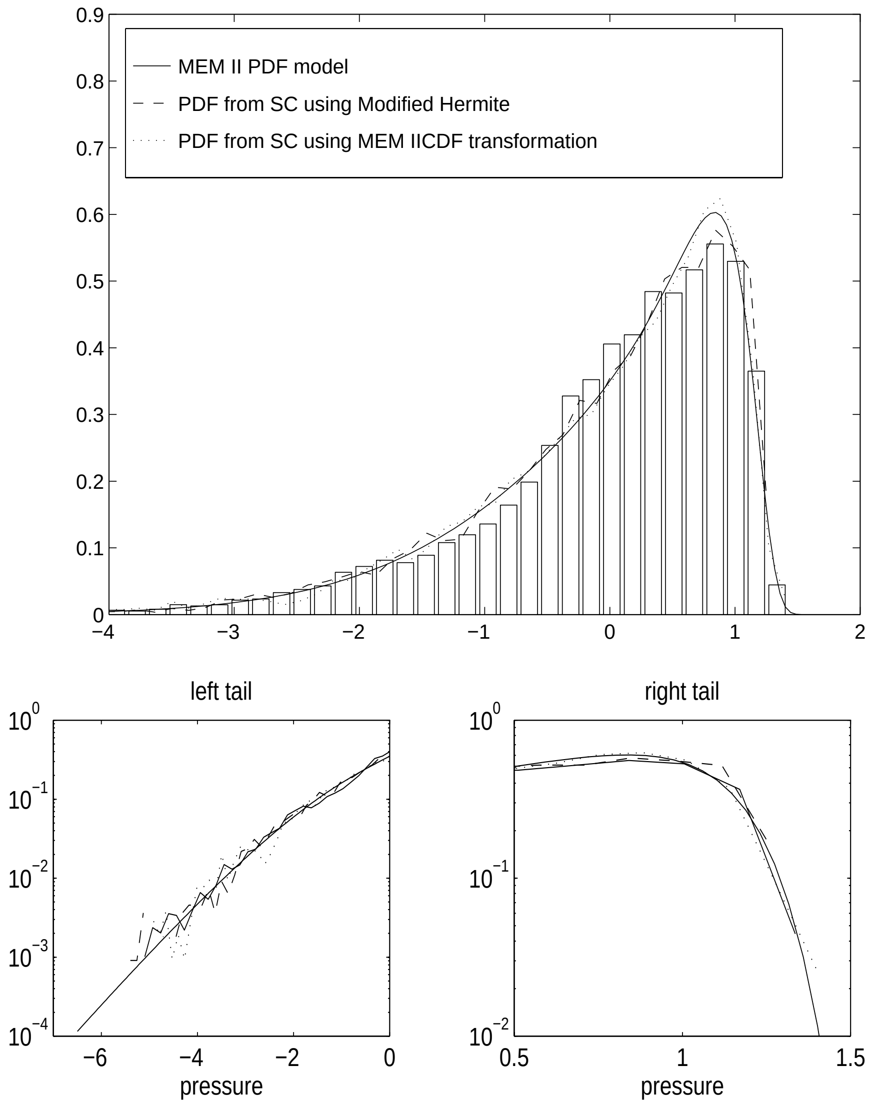


FIGURE 5.18 Measured PDF of data, MEM II model, and simulations from Fig. 5.17

lute value of the crossing rate measured from the data, and from the three simulations. These three simulations are denoted in the legend as SC-f, SC-r, and Y&S, referring to SC simulation using the raw amplitude coefficients of the Fourier transform of the data as the target spectrum, SC simulation using the pseudo RARP method described in section 5.5, and the Yamazaki and Shinozuka method, respectively. The figure shows acceptable reproduction of the measured crossing rate in the right extreme for all three simulations. Near zero the Y&S method does not reproduce the measured crossing rate well. This was discussed in section 5.3.3.2.

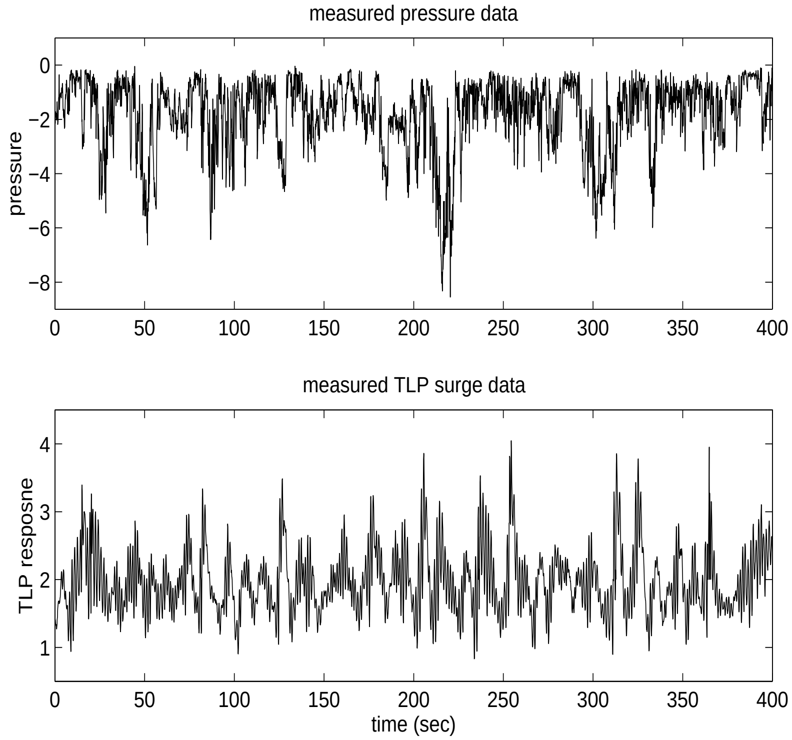


FIGURE 5.19 Measured pressure and TLP surge records used in section 5.7.

5.7.2 Case 2: TLP surge data

The response of a tension leg platform (TLP) to random wind and waves is measured in an experimental facility. The horizontal response parallel with incoming waves, surge, is the measured data under consideration. Figure 5.21 compares the measured crossing rate from data and from the three simulations. In this case all three methods produce acceptable representations over the range of measured data.

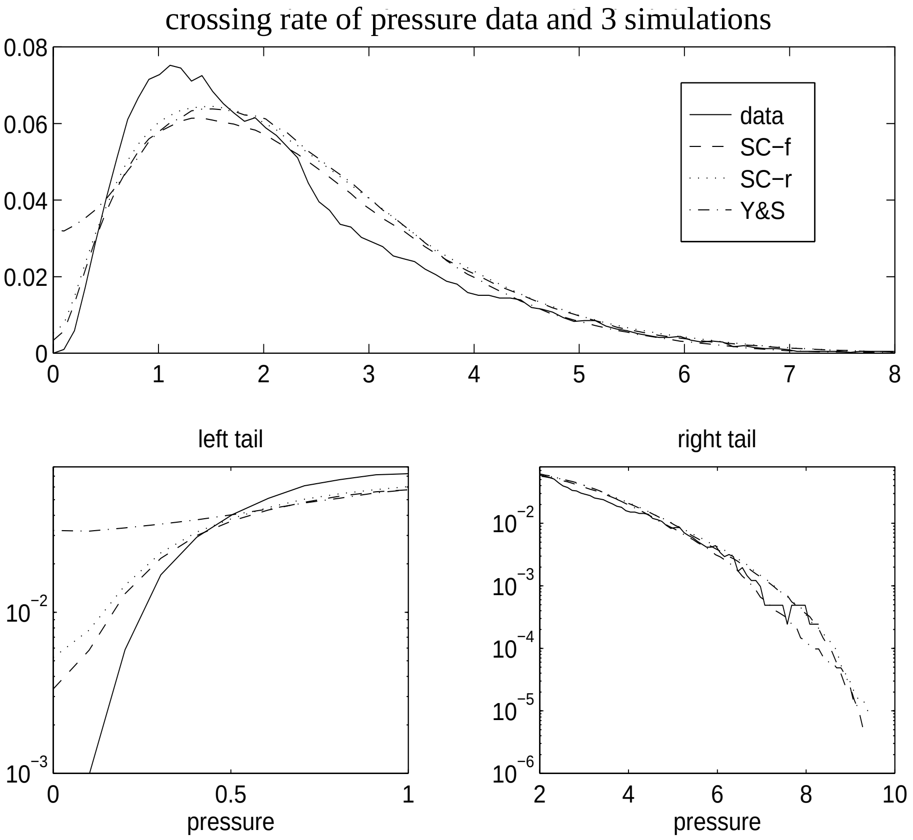


FIGURE 5.20 Crossing rate of pressure data and the average crossing rate of 50 realizations using three simulation methods.

5.8 Concluding Remarks

This chapter presents the uni-variate spectral correction simulation method (SC), which is capable of producing time history realizations that match both probabilistic and spectral input information. The stationary and convergence properties are discussed as well as several options in the algorithm. Several examples are presented and results are compared with an existing technique. It is shown that SC performs well, and succeeds even in special cases where the existing method fails to correctly reflect the target spectrum.

Spectral correction is extended to multivariate simulation in the next chapter, and is

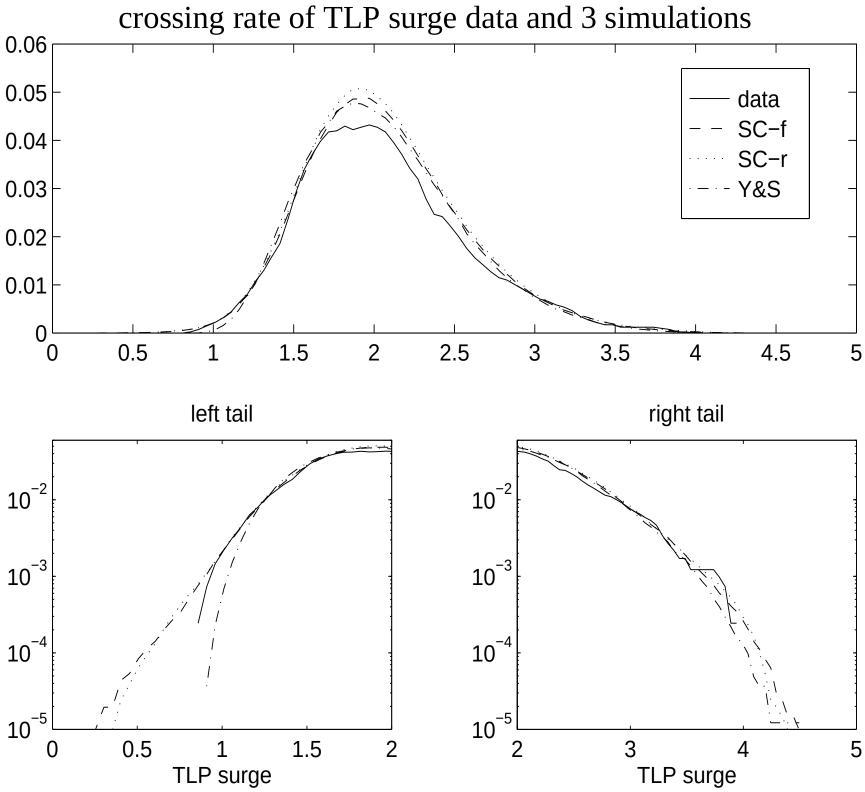


FIGURE 5.21 Crossing rate of TLP surge data and the average crossing rate of 50 realizations using three simulation methods.

used as the central algorithm for non-Gaussian conditional simulation techniques in chapter 7.

CHAPTER 6

MULTI-VARIATE SPECTRAL CORRECTION (MSC)

6.1 Introduction

The previous chapter presented the development of the spectral correction method for simulating single point (uni-variate) non-Gaussian processes with a wide range of power spectral densities and probability density functions. Spectral correction is here extended to include multi-point (multi-variate) simulation, where several non-Gaussian correlated time histories are generated simultaneously. Multi-variate simulation becomes necessary for cases where non-Gaussian processes at spatially distributed locations are desired. For example, simulation of multi-variate velocity and pressure fields. The multivariate spectral correction algorithm is referred to as MSC.

6.2 Overview

Two MSC methods are presented here. The first simulates processes which match the absolute value of the target coherence. This scheme is designed for cases where the coherence target is analytical and real valued, resulting in a target cross-spectral density matrix whose off diagonal terms are real. In this case corrections are iteratively made to the real valued design coherence function to achieve convergence within an error tolerance. This scheme is referred to as MSC I.

The second scheme is designed for cases where the cross-spectral density matrix has complex components. A correction to the absolute value of the design coherence will not properly account for the phase relationship. Iterative corrections are made to the design cross-correlation functions between each pair of variates. Proper matching of the cross-correlation function will account for both the magnitude and the phase of the cross-spectral density matrix. This scheme is intended for use when the target statistics are measured from experimental data, where the off diagonal terms in the measured target cross-spectrum will in almost all cases be complex. This scheme is referred to as MSC II.

6.3 MSC I - Matching Coherence

Coherence is a measure of linear correlation between two processes in terms of their auto-and cross-spectra

$$\gamma_{ij}^2 = \frac{(abs(G_{ij}))^2}{G_{ii}G_{jj}}. \tag{6.1}$$

where G_{ij} is the auto spectrum for $i = j$ and the cross spectrum otherwise. A reliable measure of coherence requires the ensemble averaging of numerous samples to represent the auto- and cross-spectra. Thus, the coherence cannot be corrected through direct manipulation of individual Fourier components without interfering with the desired non-Gaussian characteristics set in place by spectral correction.

It is possible to correct the coherence between processes without distorting the desired non-Gaussian transformation if this correction is made prior to the application of SC. The proposed method corrects the design coherence function used to generate the correlated parent Gaussian signals before application of SC.

The methodology is based on the simulation of Gaussian correlated processes using standard techniques, followed by transformation to the desired non-Gaussian processes using SC. The Gaussian simulation is based on the target spectral matrix and a design coherence. In some cases the coherence between the transformed non-Gaussian processes match the target coherence well enough to warrant no further corrections.

In other cases the resulting correlation is significantly distorted through the transformation. In order to correct this distortion, the Gaussian processes are re-simulated with a new design coherence. These are then re-mapped to non-Gaussian space. Typically, two or three iterations bring the coherence of the non-Gaussian processes into agreement with the target coherence.

6.3.1 Schematic: Development of MSC I

Figure 6.1 shows a schematic representation of the first multivariate spectral correction (MSC I) method. The iterative procedure is divided into 6 steps.

1) The target autospectra, G_{ii}^T, target moments, $m_{i,k}^T$, $k = 1...4$, and target coherence functions, γ_{ij}^T, describing the correlated $i = 1...num$ non-Gaussian process are input to the algorithm. The iteration counter is set to 1, and the design coherence for Gaussian simulation in the current iteration is set to the target coherence $\gamma^D{}_{ij}(it) = \gamma_{ij}^T$.

2) A set of Gaussian correlated processes x_G^i are generated using a standard multivariate

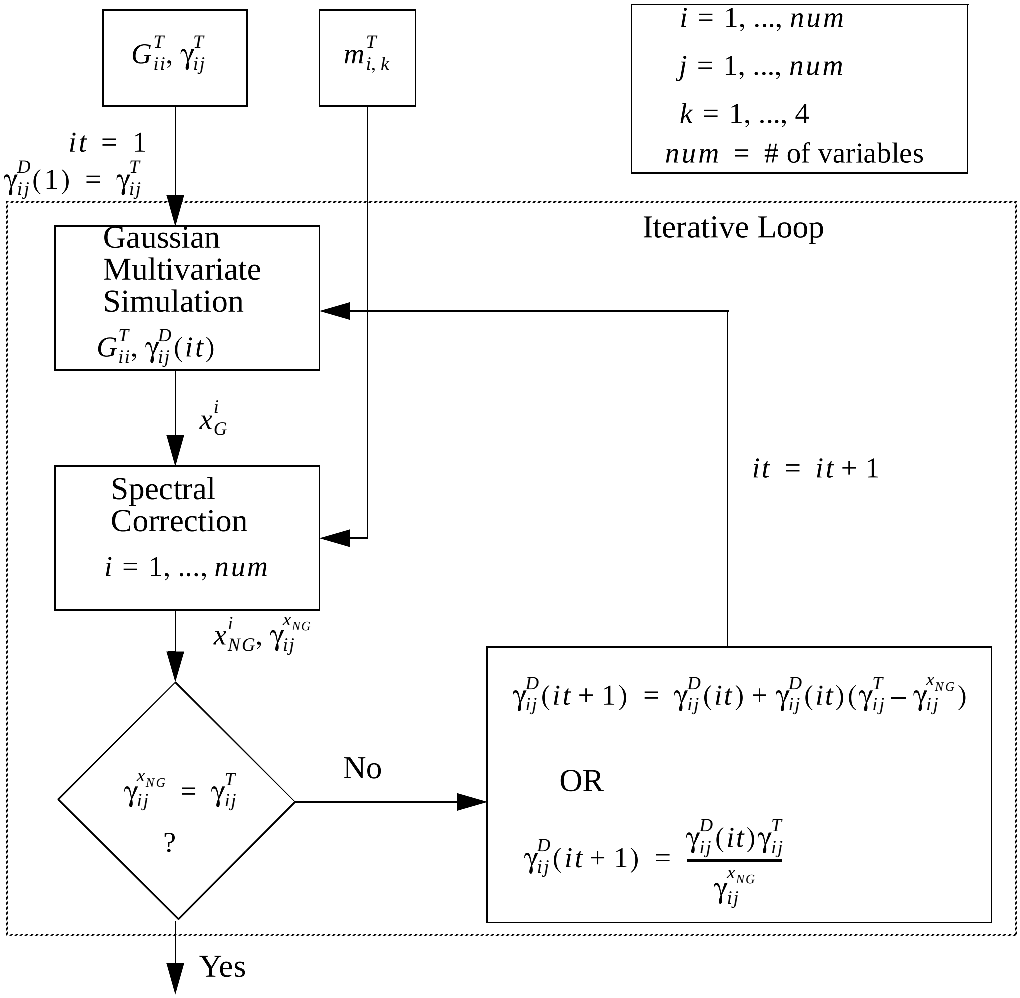


FIGURE 6.1 Schematic of the Multivariate Spectral Correction Simulation method MSC I.

Gaussian simulation algorithm. The target statistics are defined by the current iteration's design coherence $\gamma_{ij}^D(it)$, and the target autospectra G_{ii}^T.

3) The num correlated Gaussian processes x_G^i are sent to the spectral correction section. Here each realization is individually transformed to a non-Gaussian realization x_{NG}^i which matches its target spectrum, G_{ii}^T, and target moments $m_{i,k}^T$.

4) The coherence, $\gamma_{ij}^{x_{NG}}$, between the non-Gaussian realizations is measured and compared with the target coherence γ_{ij}^T to determine the error as some function of the two

$$err = f(\gamma_{ij}^T, \gamma_{ij}^{x_{NG}}) \ . \tag{6.2}$$

If the error is within acceptable limits, the algorithm ends, and x_{NG}^{i} are output as the final simulated realizations. If the error is unacceptable, the algorithm advances to the next stage. The specific error criteria will be discussed in a later section.

5) When the error between the target and measured coherence is not within the designated tolerance, the design coherence, C_{ij}^{D} used to generate the Gaussian processes is updated as a function of the current design coherence and the target and measured coherence

$$\gamma_{ij}^{D}(it+1) = f(\gamma_{ij}^{D}(it), \gamma_{ij}^{T}, \gamma_{ij}^{x_{NG}}(it)) \quad . \tag{6.3}$$

The update scheme is a hybrid of two functions, and is discussed in a later section.

6)The algorithm now returns to the top of the iteration loop with the new design coherence. Steps 2) through 5) are repeated until $\gamma_{ij}^{x_{NG}}$ is acceptable.

6.3.2 An Example: Simulated Pressure

Before proceeding to a more detailed discussion of the algorithm, an example is now shown to demonstrate the effectiveness of MSC I. The wind pressure on the edge of a building is simulated at two closely located positions. The locations lie within the separation zone where the pressure is non-Gaussian suction-type. A broad banded spectral model is used to represent the spectral contents of the pressure, and the coherence is modeled as an exponentially decaying function with parameters dependent on spatial separation. Based on measured full scale data, the target skewness and kurtosis are -.5 and 5.5, respectively.

Figure 6.2 shows the development of the coherence between the two non-Gaussian pressure simulations as the number of iterations progress from 1 through 20. It can be seen that in this case 3 iterations are enough to adequately match the target coherence. The lower right plot is the error measurement as a function of iterations. It can be seen that the error quickly drops within a few iterations.

Later examples will demonstrate the effectiveness of MSC I for simulating more than two processes. First, MSC II is developed and applied to the generation of samples based on measured rather than analytical autospectra and coherence models.

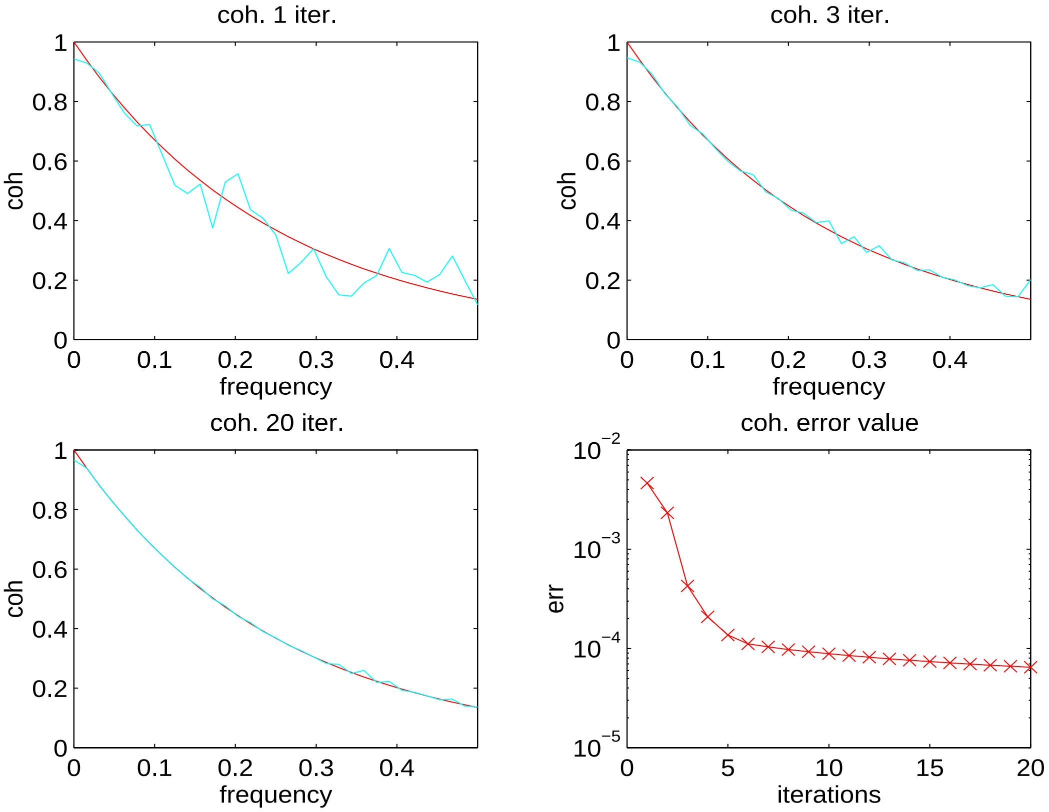


FIGURE 6.2 (top left): The target and measured coherence between two simulated pressure records after 0 iterations. (top right): The target and measured coherence after 3 iterations. (lower left): The target and measured coherence after 20 iterations. (bottom right): the error as a function of iterations on a semi-log scale.

6.4 MSC II - Matching Cross Correlation

The matching of the coherence as defined in Eq. 6.1 between locations i and j is equivalent to matching the absolute value of the target cross-spectral density $\left|G_{ij}^{T}\right|$. This is appropriate when real valued coherence models are used, and no particular phase relationship is assumed. However, when dealing with sets of correlated experimental data, it is often the case that the phase contains important information. For these circumstances the matching of the coherence function is insufficient for proper simulation, and MSC I algorithm in inappropriate.

Both the phase and magnitude information of the cross-spectral matrices are contained in the time domain equivalent cross-correlation function through inverse Fourier transform

$$R_{ij}^{T}(\tau) = \int_{-f_c}^{f_c} S_{ij}^{T}(f) e^{\sqrt{-1}(2\pi f)\tau} df \quad , \tag{6.4}$$

where $S_{ij}^{T}(f)$ is the two sided target cross-spectral density function. A pair of simulated processes with the same cross-correlation function as the target, $R_{ij}^{x_{NG}}(\tau) = R_{ij}^{T}(\tau)$, will properly reflect the phase and magnitude of the target.

The MSC II algorithm is of the same structure as MSC I, where now the iterative update is performed in the time domain on the design cross-correlation $R_{ij}^{D}(\tau)$. The algorithm proceeds exactly as outlined in the six steps in section 6.3.1, with the following substitutions

$$\gamma_{ij}^{T} = => R_{ij}^{T}, \quad \gamma_{ij}^{D} = => R_{ij}^{D}, \quad \gamma_{ij}^{x_{NG}} = => R_{ij}^{x_{NG}} \tag{6.5}$$

The schematic is shown in Fig. 6.3, where all coherence terms have been replaced with cross-correlation terms. Example applications of MSC II will follow a closer look at the three main components of the MSC algorithms.

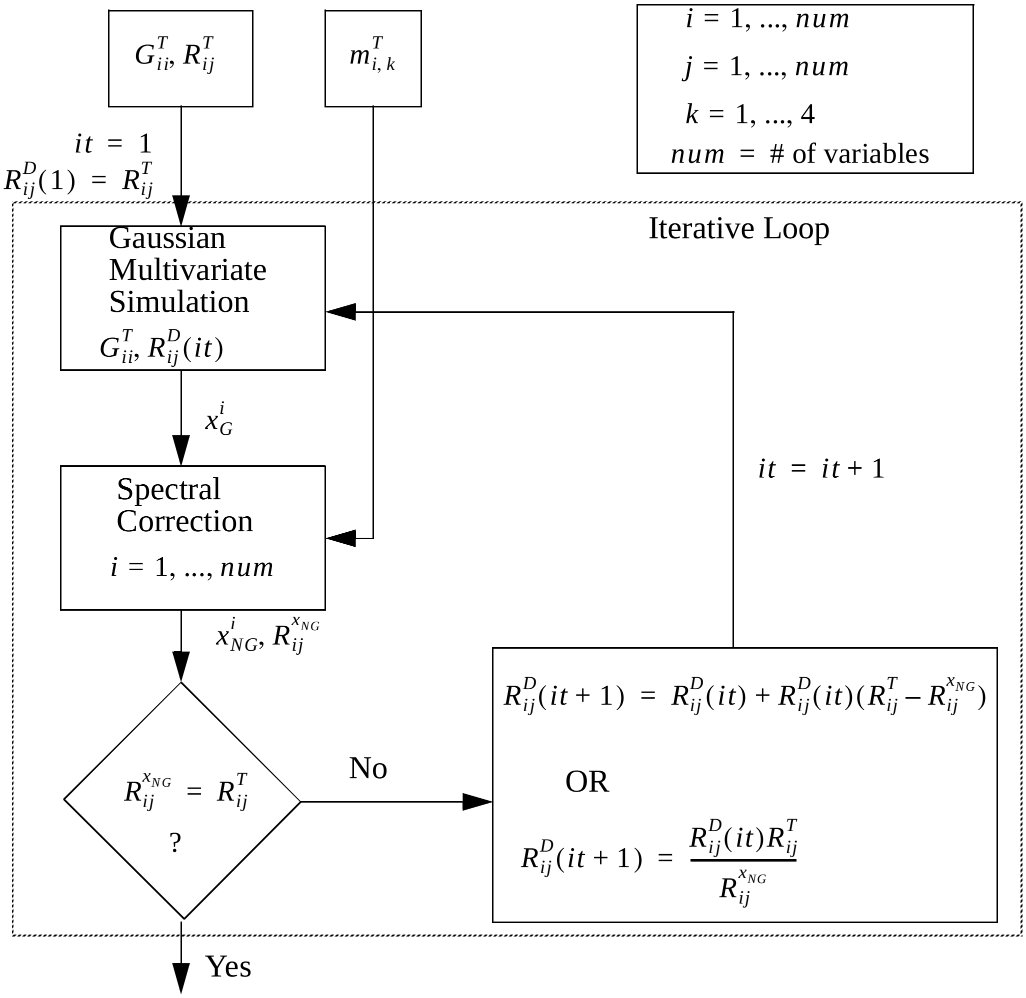


FIGURE 6.3 Schematic of the Multivariate Spectral Correction Simulation method MSC II.

6.5 The Three Main Components of the MSC algorithms

6.5.1 Gaussian Multivariate Simulation

The first part of the iterative algorithm supplies correlated Gaussian processes which match the design auto-spectra and the design coherence. The commonly used spectral decomposition method was introduced in the chapter concerning Gaussian simulation techniques, and is used in the MSC algorithms.

The major factor in this selection is the speed of the method, which is based on the efficient fast Fourier transform. The use of long realizations provides Gaussian simulations which well represent the design coherence [58].

6.5.2 Transformation to Non-Gaussian via Spectral Correction

It was shown in the previous chapter that SC is an effective tool for simulating non-Gaussian processes based on four moments and a target spectrum as input. Here it is converted to a transformation tool, where the inputs include the target moments and spectrum, and the Gaussian parent process which is to be transformed.

SC, when used for simulation, produces an initial random phase of uniform distribution to generate the simulation. For use as a transformation tool, the phase from the input parent process is used to seed the SC algorithm, and the magnitude of the parent process is replaced with the desired target spectrum.

Using SC, the $i = 1 \ldots num$ individual realizations of x_G^i are reliably transformed to non-Gaussian realizations x_{NG}^i with the desired autospectra G_{ii}^T and moments $m_{i,k}^T$. This transformation process distorts the coherence and cross-correlation associated with x_G^{ij}, and instigates the need for iteration in MSC if the distortion is significant.

6.5.3 Measuring, Comparing, and Updating the Coherence Function (MSC I)

6.5.3.1 Bias and Variance Error Effects

The coherence between the non-Gaussian processes are measured for comparison with the target coherence. These estimates contain some error due to the uncertainty in the auto and cross-spectral expectations in Eq. 6.1.

The number of segments, n, used in the ensemble average will determine the bias error

$$b[\hat{\phi}] = E[\hat{\phi}] - \phi, \tag{6.6}$$

where ϕ is the random variable, and $\hat{\phi}$ is the estimate or simulation of the random variable. Bendat and Piersol [6] suggest an estimate of bias error for measured coherence as a function of the number of segments n

$$b[\gamma_{ij}^2] = \frac{1}{n}(1-\gamma_{ij}^2)^2. \tag{6.7}$$

More significant bias error is associated with lower levels of coherence and fewer segments n.

In order to demonstrate the effect of bias error, three correlated Gaussian processes are simulated with their target coherence defined by sine waves. The coherence is measured between the three processes using both $n = 16$ and $n = 1024$ segments in the ensemble. The target coherence and Eq. 6.7 are used to estimate the coherence with the bias error added.

Figure 6.4 presents the target, simulated, and bias-added coherence functions between processes 1 and 2, and 1 and 3. The dashed line in the figure represents the target coherence, the solid line is the simulated coherence using 1024 segments in the ensemble average, the dashed-dot line is the measured coherence using 16 segments, and the dotted line is the target coherence added to the predicted bias error using Eq. 6.7. The distortion in the measured coherence is most pronounced for the 16 segment case in the low frequency region.

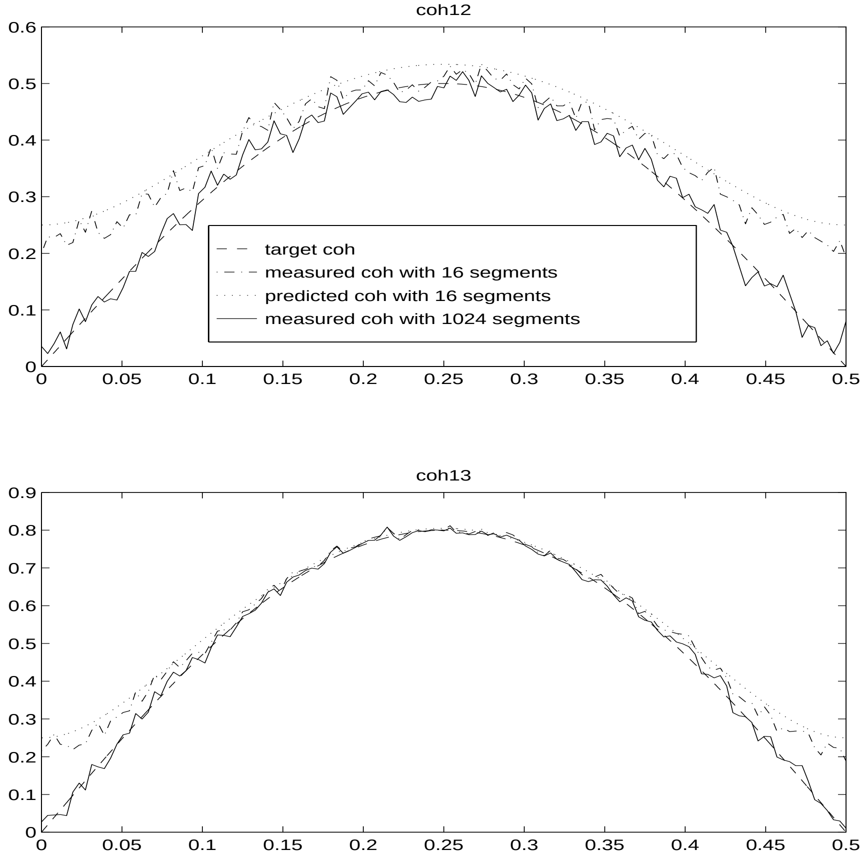


FIGURE 6.4 Effects of bias error on the measured coherence using 16 and 1024 segments in the ensemble estimation.

Eq. 6.7 provides a good estimate of the bias error due to insufficient ensembles in the average. However, removing the bias in the measured coherence using Eq. 6.7 is not a viable solution unless the exact coherence of the process is know apriori. This is not the case in the MSC algorithm. In order to remove any significant bias error, the rule of thumb applied to this study is to use at least 100 segments whenever possible to reduce maximum error in measured γ_{ij} to 0.1.

Variance error is defined as the variance of the measured process about its expected value. This is a result of the random nature of measured data, and is often called random error. An approximate expression for variance error associated with measured coherence is given in [6] as

$$var[\gamma_{ij}^2] = \frac{2\gamma_{ij}^2(1-\gamma_{ij}^2)^2}{n} . \tag{6.8}$$

More segments reduce variance error just as in the case of bias error.

6.5.3.2 Error Criterion for Comparison of Target with Resulting Coherence

The termination of the iterative algorithm MSC I is dependent on the error between the target coherence and the coherence of the simulated processes x_{NG}^i. Two error criteria are used to account for both bias and variance error. The expected bias and variance error are determined at each frequency by Eqs. 6.7 and 6.8 using the target coherence and the number of samples, n, to be used to measure the coherence of the simulation

$$b[(\gamma_{ij}^T(f))^2] = \frac{1}{n}(1-(\gamma_{ij}^T(f))^2)^2 \quad , \tag{6.9}$$

$$var[(\gamma_{ij}^T(f))^2] = \frac{2(\gamma_{ij}^T(f))^2(1-(\gamma_{ij}^T(f))^2)^2}{n} \quad . \tag{6.10}$$

The first error criterion is that the simulation coherence must lie within the bounds of the expected variance at every frequency,

$$err_v = 0 \quad \text{if} \quad (\gamma_{ij}^{x_{NG}}(f) - \gamma_{ij}^T(f))^2 < var[(\gamma_{ij}^T(f))] \qquad \text{for all } f, \tag{6.11}$$

$$err_v = 1 \text{ otherwise,} \tag{6.12}$$

where $err_v = 1$ indicates the iterations must continue.

The second criterion is that the mean error between target and simulation coherence over all frequencies must lie within the mean bias error

$$err_b = 0 \text{ if } \quad \frac{1}{nf}\sum_{f=1}^{nf}\left|\gamma_{ij}^{x_{NG}}(f) - \gamma_{ij}^{T}(f)\right| \; < \; \frac{1}{nf}\sum_{f=1}^{nf} b[(\gamma_{ij}^{T}(f))] \quad , \tag{6.13}$$

$$err_b = 1 \text{ otherwise,} \tag{6.14}$$

where nf is the number of discrete frequencies in the coherence function, and $err_b = 1$ indicates that the iterations must continue.

6.5.3.3 Updating the Design Coherence Function

Before the algorithm begins a new iteration, the design coherence used to generate the Gaussian processes is updated as a function of the previous iteration's design coherence, and the target and measured coherence functions. Two updating schemes are used in the MSC algorithm. The first is adapted from Yamazaki and Shinozuka's method for non-Gaussian simulation, [101], discussed in section 2.8.1. Equation 6.3 from the development section is

$$\gamma_{ij}^{D}(it+1) = (\gamma_{ij}^{D}(it)C_{ij}^{T})/(\gamma_{ij}^{x_{NG}}(it)) \; . \tag{6.15}$$

The second method is adapted from Elishakoff's method for conditional non-Gaussian simulation [18], discussed in section 2.10. This update is given by

$$\gamma_{ij}^{D}(it+1) = \gamma_{ij}^{D}(it) + \gamma_{ij}^{D}(it)(\gamma_{ij}^{T} - \gamma_{ij}^{x_{NG}}(it)) \; . \tag{6.16}$$

When the measured coherence exceeds the target coherence at a given frequency, $\gamma_{ij}^{x_{ng}} > \gamma_{ij}^{T}$, both Eq. 6.15 and Eq. 6.16 will reduce the design coherence, thus reducing the measured coherence in the next iteration. Conversely, the design coherence is increased if $\gamma_{ij}^{x_{ng}} < \gamma_{ij}^{T}$ to increase $\gamma_{ij}^{x_{ng}}$ in the next iteration.

An example is used to demonstrate the correction to the design coherence. Starting at the first iteration, the target and design coherence are set to 0.5 and denoted Ct and Cd. We assume that after the first non-Gaussian transformation the resulting measured coherence is as in the top plot of Fig. 6.5, and denoted Cx. The second iteration will begin with the same target spectrum, while the design spectrum is corrected by either Eq. 6.15 or 6.16, shown in the bottom plot.

The bottom plot in Fig. 6.5 demonstrates that Eq. 6.15 produces a larger correction than does Eq. 6.16. Equation 6.15 is capable of significantly reducing the coherence error within a few iterations, but under certain conditions this has the undesired effect of over-compensating for the coherence error, and results in oscillation about the target coherence without convergence. This has been observed only when the error has been reduced through several previous iterations, typically when three or more variates are being simulated. The smaller corrections using Eq. 6.16 virtually eliminate this oscillating behavior at the expense of more iterations to reduce the coherence error. Figure 6.6 demonstrates the rate of coherence error reduction using Eqs. 6.15 and 6.16. Equation 6.15 is observed to more quickly converge. Later examples will show the oscillating behavior.

The MSC algorithm uses both correction schemes to optimize speed while preventing oscillations in convergence. The first several iterations make use of Eq. 6.15, which reduces coherence error quickly, and any further iterations use Eq. 6.16 to fine tune the design coherence γ_{ij}^{D} without over-compensating.

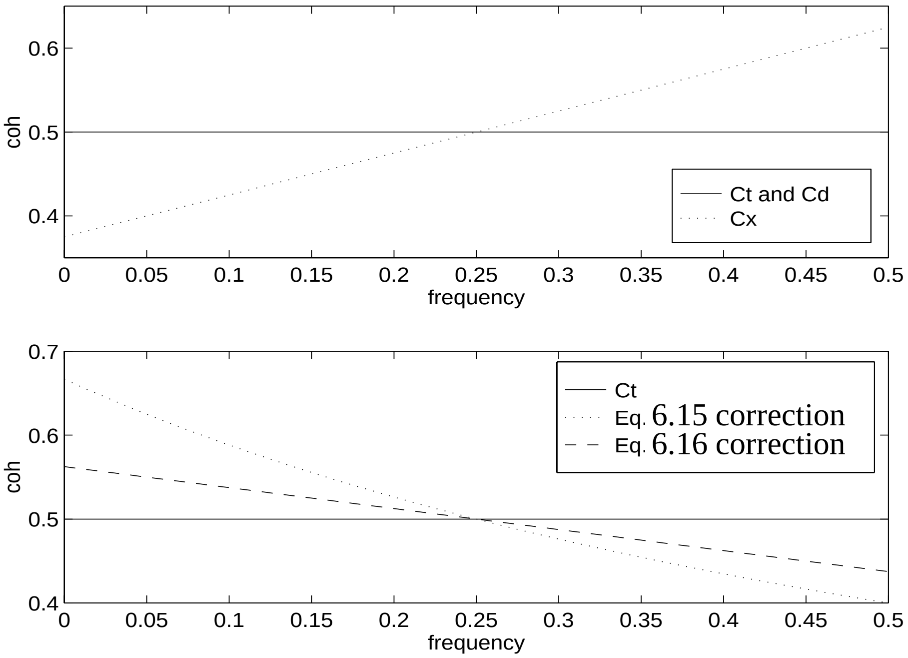


FIGURE 6.5 (top plot): An example of a target coherence and the resulting measured coherence. (bottom plot): The target coherence, and the adjusted design coherence using Eqs. 6.15 and 6.16.

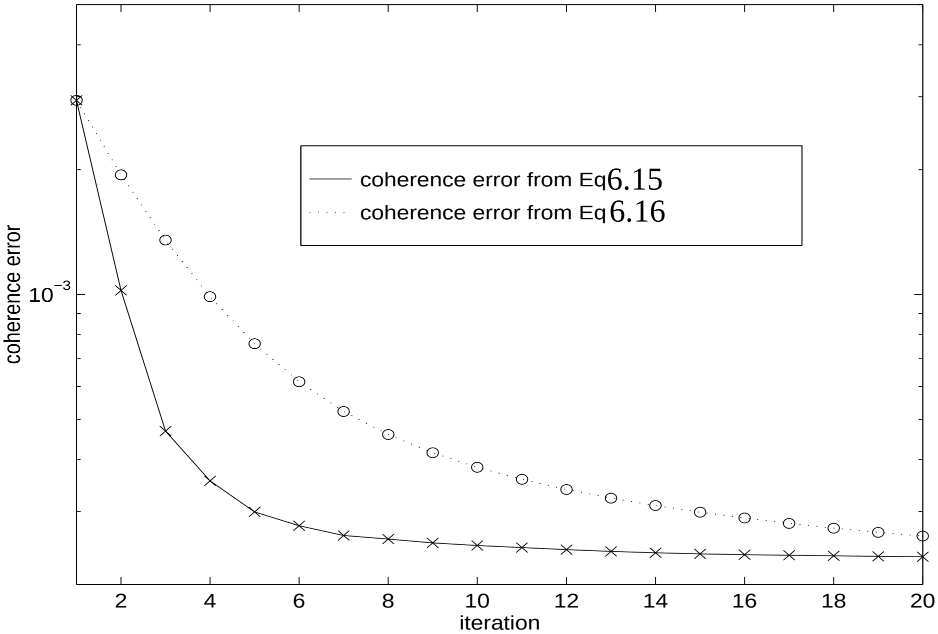


FIGURE 6.6 Coherence error reduction rate using Eq. 6.15 and Eq. 6.16.

6.5.4 Measuring, Comparing, and Updating the Cross-Correlation Function

6.5.4.1 Error Criterion for Comparison of Target and Simulated Coherence

The termination of the iterative algorithm MSC II is dependent on the error between the target cross-correlation, R_{ij}^{T}, and the cross-correlation, $R_{ij}^{x_{NG}}$ of the simulated processes x_{NG}^{i}. In MSC I, the error is measured on the coherence function, which is in the range $0 < C_{ij} < 1$. In order to quantify error for the cross-correlation function, the mean squared error and mean error are measure using the target and cross-correlations normalized by the largest value in the target cross-correlation,

$$\rho_{ij}^{T}(\tau) = (R_{ij}^{T}(\tau))/(max(\left|R_{ij}^{T}(\tau)\right|)), \qquad (6.17)$$

$$\rho_{ij}^{x_{NG}}(\tau) = (R_{ij}^{x_{NG}}(\tau))/(max(\left|R_{ij}^{T}(\tau)\right|)). \qquad (6.18)$$

The MSE error criterion is then

$$err_{v} = 0 \ \text{ if } \ (\rho_{ij}^{x_{NG}}(\tau) - \rho_{ij}^{T}(\tau))^{2} < var[(\rho_{ij}^{T}(\tau))] \ \text{ for all } \tau \qquad (6.19)$$

$$err_{v} = 1 \ \text{ otherwise}, \qquad (6.20)$$

where the variance error for cross correlation measured as the inverse Fourier transform of the cross spectrum is defined in [6] as

$$var[\rho_{ij}^{T}(\tau)] = \frac{1 + (\rho_{ij}^{T}(\tau))^{-2}}{Nn}. \qquad (6.21)$$

In the above, n is the number of segments used in the spectral estimate, and N is the number of data points used in each segment.

There is no inherent bias error associated with the cross-correlation measurement. However, the mean error between the target and simulation cross-correlation does steadily decrease as the fit becomes more accurate. The mean error criterion is set to

$$err_b = 0 \text{ if } \frac{1}{2N}\sum_{\tau=-N}^{N}\left|\rho_{ij}^{x_{NG}}(\tau)-\rho_{ij}^{T}(\tau)\right| < threshold\,, \tag{6.22}$$

$$err_b = 1 \text{ otherwise,} \tag{6.23}$$

where $threshold$ is typically set to 0.0001.

6.5.4.2 Updating the Design Cross-Correlation Function

Before the algorithm begins a new iteration, the design cross-correlation used to generate the Gaussian processes is updated as a function of the target-, simulated, and the previous iteration's design cross-correlation functions. The updating scheme used in the MSC II algorithm is

$$R_{ij}^{D}(it+1) = \left(\rho_{ij}^{D}(it)+\frac{\left|\rho_{ij}^{D}(it)\right|[\rho_{ij}^{T}-\rho_{ij}^{x_{NG}}(it)]}{max(\left|\rho_{ij}^{D}(it)\right|)}\right)(max(\left|R_{ij}^{T}\right|)) \tag{6.24}$$

where τ is removed for convenience, but implied, and

$$\rho_{ij}^{D}(\tau) = (R_{ij}^{D}(\tau))/(max(\left|R_{ij}^{T}(\tau)\right|))\,. \tag{6.25}$$

The updating scheme in Eq. 6.24 allows larger adjustments of the design cross-correlation at larger values of $\rho_{ij}^{D}(\tau)$, and was found to be the most appropriate of the various schemes tried.

6.6 Example Applications of MSC I

This section will present two examples of the simulation of multi-variate non-Gaussian processes using MSC I. A significant difference between the target and final design coherence is shown in the first example to demonstrate the necessity of iteration. A simulation of ten correlated processes is presented to show the convergence of the MSC I algorithm is not restricted to a few variates.

6.6.1 Sine Wave Coherence Functions

This example shows the contrast between the final design coherence, γ_{ij}^{D}, and the target coherence, γ_{ij}^{T}. Three non-Gaussian processes are simulated using a broad banded target spectrum describing wind turbulence, and target skewness and kurtosis values of -0.1 and 5.5, respectively. The three target coherence functions are

$$\begin{aligned} \gamma_{12}^{T} &= 0.5(\sin(\omega)) \\ \gamma_{13}^{T} &= 0.8(\sin(\omega)), \\ \gamma_{23}^{T} &= 0.7(\sin(\omega)) \end{aligned} \tag{6.26}$$

where $\omega = 0\ldots\pi$ in 129 discrete increments. The processes consist of 16384 data points, and are divided into 128 segments of 128 data points each for the ensemble measurement of coherence.

Figure 6.7 compares the target, final design, and final simulated non-Gaussian coherence functions, $\gamma_{ij}^{T}, \gamma_{ij}^{D}, \gamma_{ij}^{x_{NG}}$, and the coherence error for each case with respect to iterations. First it is noted that the design coherence has a positive mean offset from the target in each case. This indicates that the measured non-Gaussian coherence after the first iteration had a negative mean offset w.r.t. the target. The bottom plot of the coherence error with iteration demonstrates the beginning of the oscillating behavior (noted in section 6.5.3.3) at the 5th iteration. At the fifth iteration the design coherence correction in Eq. 6.15 is replaced with Eq. 6.16 and MSC achieves a steady state coherence error.

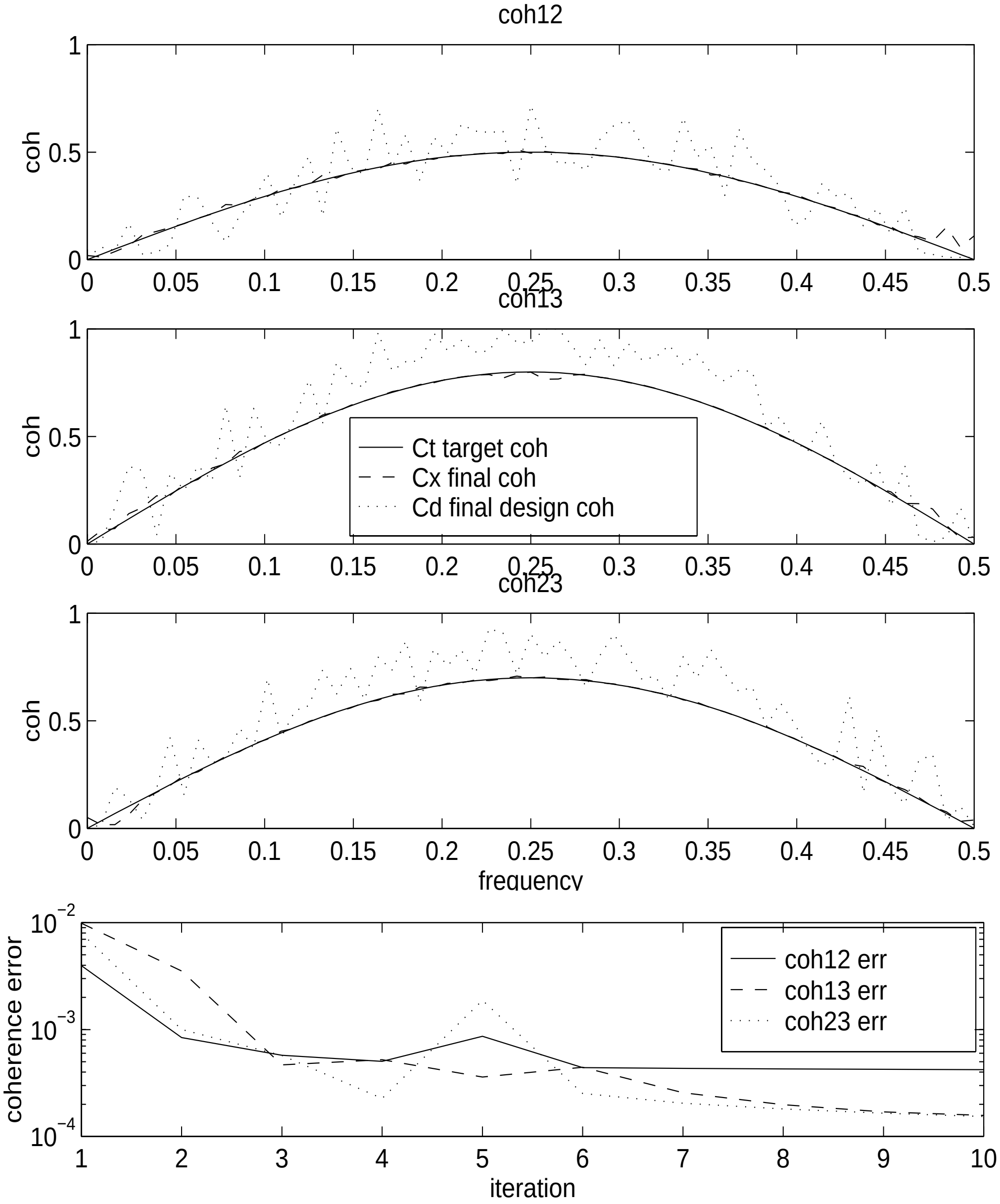


FIGURE 6.7 Example from section 6.6.1: Target, design, and measured coherence functions after 10 iterations, and the coherence error.

6.6.2 Example: 10 Correlated Wind Pressure Processes Using MSC I

This example simulates ten correlated processes representing wind velocity at ten vertically separated locations in a separated reverse flow region. A broad banded wind spectrum is used to define the target auto-spectra, and the target coherence function is

$$\gamma_{ij}^{T} = \exp\left(-f \times \frac{p_1}{p_2}\right) , \tag{6.27}$$

$$p_1 = [\mathrm{ch}(h_i - h_j)^2 + \mathrm{cv}(v_i - v_j)^2]^{1/2} , \quad p_2 = \frac{\bar{U}_i + \bar{U}_j}{2} , \tag{6.28}$$

where f is the frequency, h_i is the horizontal location of point i, v_i is the vertical location of i, ch and cv are constants with values of 256 and 100, and U_i is the mean wind velocity at location i. h_i is assigned a constant value for all locations, and $v_i = 40 + 3i$ meters. The skewness and kurtosis values of -1.0 and 6.5 are assigned to each process.

Forty-five coherence functions are necessary to describe the relationship between all possible pairs of the ten processes. Figure 6.8 is a view of the coherence between the closest two locations, the most distant, and two in between. Each plot shows the target and resulting measured coherence, delineated as Ct and Cx, where the plot title ‘cohij’ indicated the coherence between the processes at the ith and jth locations. The inability to match the very low level coherence in the bottom two plots is a result of the inherent bias error, which, from Eq. 6.7, is $b[\gamma_{ij}] = \sqrt{\frac{1}{n}(1-\gamma_{ij}^2)^2} = \sqrt{1/128} = 0.0884$.

Figure 6.9 presents the coherence error with respect to iterations. The top plot is the error between the coherence functions in Fig. 6.8. The lowest error corresponds to locations close together (1 and 2), where the target coherence is of significant value over the frequency range up to the cutoff frequency. The lower plot demonstrates the oscillatory

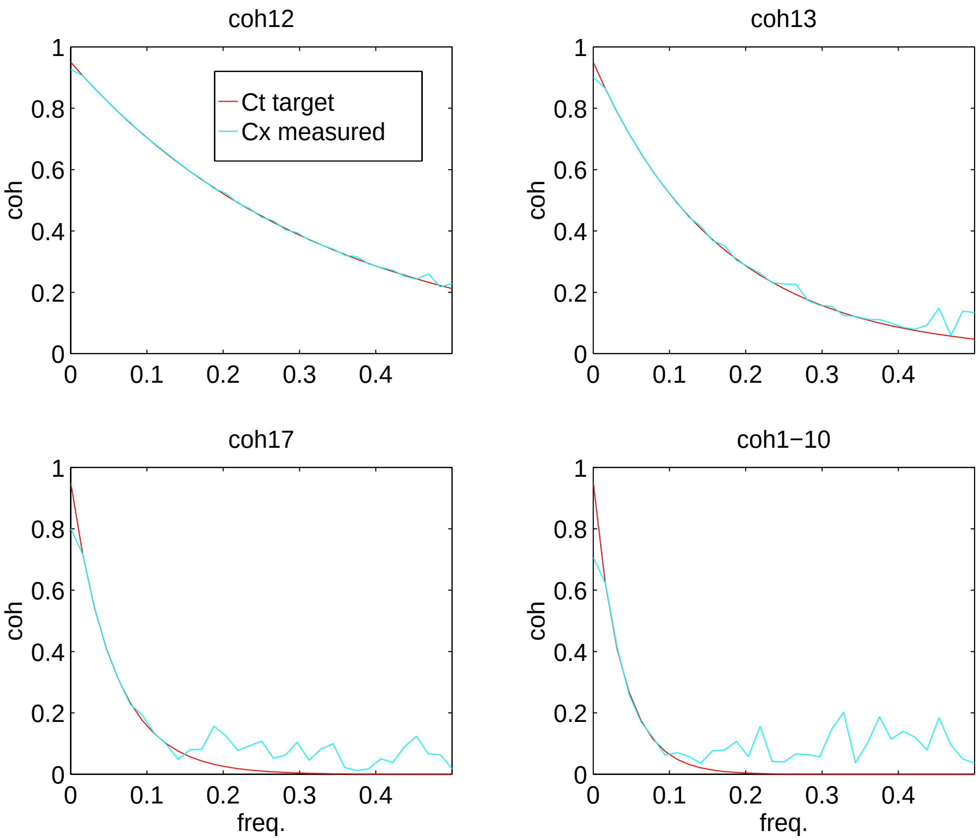


FIGURE 6.8 Section 6.6.2: Selected target coherence (Ct) and measured coherence (Cx) from ten correlated wind velocity simulations.

behavior of the coherence error for the closely spaced locations 2 and 3. When the MSC algorithm switches update schemes from Eq. 6.15 to Eq. 6.16, the coherence error continues to converge to an acceptable value. Figure 6.10 is a portion of four of the ten records. The retention of low frequency correlation in the non-Gaussian characteristics is easily observed.

6.7 Simulation of Measured Data (MSC II)

Examples are now considered where the phase of the target cross-spectrum is also considered. MSC II is used for simulations based on target inputs from measured data.

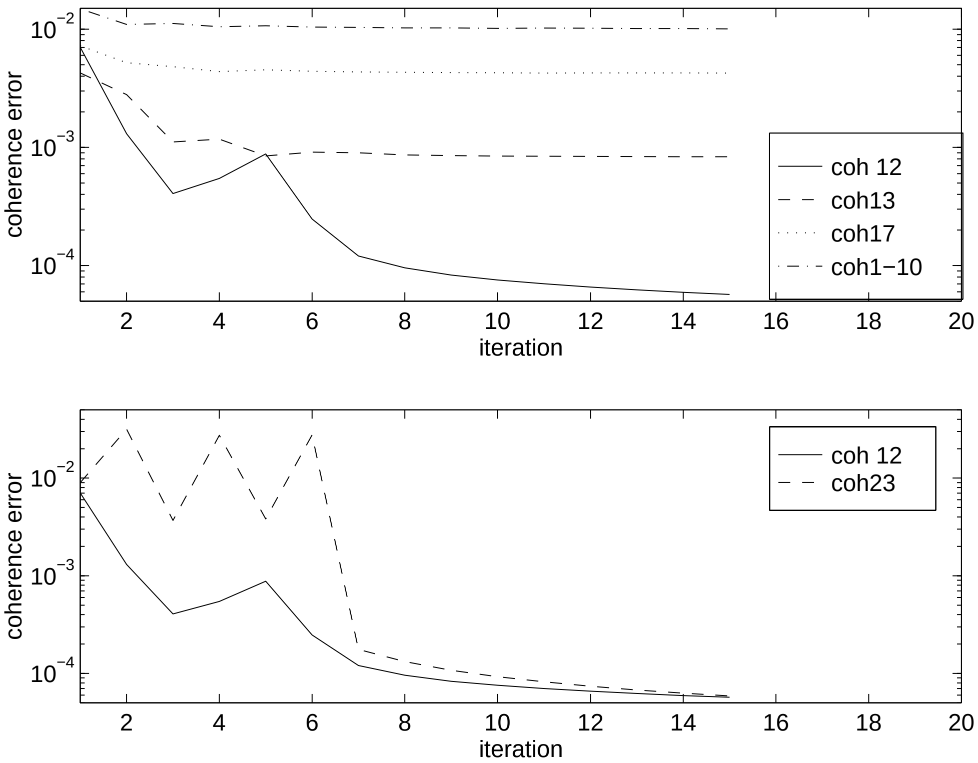


FIGURE 6.9 Section 6.6.2: Error between Ct and Cx in Fig. 6.8 (top). Demonstration of oscillating coherence error (bottom).

6.7.1 Measured Full Scale Wind Pressure Data

The first example uses full scale pressure data measured simultaneously at several locations on the roof of an instrumented building at Texas Tech University (TTU). Here we consider three locations on the roof in a separation zone where the pressure is highly non-Gaussian. Figure 6.11 is a diagram of the approximate locations of the pressure taps. Their locations are slightly offset from each other both parallel and perpendicular to the incoming wind. The offset parallel with incoming wind leads to a non-zero phase relationship between locations, as the turbulent wind field arrives at each location at slightly dif-

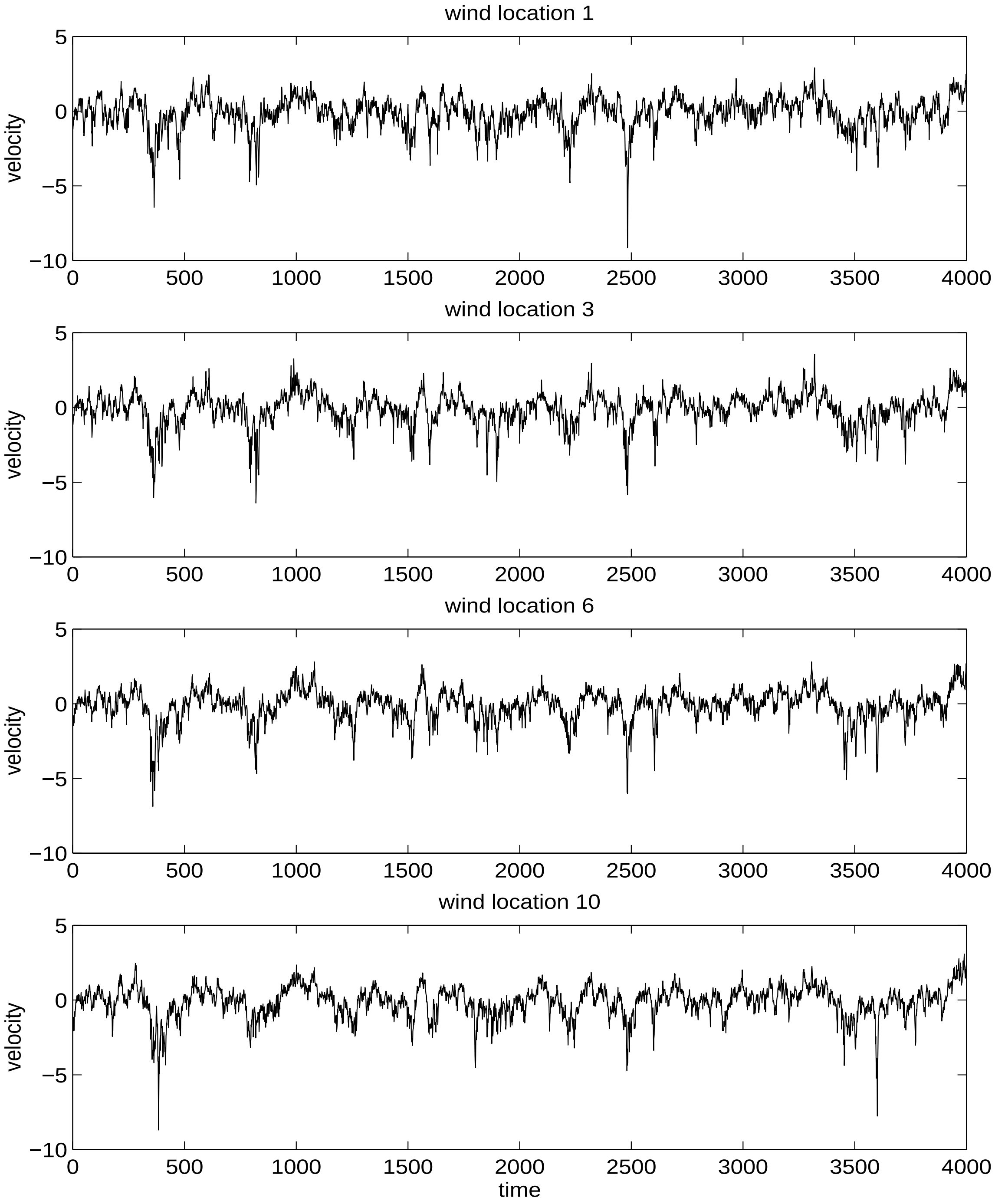


FIGURE 6.10 Section 6.6.2: Four records from a set of ten correlated wind velocity simulations.

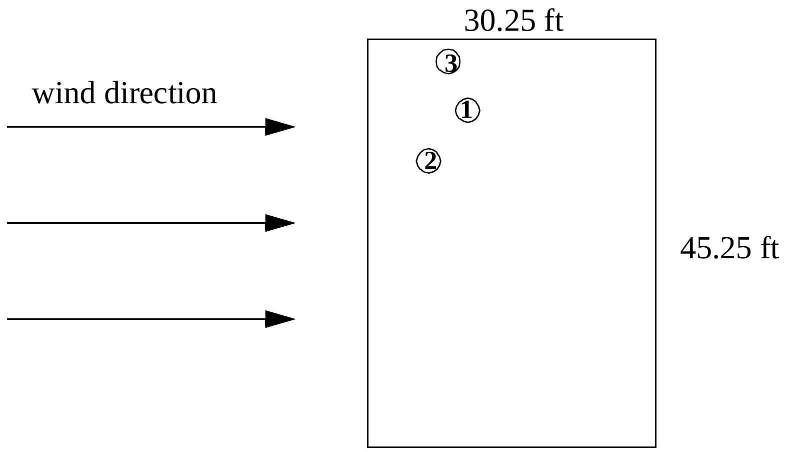


FIGURE 6.11 Section 6.7.1: Location of pressure taps on Texas Tech building.

ferent times. This was not the case in the previous examples, where the locations had no offset parallel with the incoming wind.

The statistics of the measured records used to define the target inputs for this example are listed in Table 6.1. Using Spectral correction as the non-Gaussian transformation technique, the moments of the resulting simulated pressure records are within 5% of their targets in Table 6.1.

TABLE 6.1 STATISTICS OF THE 3 TAPS IN FIG. 6.11

	mean	std	skewness	kurtosis
tap 1	-0.7131	0.2862	-0.9584	5.2566
tap 2	-0.7112	0.2944	-1.1076	5.6147
tap 3	-0.8994	0.3393	-0.8049	4.1397

The goal of this example is to generate realizations of three pressure records which maintain the complex coherence between the measured records, while properly reflecting the higher order statistics and spectral contents. Figures 6.12-6.14 compare the three target and simulated cross-spectral functions in the significant frequency range, and the target and simulated cross-correlations. The match between target and simulated cross-spectra are excellent in the low frequency range where there is significant energy. The cross-correlation errors are shown in Fig. 6.15 as a function of the 20 iterations used.

Figures 6.16 and 6.17 show the time history of the three full scale measured pressure records, and their simulations using MSC II. The low frequency correlation can be seen in both figures, and the non-Gaussian characteristics of the measured and simulated records look similar. To quantify the similarity of the non-Gaussian characteristics, linear and semi-log views of the measured and simulated histograms are shown in Fig. 6.18, and show excellent compatibility between measured and simulated data. This is expected, as

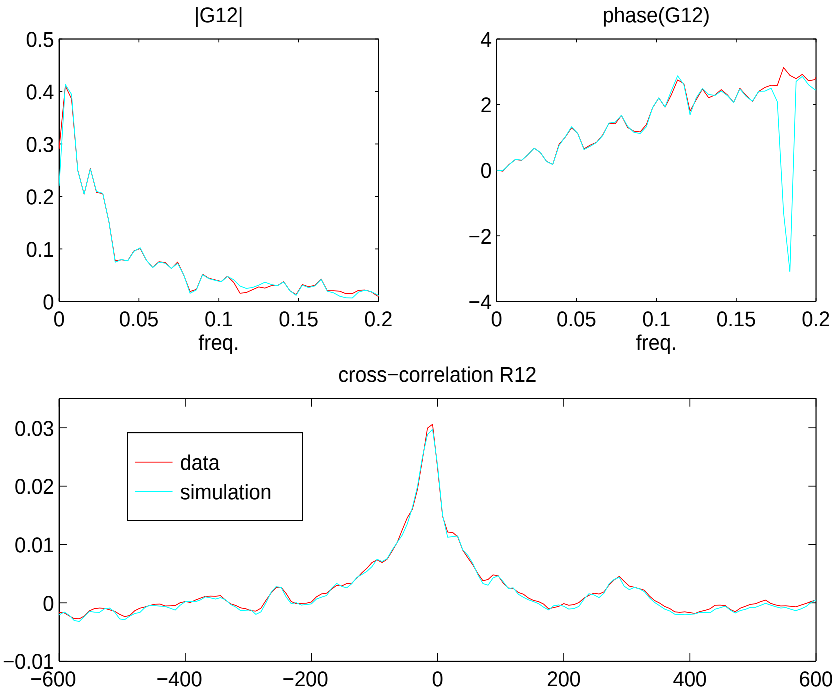


FIGURE 6.12 Section 6.7.1: Target and simulated statistics between wind pressure records 1 and 2. Top left: Absolute value of the cross-spectrum, Top right: phase of the cross-spectrum, Bottom: cross-correlation function.

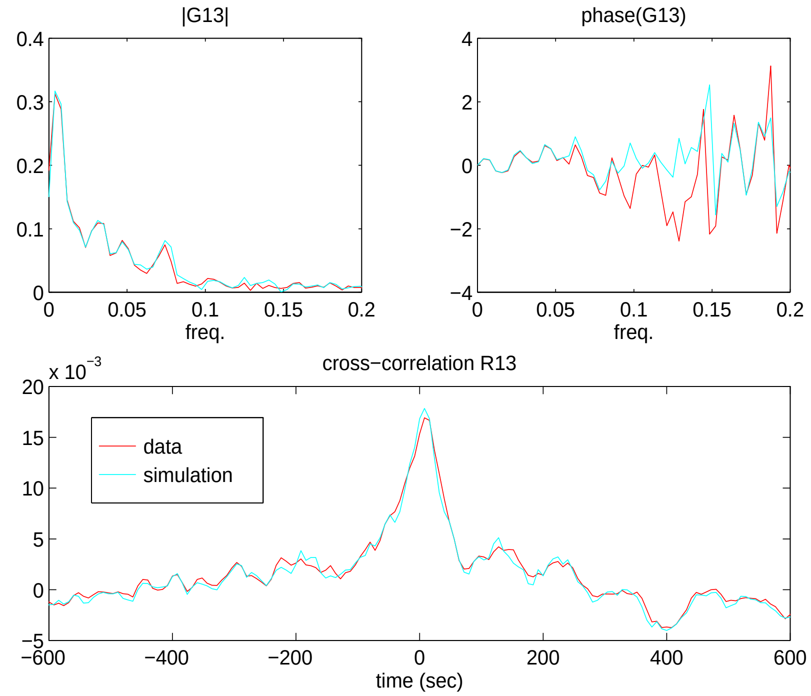


FIGURE 6.13 Section 6.7.1: Target and simulated statistics between wind pressure records 1 and 3. Top left: Absolute value of the cross-spectrum, Top right: phase of the cross-spectrum, Bottom: cross-correlation function

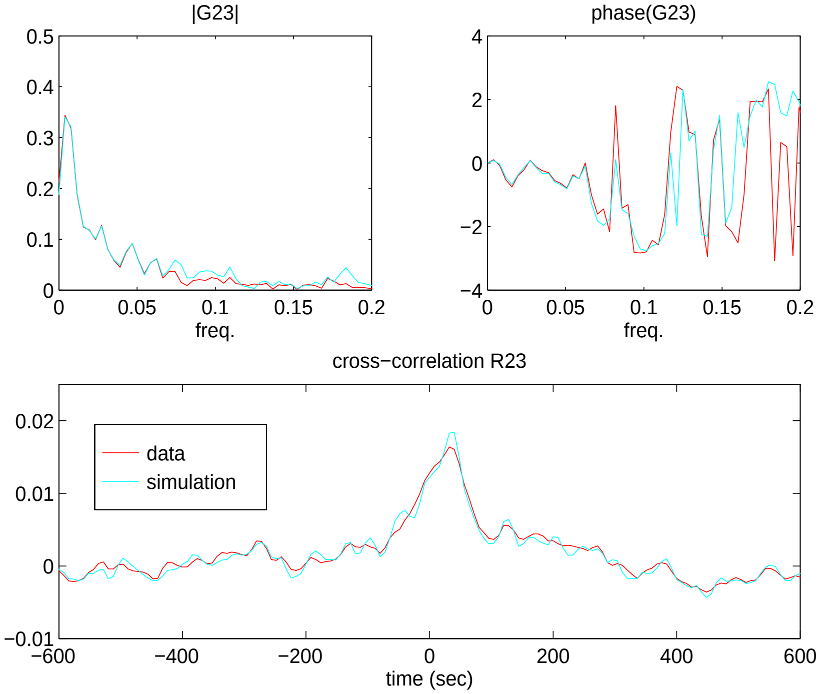


FIGURE 6.14 Section 6.7.1: Target and simulated statistics between wind pressure records 2 and 3. Top left: Absolute value of the cross-spectrum, Top right: phase of the cross-spectrum, Bottom: cross-correlation function.

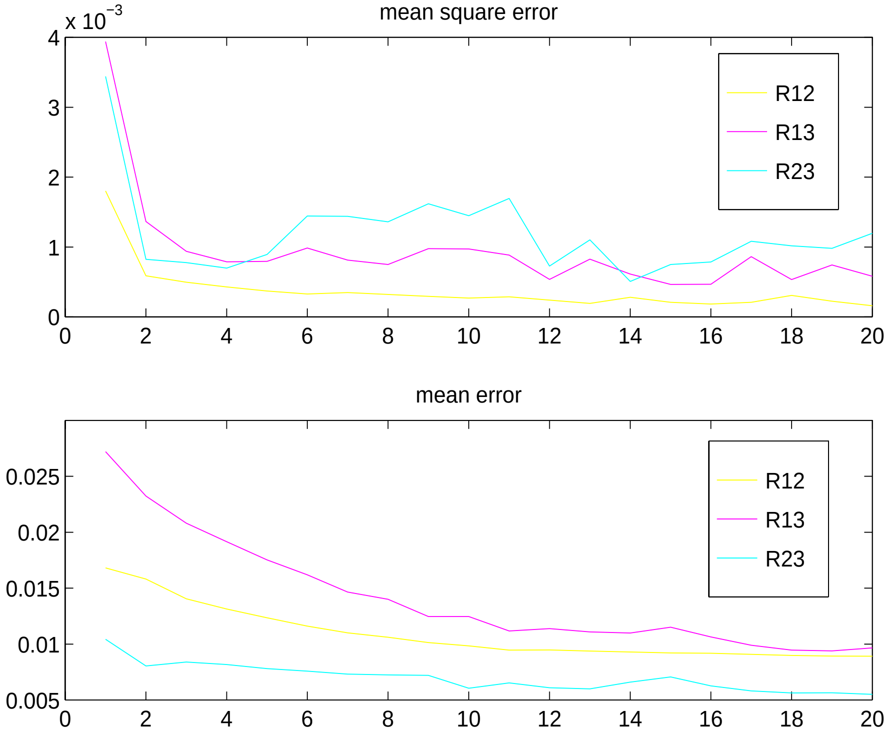


FIGURE 6.15 Section 6.7.1: Normalized mean and mean square error between RT and RX in Figs. 6.12-6.14.

the first four moments of each realization are within 5% of their corresponding measured sample through the error criteria specified in the spectral correction algorithm.

6.7.2 Measured Offshore Tension Leg Platform Response

This last example utilizes experimental data from extensive tests performed on the response of a scale model tension leg platform under simultaneous random wind and wave input. The measured data records under consideration describe the wave height elevation as it impinges upon the structure, and the horizontal response of the platform parallel with the wave direction (surge response). These are referred to in this example as records 1 and 2, respectively. The tests conducted were intended to represent extreme sea conditions, resulting in non-Gaussian input waves. The TLP system stiffness is nonlinear, and is sub-

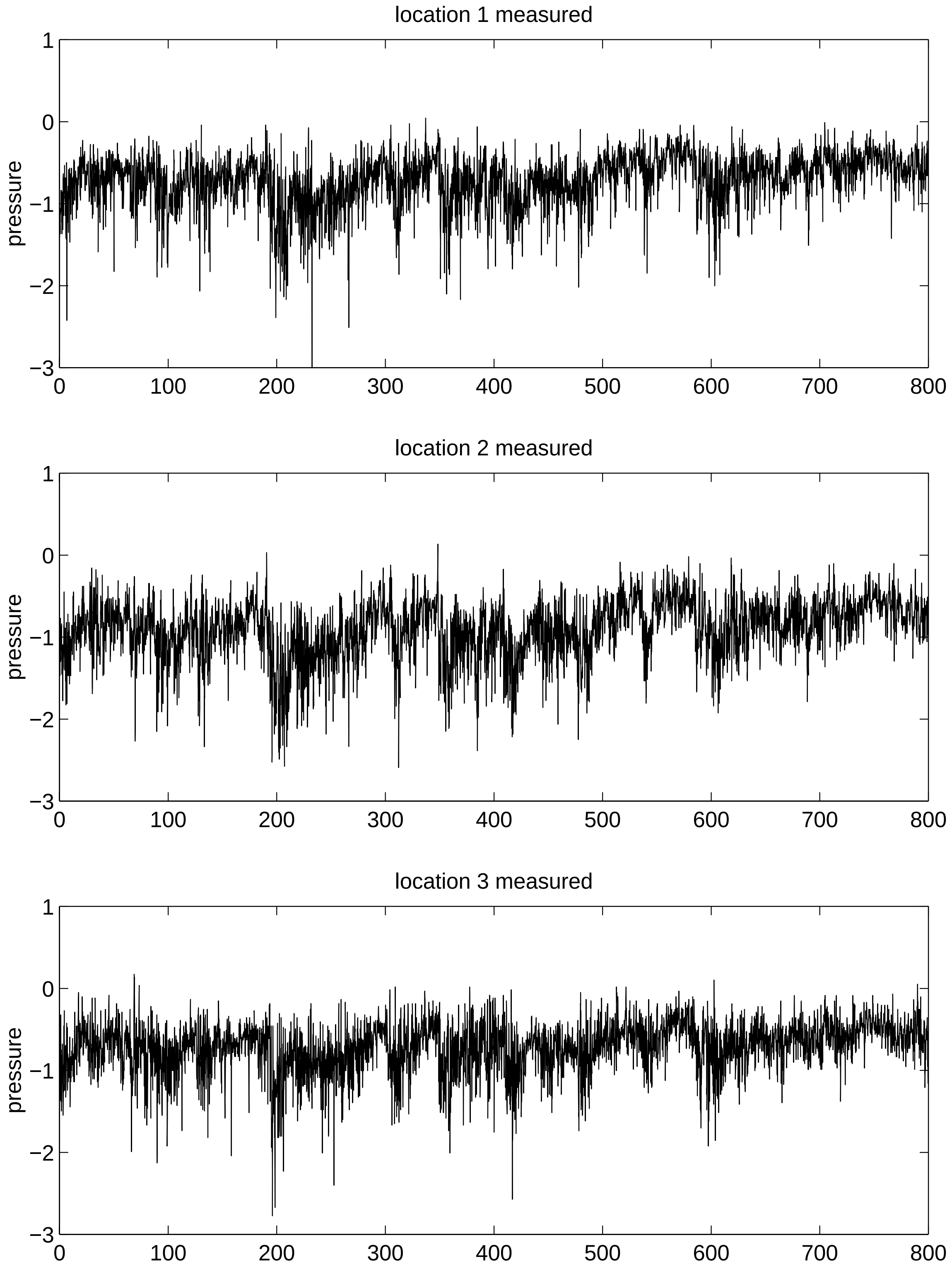


FIGURE 6.16 Section 6.7.1: Measured rooftop pressure at the three locations in Fig. 6.11.

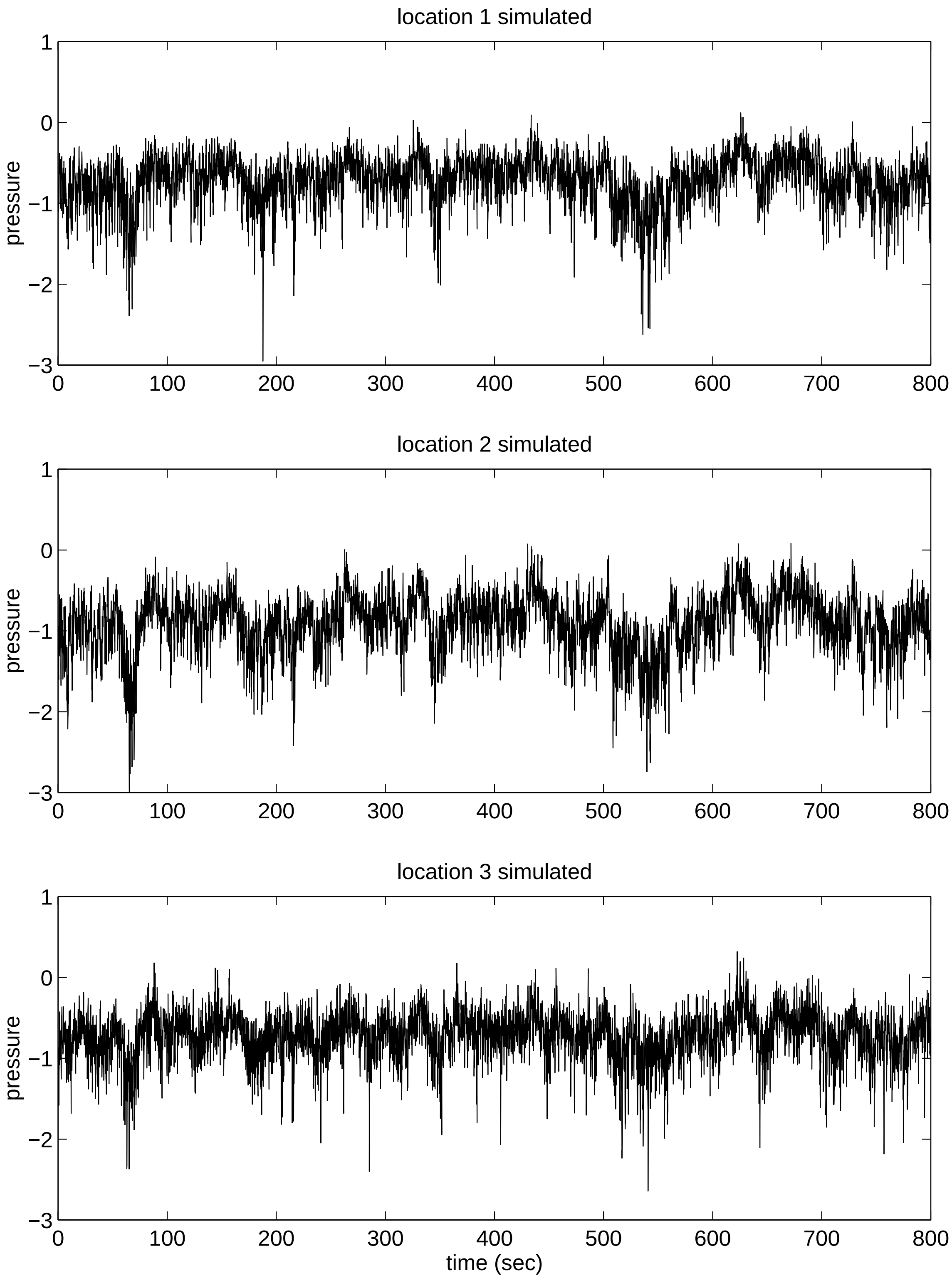


FIGURE 6.17 Section 6.7.1: Simulated rooftop pressure at the three locations in Fig. 6.11.

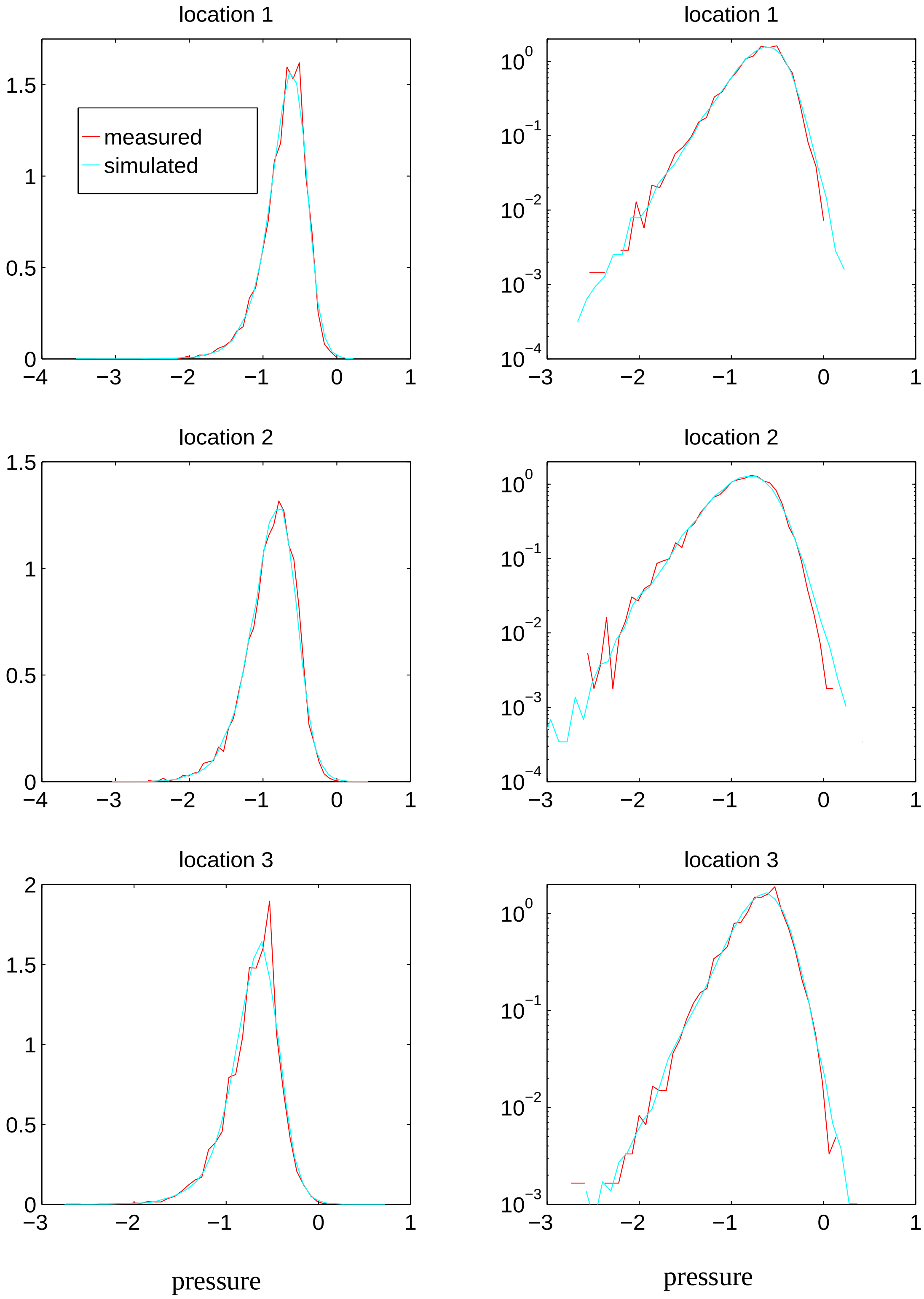


FIGURE 6.18 Section 6.7.1: PDF histograms of measured and simulated pressure data at the 3 rooftop locations in Fig. 6.11 using linear and semi-log scales.

jected to non-Gaussian input, leading to a non-Gaussian surge response. The statistics of the two records used in this example are reported in Table 6.2, and the moments of the simulations are within 5% of the targets using spectral correction.

TABLE 6.2 STATISTICS FOR TLP MEASURED DATA

	mean	std	skewness	kurtosis
wave elevation	0.003	0.193	0.227	3.604
surge response	0.208	0.2742	0.6021	3.529

Figure 6.19 shows the target and simulated cross-spectrum function, and the match between the target and simulated cross-correlation function. Both the phase and magnitude of the target cross-spectrum are well represented by the simulation, with small discrepancies in the high frequency range where there is very little process energy. Figure 6.20 compares the histograms of the measured and simulated data on linear and semi-log scales. Figures 6.19 and 6.20 demonstrate the ability of the MSC II algorithm to simulate correlated non-Gaussian processes.

6.8 Concluding Remarks

Spectral correction is extended to multivariate simulation. Two algorithms, MSC I and MSC II, are developed for the simulation of processes where the cross spectral terms are real or complex, respectively. MSC I iterates in the frequency domain on a design coherence function used to generate Gaussian realizations for transformation using SC, whereas MSC II iterates in the time domain on the design cross-correlation used to generate the Gaussian realizations. Examples demonstrate the effectiveness of both algorithms.

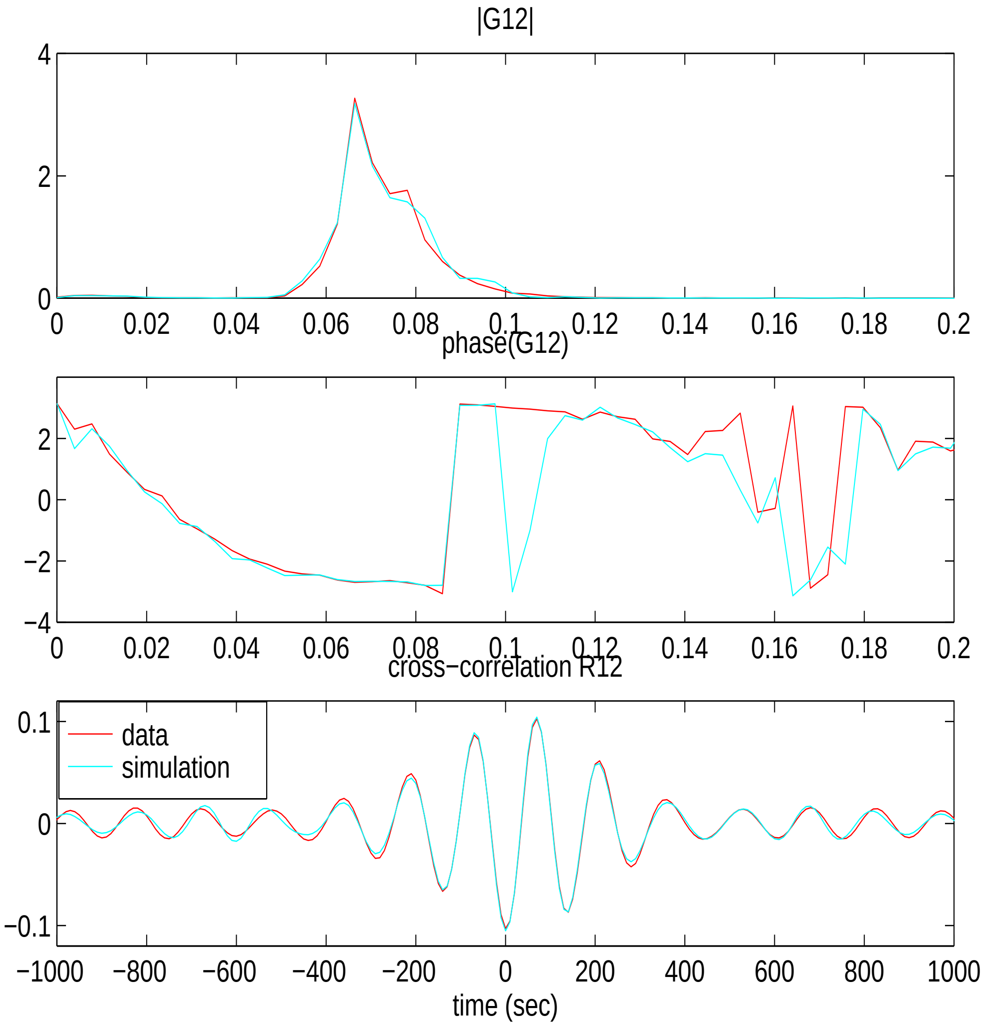


FIGURE 6.19 Section 6.7.2: Target and simulated statistics between wave and TLP surge response. (top): absolute value of the cross-spectrum, (middle): phase of the cross-spectrum, (bottom): cross correlation function.

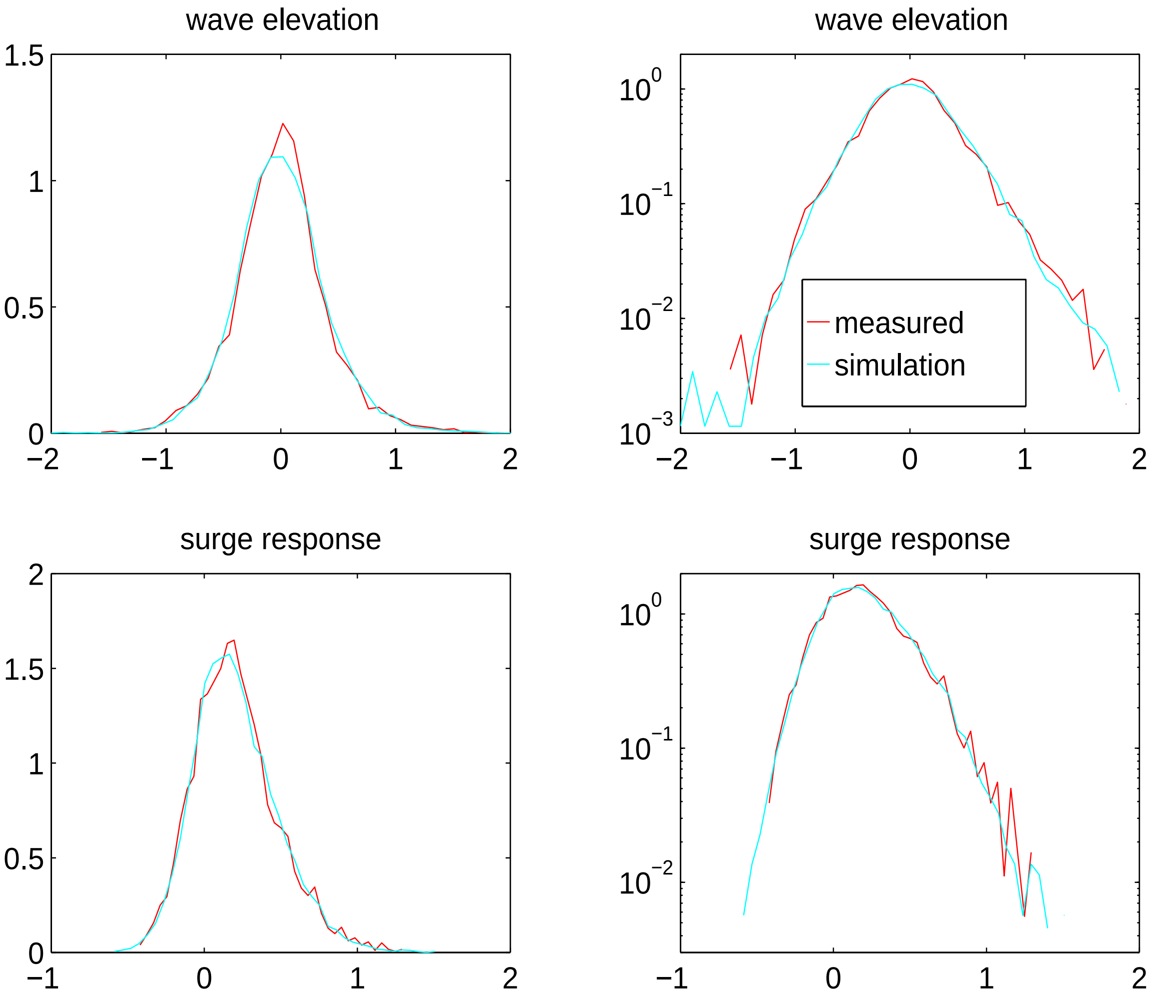


FIGURE 6.20 Section 6.7.2: PDF histograms of measured and simulated wave elevation and TLP surge data.

CHAPTER 7

CONDITIONAL SIMULATION OF NON-GAUSSIAN PROCESSES

7.1 Introduction

Simulated random signals at un-instrumented locations of a structure are often needed in areas of wind, wave and earthquake engineering. For example, malfunctioning equipment may leave a hole in a data set or information may be lacking due to a limited number of instruments. Dynamic analysis schemes which utilize integration techniques may require loading time histories at a number of unmeasured locations. Conditional simulation methods simulate realizations at desired locations conditioned on the measured data at other locations. Gaussian methods were discussed in a previous chapter, here we present a non-Gaussian conditional simulation method using spectral correction.

The previous chapter dealt with the simultaneous simulation of multiple correlated non-Gaussian processes, an unconditional simulation method. The multi-variate non-Gaussian simulation algorithms have been extended to include conditional simulation in both the frequency and time domains. The frequency domain conditional simulation generates time histories at new locations when one or more time series for the full length interval are given. Time domain conditional simulation extends existing records beyond the total sample time for cases where conditioning time series are limited to a small subinterval of the full length.

7.2 Review of Gaussian Conditional Simulation

As discussed in the chapter on Gaussian conditional simulation, two approaches have been introduced in which the simulation is either based on a linear estimation (kriging) or on a conditional probability density function. It was shown that these two approaches are equivalent. Following Borgman's work on ocean waves [7], Kareem and Murlidharan [58] have developed schemes for conditional simulation of Gaussian wind fields utilizing both frequency and time domain conditioning. This work is outlined below as a short overview of the conditional simulation methods presented in the Gaussian simulation chapter.

Consider a pair of correlated Gaussian random vectors $\boldsymbol{V}_1$ and $\boldsymbol{V}_2$. Let the two variate normal distribution of these variables be denoted

$$p(\boldsymbol{V}) = p\begin{bmatrix} \boldsymbol{V}_1 \\ \boldsymbol{V}_2 \end{bmatrix} = N\left(\begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix}, \begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix}\right) , \tag{7.1}$$

where μ_i is the mean value of the variable i, and C_{ij} is the auto or cross-covariance between the variables i and j. If a sample of $\boldsymbol{V}_1$ is measured and denoted as $\boldsymbol{v}_1$, then it is the conditional simulation of $\boldsymbol{V}_2$ based on the measured record $\boldsymbol{v}_1$ that is desired. The conditional PDF for $\boldsymbol{V}_2$ given the information on $\boldsymbol{V}_1$ is expressed as

$$p(\boldsymbol{V}_2 \ | \boldsymbol{V}_1 = \boldsymbol{v}_1) = N(\mu_2 + C_{12}^T C_{11}^{-1}(\boldsymbol{v}_1 - \mu_1), C_{22} - C_{12}^T C_{11}^{-1} C_{12}) , \tag{7.2}$$

and a conditional simulation is provided by

$$(\boldsymbol{V_2}|\boldsymbol{V}_1 = \boldsymbol{v}_1) = C_{12}^T C_{11}^{-1}(\boldsymbol{v}_1 - \boldsymbol{V}_1) + \boldsymbol{V}_2 \ . \tag{7.3}$$

where $\boldsymbol{V}_1, \boldsymbol{V_2}$ are unconditionally simulated known and unknown variates, respectively, and $\boldsymbol{v}_1$ is the known measured variate. Derivations of the covariance matrices C_{11} and C_{12} in the time and frequency domain provide the necessary information needed for conditional simulation.

For clarity in the following presentation, the subscripts k, u will replace $1, 2$ to indicate known and unknown processes, respectively. Using this convention, the simulation of the known portion, simulation of the unknown, the measured known, and the covariance matrices, $\boldsymbol{V}_1, \boldsymbol{V}_2, \boldsymbol{v}_1, \boldsymbol{C}_{11}, \boldsymbol{C}_{12}$ will be represented in the schematics as $\boldsymbol{V}_k, \boldsymbol{V}_u, \boldsymbol{v}_k, \boldsymbol{C}_{kk}, \boldsymbol{C}_{ku}$, and Eq. 7.3 can be replaced with

$$(\boldsymbol{V}_u|\boldsymbol{v}_k) = C_{ku}^T C_{kk}^{-1}(\boldsymbol{v}_k - \boldsymbol{V}_k) + \boldsymbol{V}_u \ . \tag{7.4}$$

When more than one known or unknown process is under consideration, the subscripts k and u will represent all the processes. Thus, for example, C_{kk} could be written more explicitly as $C_{k_i k_j}$, where $i, j = 1 \ldots n-1$ known processes are under consideration.

7.3 Methodology for Extension to Non-Gaussian Simulation

7.3.1 Problem Statement

Conditional simulation begins with the unconditional simulation of all the known and desired processes, denoted above as $\boldsymbol{V}_k, \boldsymbol{V}_u$. In frequency based simulation, this means the unknown spatially separated variables are simulated along with new realizations of the known processes. In time based simulation, the single record being conditionally extended based on its known portion is first unconditionally simulated to its full desired length. In both cases, the unconditionally simulated processes are then combined with the known processes using either the spectral or covariance matrix (Eq. 7.3) under the condition that all processes involved are Gaussian.

We are restricted to using Gaussian processes when combining simulated and measured records to arrive at conditionally simulated records using Eq. 7.3. Thus a direct approach of simulating $\boldsymbol{V}_k, \boldsymbol{V}_u$ as non-Gaussian records through spectral correction and applying Eq. 7.3 is not possible. Methods using a combination of nonlinear transformation, spectral correction, and application of Eq. 7.4 on Gaussian parent processes are presented next.

7.3.2 Time Domain Non-Gaussian Conditional Simulation

In the time domain case we seek to extend the length of a known (presumably measured) non-Gaussian time history using the statistics from the known portion of length τ, and a simulated record of the desired length T, where $\tau < T$. With respect to Eq. 7.4, the simulated record from $1 \ldots \tau$ is denoted $\boldsymbol{V}_k$, and from $\tau \ldots T$ is denoted $\boldsymbol{V}_u$. The known record is of length $1 \ldots \tau$ and denoted $\boldsymbol{v}_k$. The covariance matrix measured from $\boldsymbol{V}_k, \boldsymbol{V}_u$ is used to generate C_{kk} and C_{ku}, and Eq. 7.4 is used to generate the conditional simulation of length $1 \ldots T$.

The approach taken in the time domain is to transform the non-Gaussian measured (known) process such that its third and forth order moments are the expected values for a Gaussian process via backward modified Hermite transformation. Once the known non-

Gaussian process is back-transformed, its covariance and auto-spectrum are measured and used to simulate the needed $\boldsymbol{V}_k, \boldsymbol{V}_u$ for Eq. 7.4 using a standard Gaussian unconditional simulation technique. After application of Eq. 7.4, the resulting conditionally simulated Gaussian process is transformed to the desired non-Gaussian domain.

7.3.2.1 Schematic Representation

Figure 7.1 is a schematic of the method to conditionally simulate an extended length of a measured non-Gaussian process. The method is divided into 7 steps.

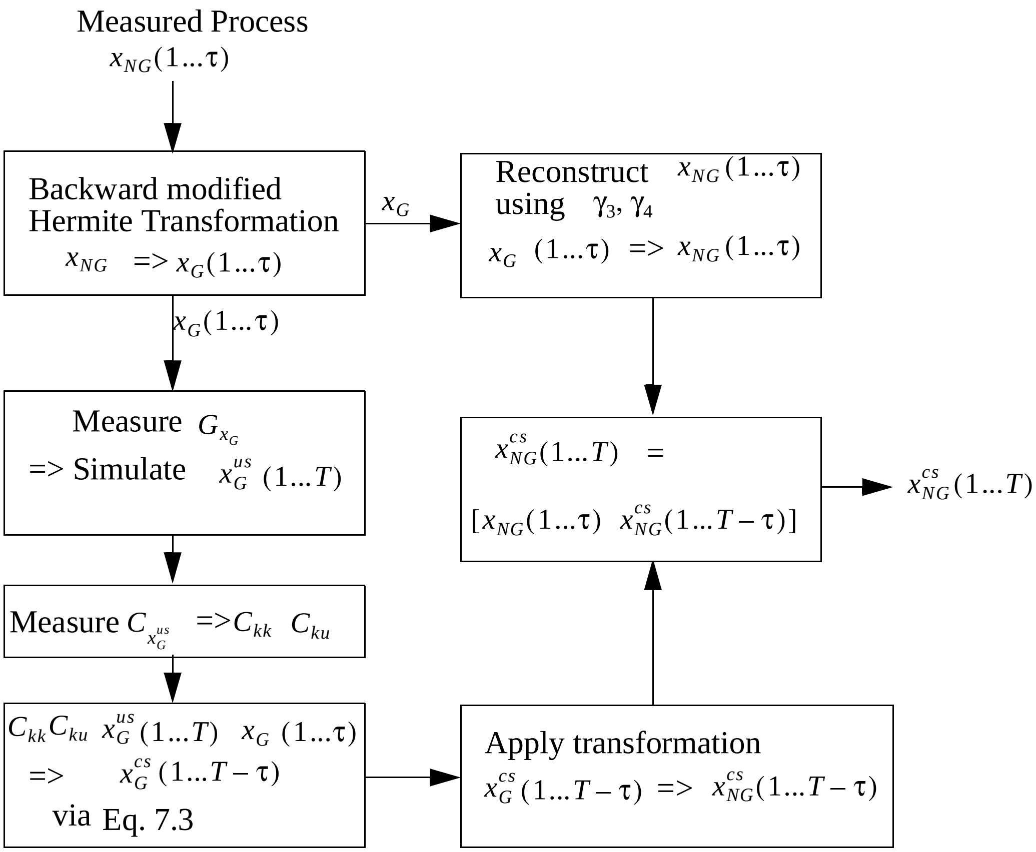


FIGURE 7.1 Schematic of the Time Domain non-Gaussian conditional simulation method.

1) Begin with a measured non-Gaussian process x_{NG} of length τ. A backward modified Hermite transformation (see section 3.2.3 and [32]) is applied to x_{NG} to produce a process x_G which is statistically Gaussian. The necessary parameters γ_3, γ_4 for the inverse operation $x_G \to x_{NG}$ are retained for later reconstruction of x_{NG}.

2) The power spectral density G_{x_G} of the Gaussian process x_G is then measured and used to produce an unconditional simulation of x_G denoted x_G^{us} which is of length T.

3) The covariance of x_G^{us} is measured, denoted $C_{x_g^{us}}$, and split into its components C_{kk}, C_{ku}.

4) Equation 7.4 is now applied using C_{kk}, C_{ku}, $\nu_k = x_G$, $\boldsymbol{V}_k = x_G^{us}(1...\tau)$, and $\boldsymbol{V}_u = x_G^{us}(\tau...T)$. This results in a conditionally simulated Gaussian process of length $T - \tau$, denoted $x_G^{cs}(1...T-\tau)$.

5) The known portion of the original non-Gaussian record, $x_{NG}(1...\tau)$, is reconstructed from $x_G(1...\tau)$ by applying a forward transformation using the parameters γ_3, γ_4 retained in step 1).

6) A nonlinear transform is applied to the Gaussian conditionally simulated record $x_G^{cs}(1...T-\tau)$, producing the conditionally simulated non-Gaussian record $x_{NG}^{cs}(1...T-\tau)$.

7) $x_{NG}^{cs}(1...T-\tau)$ is placed at the end of the known record $x_{NG}(1...\tau)$ to yield a conditionally simulated non-Gaussian process of length T as in

$$x_{NG}^{cs}(1...T) = [x_{NG}(1...\tau) \;\; x_{NG}^{cs}(1...T-\tau)] \; . \tag{7.5}$$

The nonlinear transform in step 6) is not straightforward, and will be discussed following an example of the method.

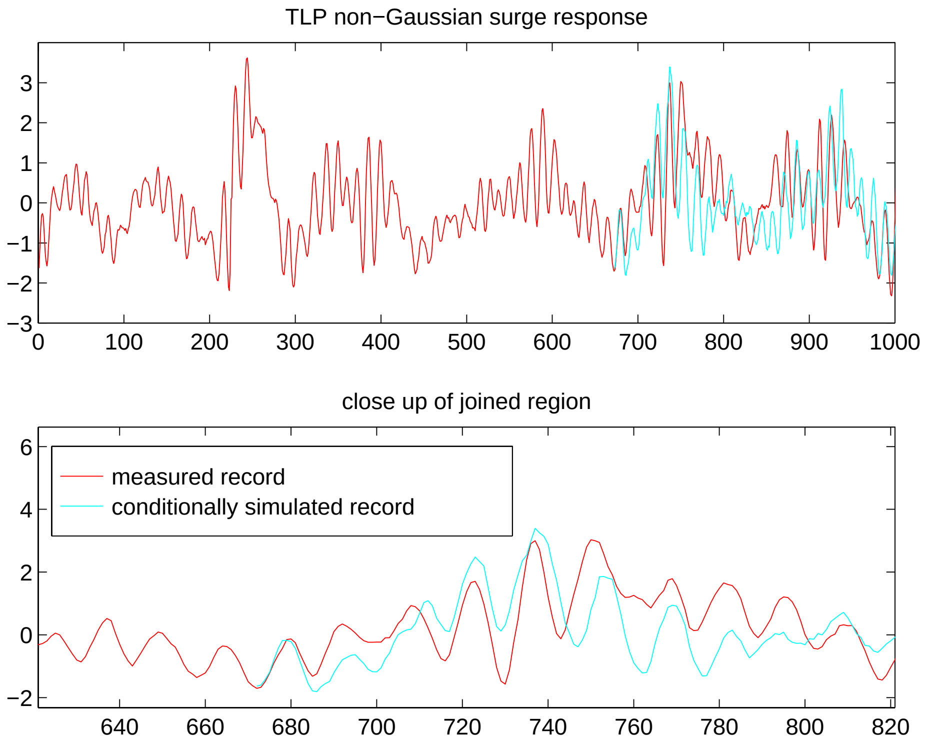
7.3.2.2 Example: TLP Response

A demonstration of the proposed technique is seen in Fig. 7.2. The measured data is an experimental non-Gaussian TLP surge response record consisting of 1000 data points. Let us assume that the only the first 670 data points were measured, and it is desired to augment the record such that it consists of 1000 points. In terms of the schematic, $t = 670$, and $T = 1000$ for a sampling rate of 1.0 Hz.

The first 670 data points are the known portion of a non-Gaussian record, denoted v_k in section 7.2, and also denoted $x_{NG}(1...\tau)$ in the schematic representation. The remaining $x_{NG}^{cs}(1...T-\tau) = x_{NG}^{cs}(1...330)$ data points are simulated using the method discussed in section 7.3.2. The full measured record is shown in Fig. 7.2 as the black line. The conditional simulation, x_{NG}^{cs}, of the last 671-1000 points is overlaid in gray for comparison. The conditional simulation is expected to have statistical characteristics similar to the record it replaces, as it does in this example. Note that the portion of the dark line record from 671-1000 is not used in the simulation. Statistical comparisons will be made in later examples following a discussion of the transformation in step 6).

7.3.2.3 Non-Gaussian Transformation in Step 6)

In the previous example, the nonlinear transformation applied in step 6) to x_G^{cs} is the same transformation used to reconstruct the known portion of the record in step 5), which is just the inverse of the transformation used in step 1). Application of the same transformation to both the known and simulated portions of the Gaussian record $x_G^{cs}(1...T)$ maintains the property of smooth transition at the joint.


TLP non−Gaussian surge response

close up of joined region

FIGURE 7.2 A realization of conditionally simulated TLP non-Gaussian surge response.

The difficulty is that this transformation leads to distortions in the statistics of the non-Gaussian simulated portion $x_{NG}^{cs}(1\ldots T-\tau)$. Over an ensemble of conditional simulations using the same data as in the previous example, the skewness and kurtosis of $x_{NG}^{cs}(1\ldots T)$ tend to overshoot the target values by approximately 10%. Thus the conditionally simulated non-Gaussian portion does not match the target non-Gaussian statistics, and does not provide an adequate simulation.

One possible solution is to apply the optimizing modified Hermite transformation to the conditionally simulated portion of the Gaussian time history, $x_{G}^{cs}(1\ldots T-\tau)$. The coefficients used to bring the first half (known) back to its original non-Gaussian form are not applied to the second half, rather a new set of coefficients are found such that the first four moments of the second half match those of the first half. The resulting statistics of the conditionally simulated time history will match the target statistics.

The problem with the above solution is at the joining portion, where the known non-Gaussian time history meets with the conditionally simulated portion. In the Gaussian domain the known portion joins smoothly with the simulated portion via Eq. 7.4. If one nonlinear transformation is applied to the known half, and another to the conditionally simulated half, the smooth transition is distorted.

Thus we have two competing distortions. In order to maintain smoothness at the transition, the same nonlinear transformation should be applied to both the known and simulated portions of the Gaussian signal. In order avoid a bias in the desired statistics of the simulated half, two separate nonlinear transformations should be applied to the known and conditionally simulated portions. To correct one is to distort the other.

It is possible to have both smoothness and accurate moments by combining the two different transformation options. The conditionally simulated portion of the Gaussian signal $x_G^{cs}(1...T-\tau)$ is transformed to two separate cases of $x_{NG}^{cs}(1...T-\tau)$ using each of the two transformation options. These two are averaged together over a number of data points using a linear weighting function. The simulation designed for smoothness dominates where the known and simulated processes join, and in the remaining portion the simulation designed for accurate moments prevails. The two cases of $x_{NG}^{cs}(1...T-\tau)$ are created using only slightly different input parameters to a Hermite transformation. Thus their differences are small, and the smoothing function does not have to account for any large discrepancies.

This hybrid method is shown in a schematic in Fig. 7.3., where the two transformations of x_G^{cs} are delineated x_G^{cs1} and x_G^{cs2}. The simulation, x_{NG}^{cs1}, which uses the same transformation as the known portion will contribute to the first NM points of the hybrid simulation, each successively weighted less than the previous. x_{NG}^{cs2} uses a separate transformation optimization on the Gaussian simulation, $x_G^{cs}(1...T-\tau)$ to bring it back to the desired higher moments. x_{NG}^{cs2} is un-weighted at the transition, then weighted successively higher throughout the first NM points. Beyond the first NM points the hybrid simulation consists entirely of x_{NG}^{cs2}. The number of points, NM, used in the weighted combination of x_{NG}^{cs1} and x_{NG}^{cs2} is set such that the combining procedure takes place over several cycles of a dominant frequency. This method results in a conditionally simulated non-Gaussian time history, x_{NG}^{cs}, which matches the target moments, and provides a smooth transition between the known and simulated portions of the conditional simulation.

A similar linear combination technique has been applied for the simulation of non-stationary earthquake records [97].

7.3.2.4 Examples of the Hybrid Method

Examples of this method are shown in Figs. 7.4 and 7.5, where TLP surge response and measured rooftop suction pressure are the signals being conditionally simulated. For each of these examples, the measured data consists of $\tau = 512$ points, and the total desired length is $T = 1024$ points. A close up of the joining region is shown in each figure to demonstrate the smooth transition. Table 7.1 below lists the desired higher order statistics from the known record portion, and the resulting statistics of the ensemble average of 10 simulations out to the desired time. The skewness and kurtosis from simulations are measured over the entire extended record, including both known and simulated portions,

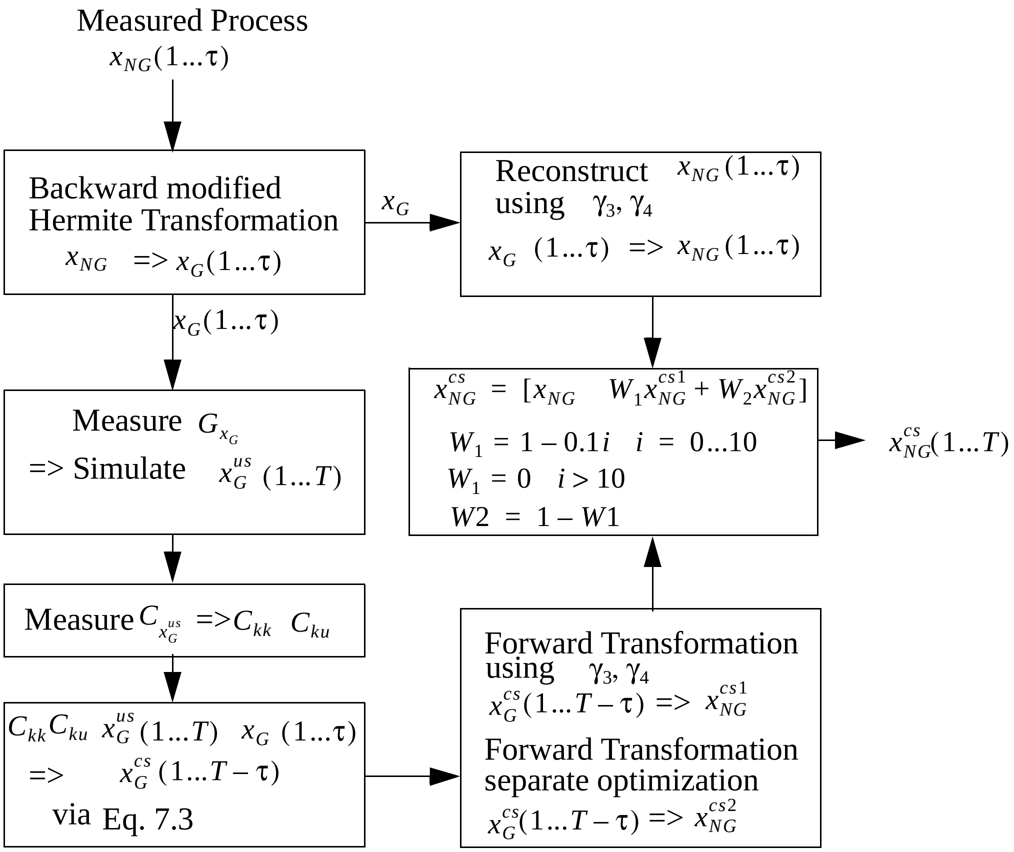


FIGURE 7.3 Schematic of the Time Domain non-Gaussian conditional simulation method with hybrid transformation.

and are clearly within an acceptable range of error with respect to the target taken from the known portion only.

The PDF and PSD of the target and an ensemble of 10 simulations are shown in Figs. 7.6-7.9. The measured data histograms and PSDs are estimated from the limited data of the known portion of the non-Gaussian record, and as such contain significant variance error. The results presented in table 7.1, and Figs. 7.6 and 7.7 indicate an acceptable representation of the known data. The target PSD in Fig. 7.8 and 7.9 have a variance error of about 35%. The measured spectrum from the simulated records fit well within the variance error bounds.

TABLE 7.1 TARGET HIGHER MOMENTS AND SIMULATION RESULTS FOR FIG. 7.4

	std	**skewness**	**kurtosis**
TLP data	1.0000	0.6505	3.2350
TLP simulation, 100 ensembles (std)	0.9996	0.6533 (.054)	3.2989 (.110)
Pressure data	1.0000	-1.1631	4.6980
Pressure simulation, 100 ensembles (std)	0.9996	-1.1586 (.04)	4.7369 (.174)

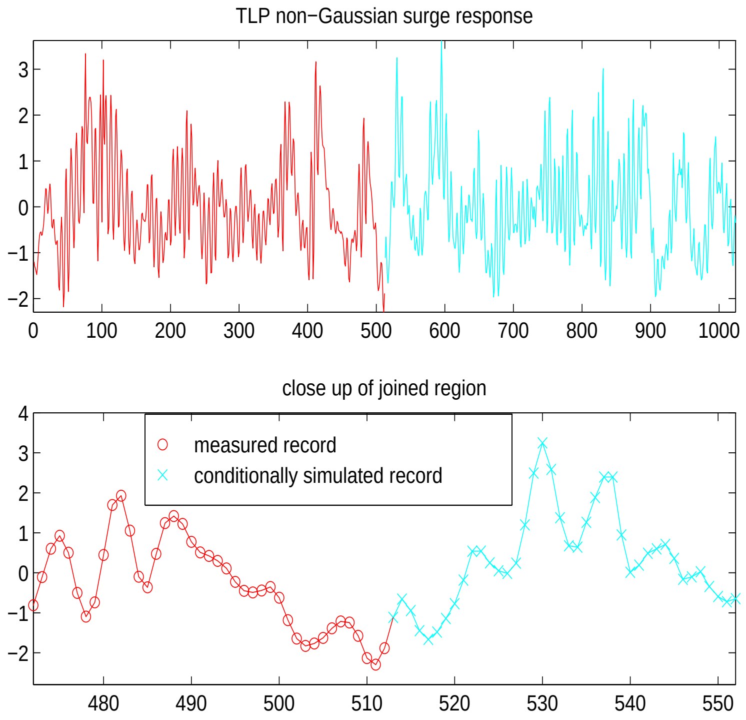


FIGURE 7.4 Measured TLP surge response data and a time domain non-Gaussian conditional simulation.

7.4 Frequency Domain Conditional Simulation

In the frequency domain, conditional simulation adds additional records at new locations using the statistics from records measured at other locations. Conditional simulation begins with the unconditional simulation of all the known and unknown processes, $\boldsymbol{V}_k$, $\boldsymbol{V}_u$, where $\boldsymbol{V}_k$ are new realizations of the known $n-1$ processes $\boldsymbol{v}_k$. The cospectral matrix

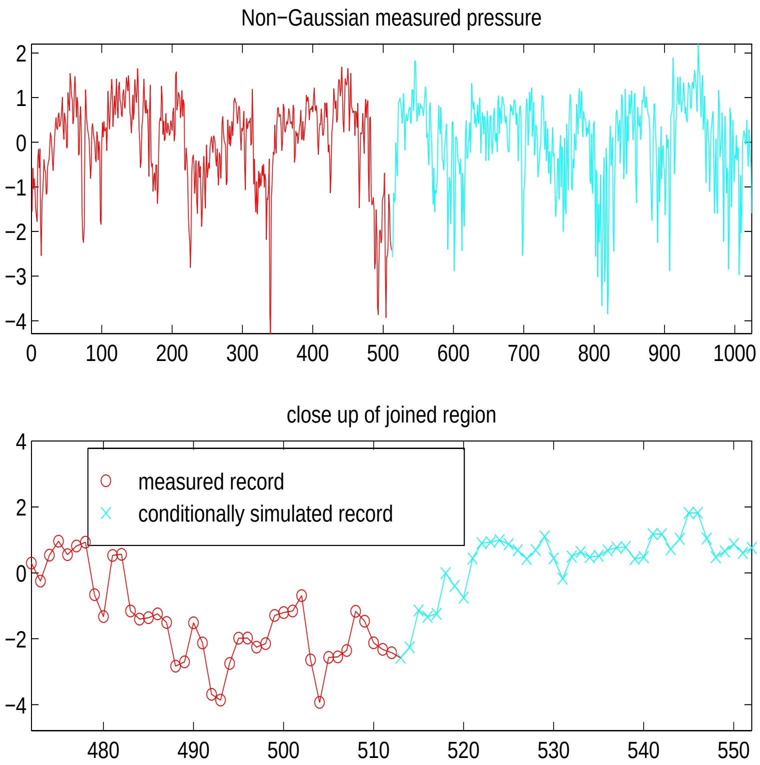


FIGURE 7.5 Measured rooftop pressure and a time domain non-Gaussian conditional simulation.

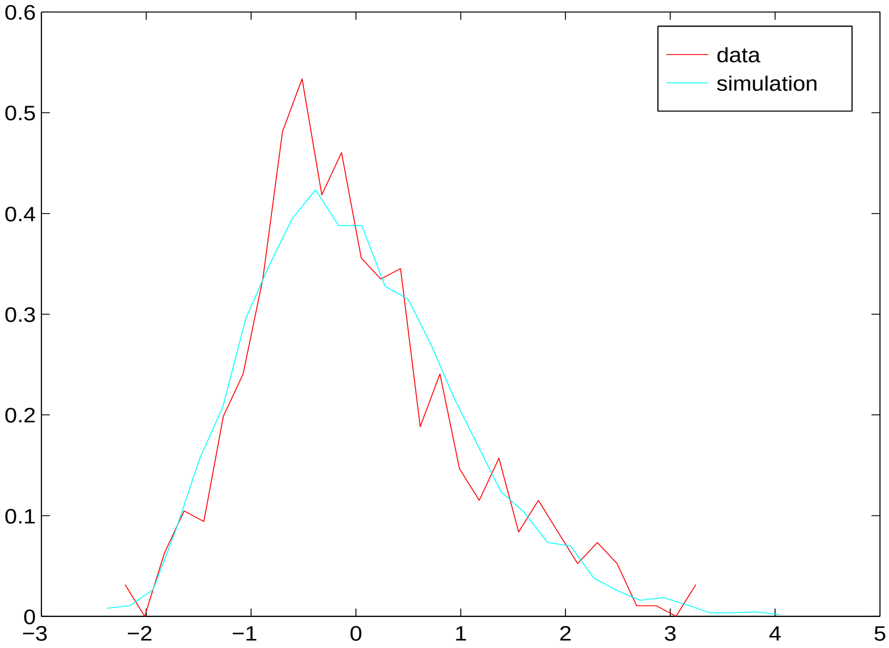


FIGURE 7.6 PDF Histogram of known TLP data and conditional simulation using 100 ensembles

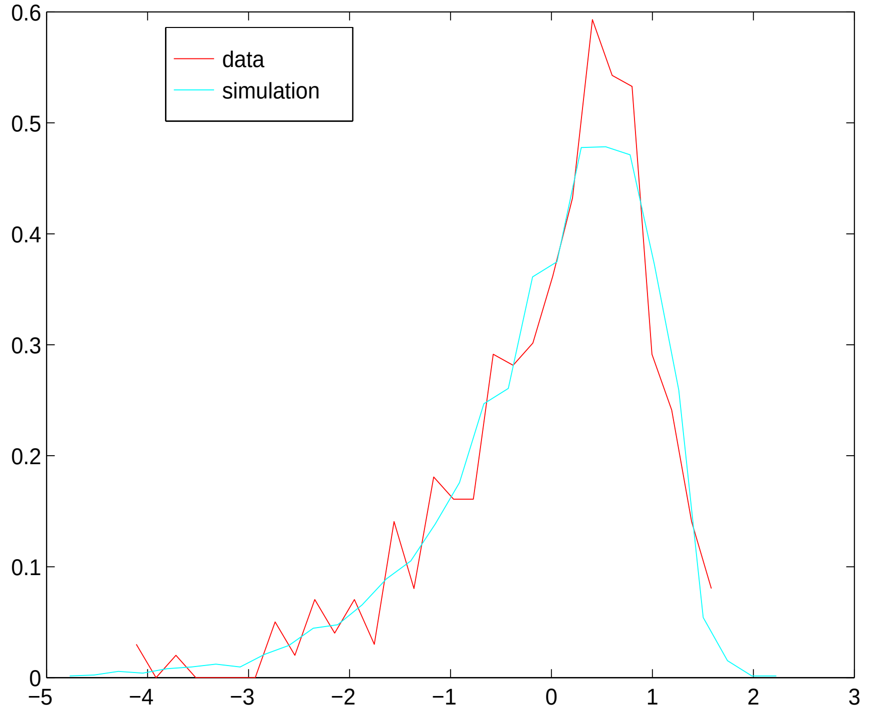


FIGURE 7.7 PDF Histogram of known pressure data and conditional simulation using 100 ensembles

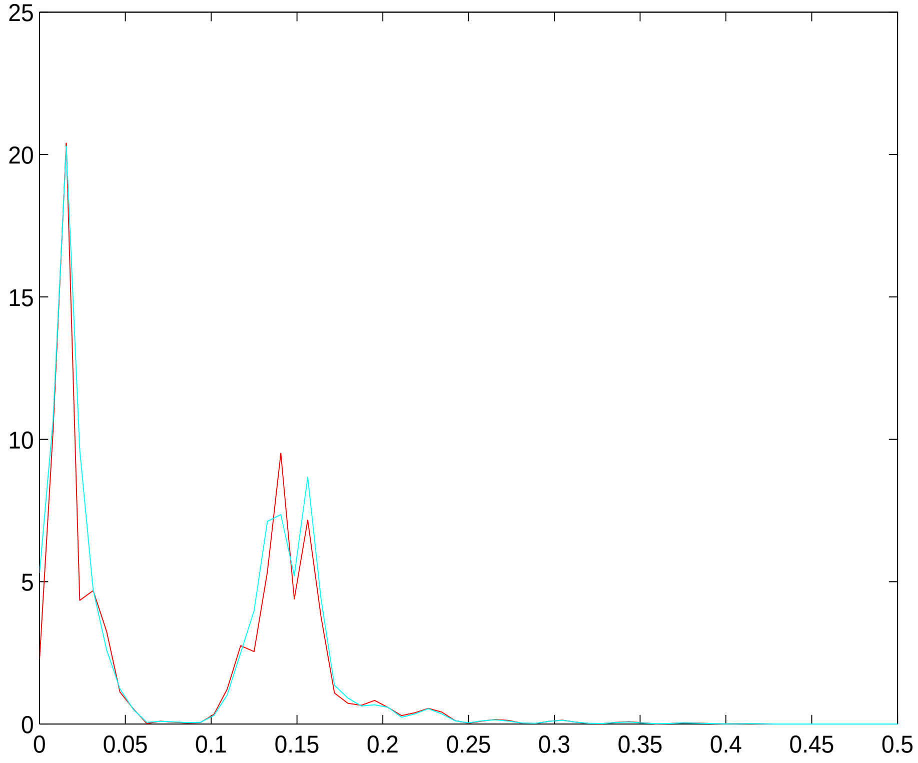


FIGURE 7.8 Power spectral density of known TLP data and conditional simulation using 100 ensembles

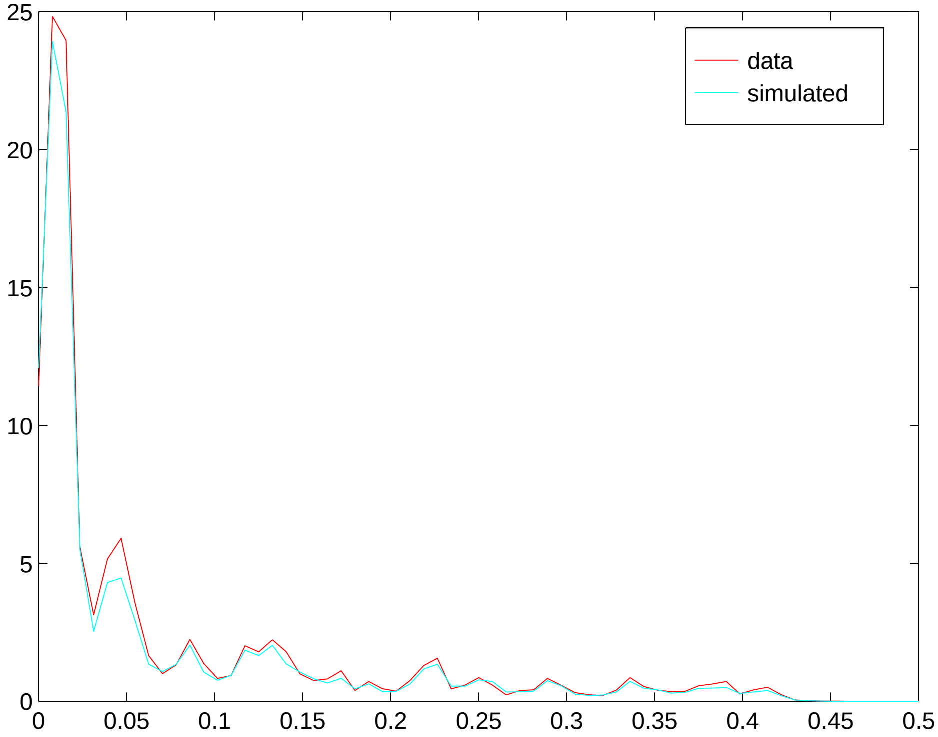


FIGURE 7.9 Power spectral density of known pressure data and conditional simulation using 100 ensembles

measured between $\boldsymbol{V}_k, \boldsymbol{V}_u$ is used to generate R_{kk}, R_{ku}, and the cross-covariance functions become

$$C_{kk} = R_{kk} - \mu_{k_i}\mu_{k_j}, \tag{7.6}$$

where $\mu_{k_i}\mu_{k_j}$ are the expected value of the ith and jth known processes, and $i, j = 1 \ldots n-1$ known processes. The subscripts are left off the covariance and correlation terms for convenience.

Equation 7.4 is then applied under the condition that all processes involved are Gaussian.

7.4.1 Methodology for Extension to Non-Gaussian Simulation

Recall the chapter concerning multivariate unconditional simulation. The method iteratively updates the design cross-correlation functions used to generate Gaussian processes, such that application of spectral correction for transformation to the non-Gaussian domain distorts the resulting cross-spectrum toward the target.

The same idea is applied for conditional simulation, where the multivariate algorithm is iteratively used to simulate the fields $\boldsymbol{V}_k, \boldsymbol{V}_u$. Equation 7.4 is applied in the Gaussian domain, and the resulting conditionally simulated process is transformed to non-Gaussian using spectral correction. The distortion from the transformation is corrected through iterative changes to the target cross spectral matrix input to the multi-variate spectral correction algorithm used to generate $\boldsymbol{V}_k, \boldsymbol{V}_u$.

7.4.2 Schematic Representation

Figure 7.10 is a schematic of the method to conditionally simulate non-Gaussian processes in the frequency domain. The algorithm is given in 8 steps.

1) The algorithm begins with the known non-Gaussian processes $\boldsymbol{v}_k$, their measured covariance matrices C_{kk}, the target spectra of the unknown processes G_{uu}, their target moments $m_u^{1\ldots4}$, and the target cross-covariance matrix between the measured and simulated processes G_{ku}^T. The design cross-covariance is set to the target cross-covariance $C_{ku}^D = C_{ku}^T$ for the first iteration.

2) MSC is applied to produce Gaussian versions of $\boldsymbol{V}_k, \boldsymbol{V}_u$, denoted $\boldsymbol{V}_k^G, \boldsymbol{V}_u^G$ using the input C_{kk}, C_{ku}^D, G_{uu}.

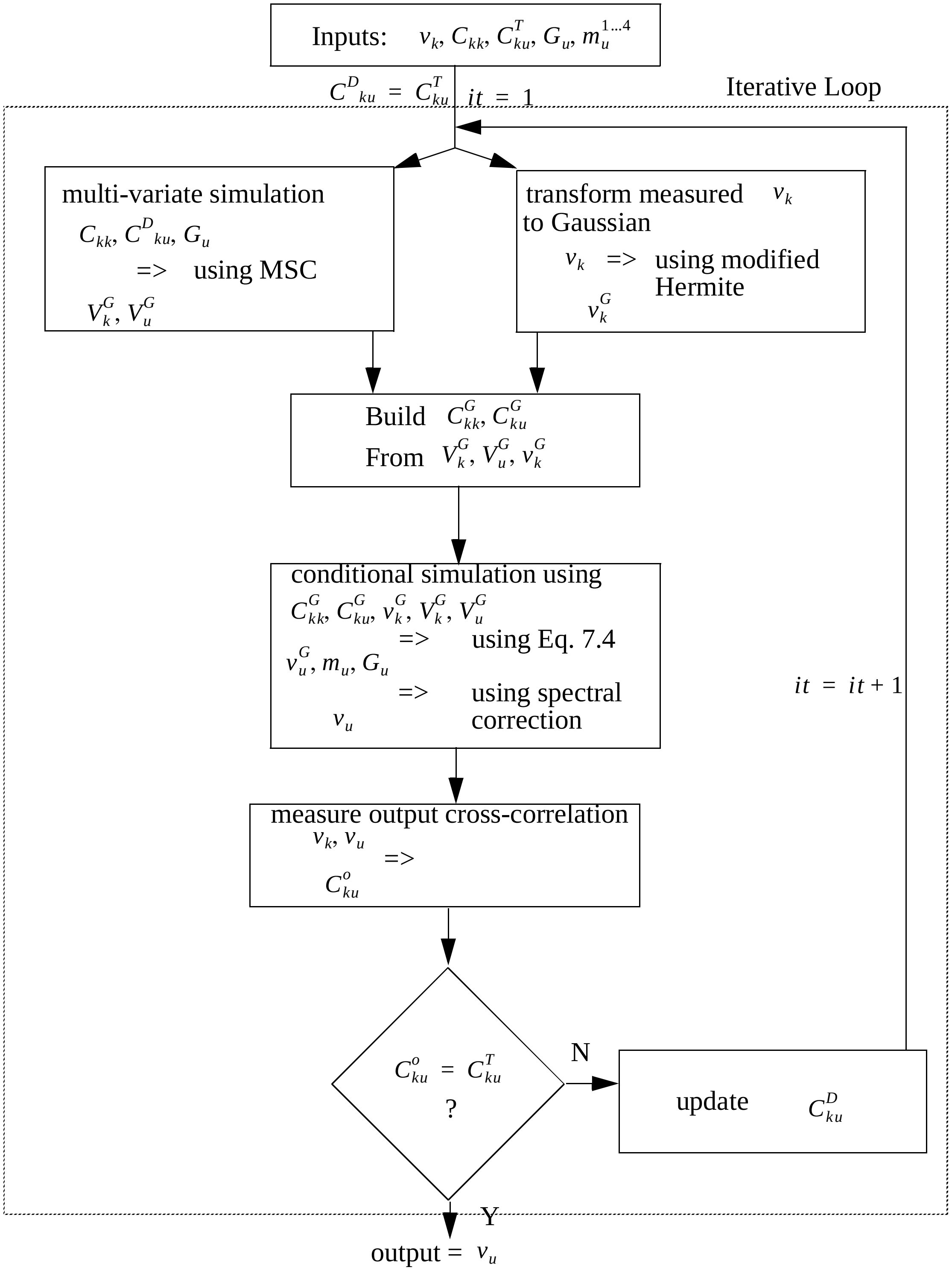


FIGURE 7.10 Schematic of the frequency domain conditional simulation algorithm using spectral correction.

3) A modified backward Hermite transform is applied to $\boldsymbol{v}_k$ to produce a Gaussian version $\boldsymbol{v}_k^G$ and $\boldsymbol{V}_k^G, \boldsymbol{V}_u^G$ have the desired spectral characteristics of the final non-Gaussian processes.

4) The output of steps 1 and 2 are the Gaussian processes $\boldsymbol{V}_k^G, \boldsymbol{V}_u^G, \boldsymbol{v}_k^G$ from which the spectral matrices C_{11}^G, C_{12}^G are estimated.

5) Equation 7.4 is then applied to produce a Gaussian conditional simulation at the unknown locations, denoted v_u^G.

6) The Gaussian process v_u^G are sent through the spectral correction transformation to produce the non-Gaussian simulation v_u with target moments, $m_u^{1...4}$, and autospectra G_{uu}.

7) The cross-covariance between v_k and v_u are measured, denoted C_{ku}^o, and compared with the desired target cross-covariance C_{ku}^T. If the error between the measured and target covariance is not acceptable, the design cross-covariance used to generate $\boldsymbol{V}_k^G, \boldsymbol{V}_u^G$ is updated for use in the next iteration. The error criteria and the updating scheme for C_{ku}^D are the same as are applied in the MSC algorithm.

8) The updated $C^D{}_{ku}$ is sent to step 2) for the next iteration. The same random phase vectors are used to generate $\boldsymbol{V}_k^G, \boldsymbol{V}_u^G$ in each iteration.

7.4.3 Example Application: Measured Pressure

The statistics of a measured non-Gaussian roof suction pressure record are used in conjunction with a standard wind velocity coherence model to generate four spatially separated non-Gaussian pressure records using MSC. The points of measurement are assumed to be 4 meters apart perpendicular to the wind direction, and no separation parallel to the wind direction. The first four normalized central moments of each process is [0, 1, -1.05, 5.5]. The fourth location is assumed now to be damaged, and the conditional simulation algorithm presented above is used to replace it based on knowledge of the other three existing records.

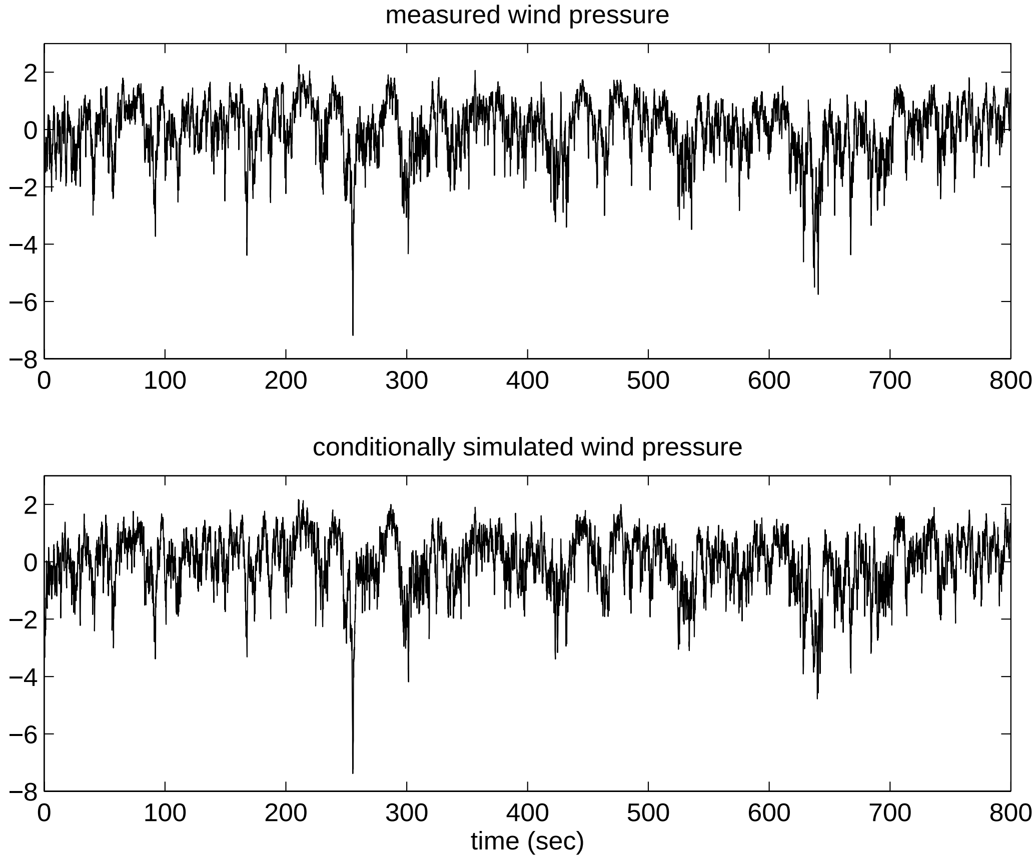


FIGURE 7.11 The pressure record assumed missing, and its simulated replacement using frequency domain non-Gaussian conditional simulation.

Figure 7.11 is a view of the missing record and its conditionally simulated replacement. Note that the records were not simultaneously simulated, and have very strong low frequency correlation. Fig. 7.12 is a view of the three known records and the simulated fourth record in the bottom plot. The fourth record has been simulated conditioned on the first three. Figure 7.13 shows the coherence between the six pairs of processes. Coherence functions involving record 4 match the target coherence very well, demonstrating that it is a suitable replacement for the missing record. Figure 7.14 is a comparison of the PDF of the missing recorded and its replacement. The use of the spectral correction algorithm within the conditional simulation algorithm ensures excellent agreement between target and simulated marginal distribution and auto-spectrum.

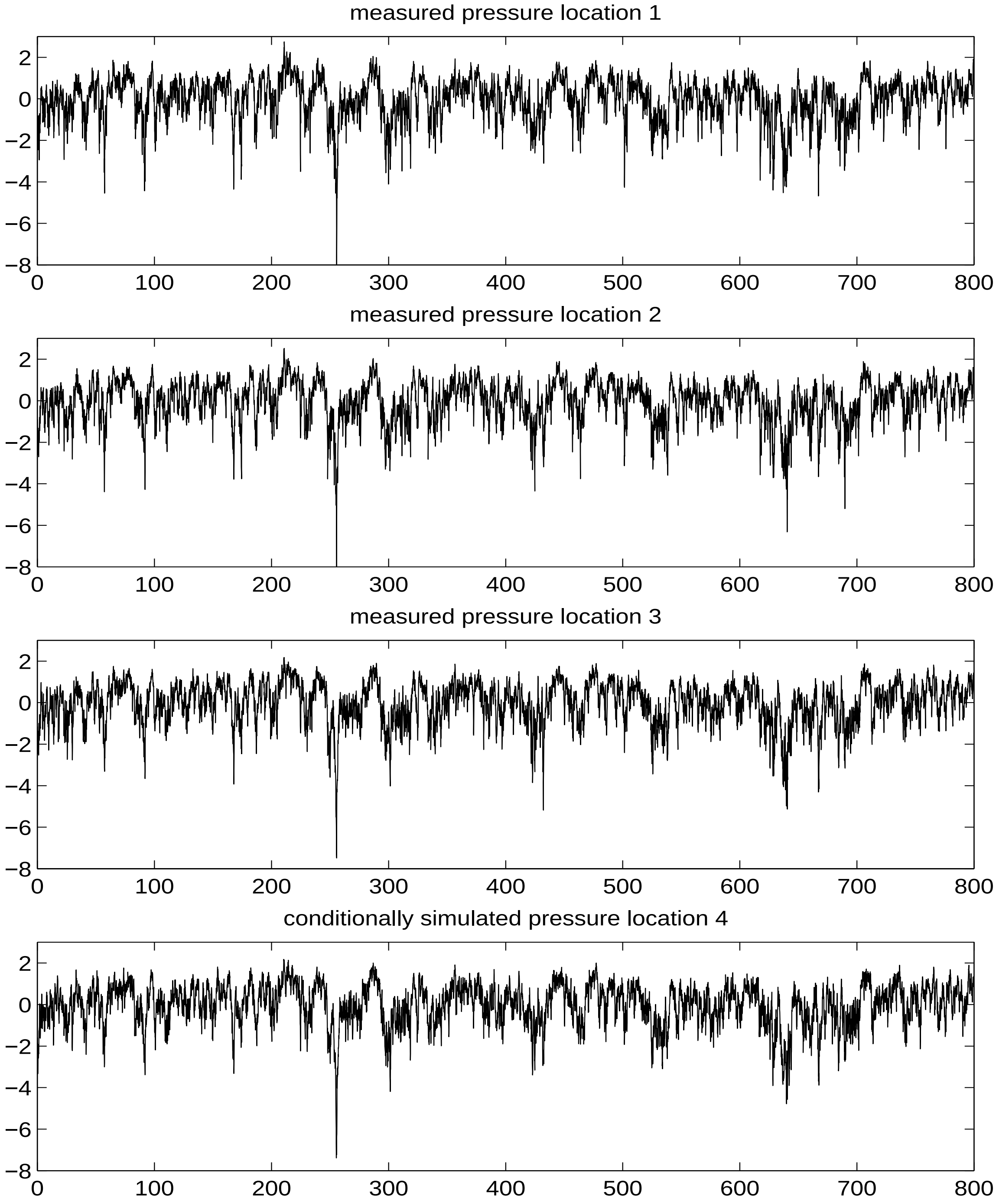


FIGURE 7.12 Three known pressure records at locations 1,2 and 3 (top three plots). A simulated record at location 4 using frequency domain non-Gaussian conditional simulation (bottom plot).

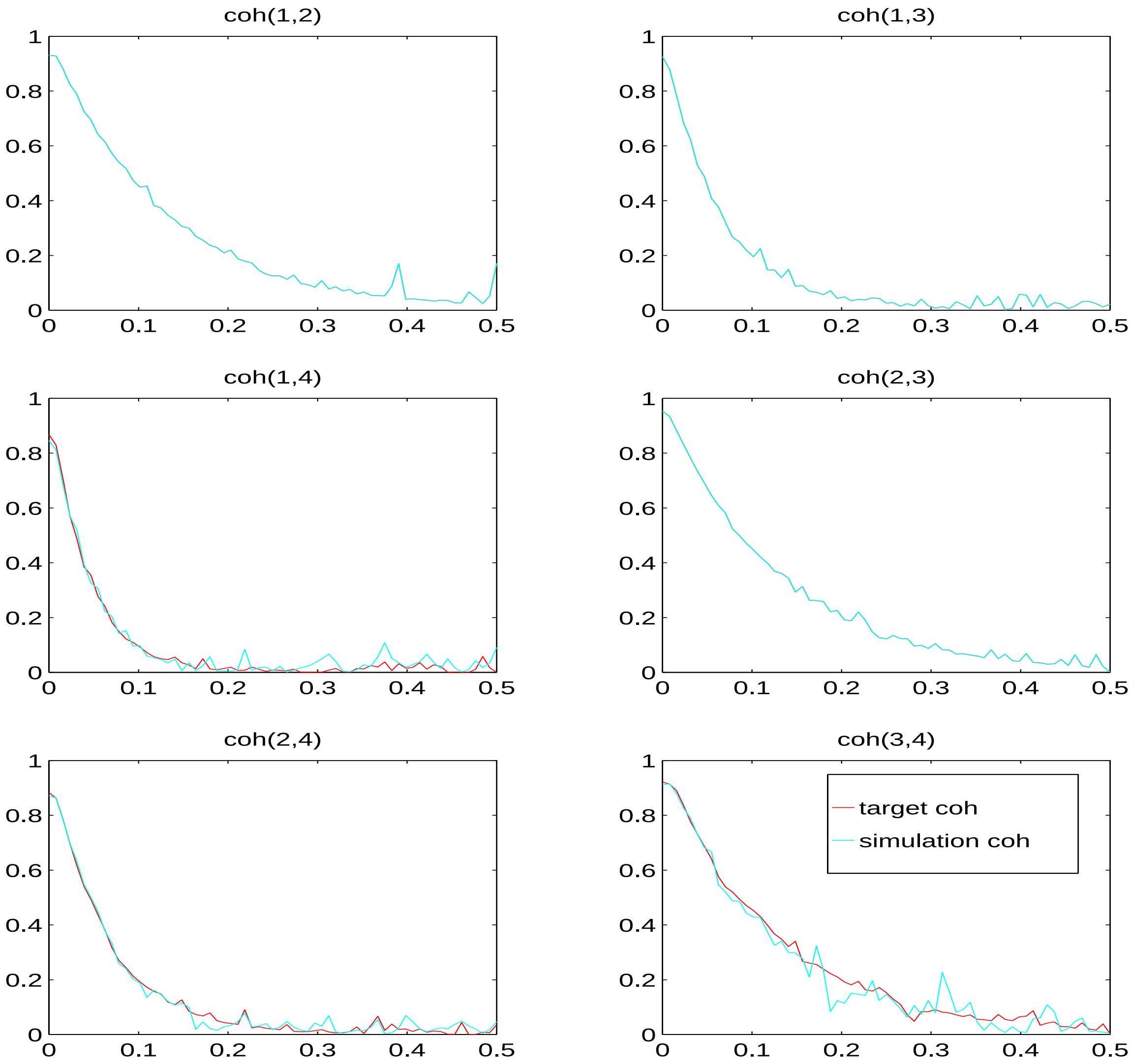


FIGURE 7.13 Comparison of the target and simulated coherence functions for the pressure data at 4 locations in the section 7.4.3 example. The simulated record is at location 4, and the known records are at locations 1,2 and 3.

7.5 Concluding Remarks

The multi-variate non-Gaussian spectral correction algorithms have been extended to include conditional simulation in both the frequency and time domains. Time domain conditional simulation extends existing records beyond the total sample time for cases where conditioning time series are limited to a small sub-interval of the full length. A Gaussian

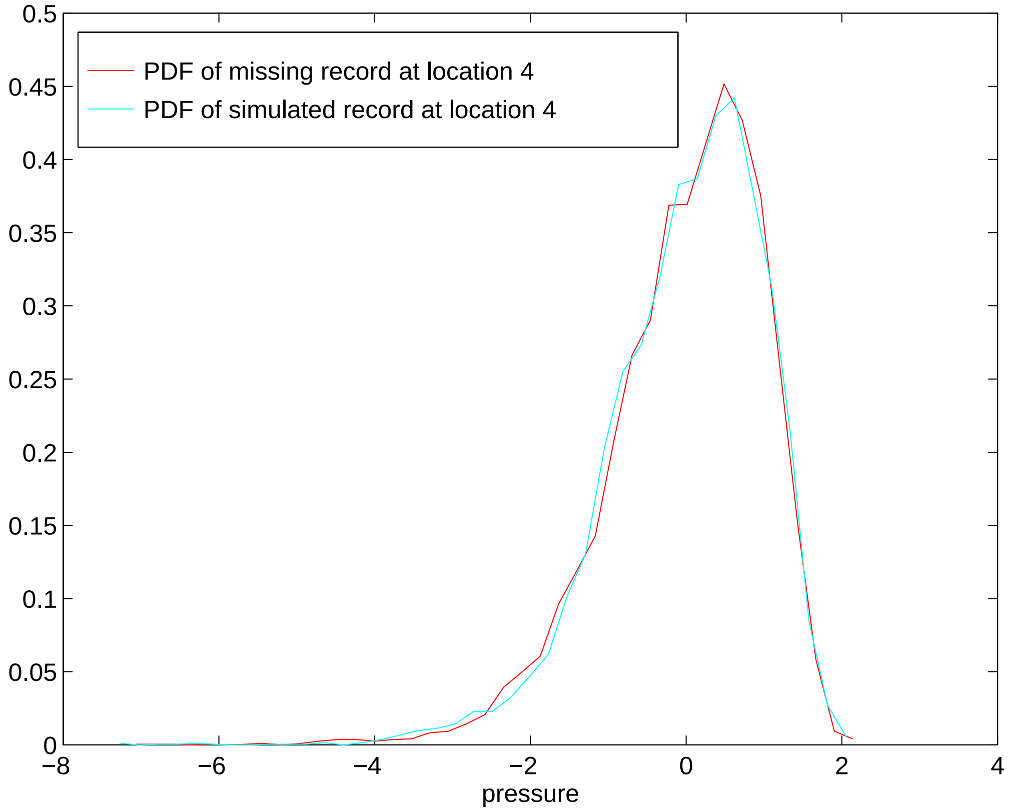


FIGURE 7.14 PDF Histogram of the pressure record at location 4 that is assumed missing (denoted measured), and histogram of its replacement using frequency domain non-Gaussian conditional simulation.

conditional simulation is transformed to non-Gaussian, and combined with the known non-Gaussian measured realization using a hybrid scheme.

The frequency domain conditional simulation generates time histories at new locations based on one or more measured time series at other locations. An iterative mapping technique is employed between the Gaussian domain, where the Gaussian conditional simulation method is applied, and the non-Gaussian domain, where target and measured statistics are compared, and updated for the next iteration are made. Spectral correction is used as the mapping from Gaussian to non-Gaussian, and ensures that the auto-spectral and non-Gaussian characteristics of each realization are accurate. Examples validate these non-Gaussian conditional simulation methods.

CHAPTER 8

SUMMARY AND FUTURE WORK

8.1 Summary of Contributions

8.1.1 Chapter 2

Section 2.5 presents two formulations for Gaussian conditional simulation. One formulation is referred to as kriging, and is on the best linear unbiased estimator. A second formulation is derived based on a conditional probability density function. Both approaches have been shown to be equivalent, but CPDF is often preferred for its basis on probability theory rather than a best linear unbiased estimate. It is shown in section 2.5.3 that the kriging formulation has been derived based on probability theory.

8.1.2 Chapter 3

This chapter presents several non-Gaussian simulation techniques developed during the course of this research [32, 34].

A static transformation method referred to as Modified Direct Transformation is able to produce realizations of a sample measured process using optimized iterative Hermite polynomial transformation and its inverse. An example using measured non-Gaussian suction pressure on a rooftop demonstrates the ability to produce realizations which match the spectral and probabilistic characteristics of the sample measured process. The application of this method is restricted to cases where a sample record is available, and is therefore not a generally applicable tool.

A simulation technique using neural network system identification is also presented, and its ability to reproduce higher-order spectra is demonstrated in an example. This nonlinear memory model is restricted to cases where both system input and output are available.

The Volterra series is applied to simulation when the higher order transfer functions are available. A technique is developed to allow very long simulations while maintaining computational efficiency in the frequency domain convolution with a second order transfer function of fixed size. This technique has recently been applied to a probabilistic approach to modelling nonlinear transient offshore system response [35].

8.1.3 Chapter 4

This chapter presents several approaches to modelling the PDF of known systems and measured non-Gaussian data. The traditional maximum entropy and Hermite-moments PDF models are first discussed, and modifications to these methods are presented [28]. Examples are provided to show that the modified MEM and Hermite PDF models have the potential to provide more accurate probabilistic descriptions of the tail regions of measured data. These modified models are then applied as input for simulation methods in chapter 5.

8.1.4 Chapter 5

This chapter presents the uni-variate spectral correction simulation method (SC), which is capable of producing time history realizations that match both target probabilistic and target spectral information. The desired non-Gaussian characteristics are input as either the first four central moments, or as a PDF model. The stationary and convergence properties are discussed as well as several options in the algorithm. Several examples are presented and results are compared with an existing technique. It is shown that SC performs as well and potentially better than the existing method used for comparison, and succeeds even in special cases where the existing method fails [28].

8.1.5 Chapter 6

Spectral correction is extended to multi-variate simulation, where several non-Gaussian correlated time histories are generated simultaneously. Strategies are developed to handle cases where the cross spectrum between realizations is real and complex. Examples are shown using both measured data and models to describe the non-Gaussian statistics and correlation [31].

8.1.6 Chapter 7

In the frequency domain, multivariate non-Gaussian spectral correction simulation is extended to multivariate non-Gaussian conditional simulation. This facilitates the generation time histories at new locations based on one or more measured time series at other locations. Spectral correction is used as the mapping from Gaussian to non-Gaussian, and ensures the auto-spectral and non-Gaussian characteristics of each realization are accurate.

Time domain conditional simulation extends existing records beyond the total sample time for cases where conditioning time series are limited to a small sub-interval of the full length. A Gaussian conditional simulation is transformed to non-Gaussian, and combined with the known non-Gaussian measured realization using a hybrid scheme [29].

8.2 Future Considerations

The spectral correction simulation method is an iterative approach which applies numerical optimization of the static transformation in each iteration. While the computational cost of simulating a realization is on the same order as that of existing methods, there is a potential to greatly improve the efficiency of SC. There are several techniques currently under investigation which may significantly reduce the number of iterations necessary to complete a SC simulation. One such techniques is the incorporation of adaptive correlation methods to find the most appropriate initial Gaussian design spectrum before static transformation. Also, extension of the current multi-variate case to multidimensional is desirable.

Simulation of Gaussian Non-stationary processes has received considerable attention in the literature. Spectral correction has been developed for the simulation of stationary non-Gaussian processes. A useful extension would include the simulation of non-stationary non-Gaussian processes.

The work in this study focuses on simulation using nonlinear static transformation methods, and retains higher order spectral information in a scalar sense. The extension to nonlinear memory transformation can potentially provide more accurate representation of the distribution of higher-order moments with respect to frequency.

A new simulation method is currently being investigated which represents both the probabilistic and spectral characteristics through a vector or parameters. The simultaneous optimization of this set of parameters provides simulations with the desired characteristics.

REFERENCES

[1] Agmon, N., Alhassid, Y., and Levine, R.D., 1970, “An Algorithm for Finding the Distribution of Maximal Entropy,” Journal of Computational Physics, vol. 30, 250-258.

[2] Ammon, D., 1990, “Approximation and Generation of Gaussian and Non-Gaussian Stationary Processes,” Structural Safety, 8, 153-160.

[3] Anderson, T.W., 1958, An Introduction to Multivariate Statistical Analysis, John Wiley, New York.

[4] Ang, A. H-S. and Tang, W.H., 1975, Probability Concepts in Engineering Planning & Design, Volume I - Basic Principles. John Wiley & Sons, Inc., New York.

[5] Bendat, J.S., 1990, Nonlinear system analysis and identification from random data. John Wiley & Sons, New York

[6] Bendat, J.S. and Piersol, A.G., 1986, Random Data Analysis and Measurement Procedures, John Wiley and Sons.

[7] Borgman, L.E., 1990, “Irregular Ocean Waves: Kinematics and Forces,” The Sea, Prentice Hall.

[8] Borgman, L.E., Taheri, M., and Hagan, R., 1983, “Three-Dimensional, Frequency-domain, Simulation of Geological Variables,” Geostatistics for natural resources characterization, Part I. NATO ASI series, Verly, G., David, M., Journel, A.G., and Marechal, A., eds. Reidel, D., Dordrecht, 517-541.

[9] Chang, R.J., 1991, “Maximum Entropy Approach for Stationary Response of Non-linear Stochastic Oscillators,” J. Eng. Mech. 58, 266-271.

[10] Ciulli, S., Mounsif, M., Gorman, N. and Spearman, T.D., 1991, “On the Application of Maximum Entropy to the Moments Problem,” J. Math. Phys., 32(7), 1717-1719.

[11] Conner, D.A. and Hammond, J.L., 1979, “Modeling of Stochastic System Inputs Having Prescribed Distribution and Covariance Functions,” Applied Mathematical Modeling, 3(2).

[12] Deodatis, G., 1996, “Simulation of Ergodic Multi-Variate Stochastic Processes,” J. Eng. Mech., ASCE, 122(8).

[13] Deodatis, G., 1996, “Non-Stationary Stochastic Vector Processes: Seismic Ground Motion Applications,” Prob. Eng. Mech., 11(3).

[14] Deodatis, G., 1996, “Simulation of Stochastic Processes and Fields to Model Loading and Material Uncertainties,” to appear in book on Probabilistic Methods for Structural Design by Kluwer Academic Publishers.

[15] Deodatis, G. and Shinozuka, M., 1989, “Simulation of Seismic Ground Motion Using Stochastic Waves,” J. Eng. Mech., ASCE, 115(12), 2723-2737.

[16] Deutsch, R., 1962, Nonlinear Transformations of Random Processes, Prentice-Hall, Englewood Cliffs.

[17] Ditlevsen, O., 1981, Uncertainty Modelling. McGraw-Hill, New York, N.Y.

[18] Elishakoff, I. Ren, Y.J., and Shinozuka, M., 1994, “Conditional Simulation of Non-Gaussian Random Fields,” Eng. Struct., 16(7), 558-563

[19] Gersch, W., and Kitagawa, 1985, “A Time Varying AR Coefficient Model for Modelling and Simulation of Earthquake Ground Motions,” Earthquake Engineering and Structural Dynamics, 13(2), 124-131.

[20] Gersch, W. and Yonemoto, J. 1977, “Synthesis of Multi-Variate Random Vibrational Systems: A Two-Stage Least Squares ARMA Model Approach,” J. Snd. Vib., 52(4), 553-565.

[21] Grigoriu, M., 1984, “Crossing of Non-Gaussian Translation Processes,” J. Eng. Mech., ASCE, 110(EM4), 610-620.

[22] Grigoriu, M., 1993, “On the Spectral Representation Method In Simulation,” Prob. Eng. Mech., 8, 75-90.

[23] Grigoriu, M., 1995, Applied Non-Gaussian Processes. PTR Prentice Hall, Englewood Cliffs, NJ.

[24] Gurley, K., and Kareem, A., 1993, ”Summary of programs for and results of non-Gaussian process simulation and phase specified Gaussian simulation,” Technical report No. NDCE93-1, Dept, of Civil Engineering, University of Notre Dame.

[25] Gurley, K., and Kareem, A., 1994, “On the Analysis and Simulation of Random Processes Utilizing Higher-Order Spectra and Wavelet Transforms,” Proceedings of the second international conference on computational stochastic mechanics, Athens, Greece, Balkema Press, Netherlands.

[26] Gurley, K. and Kareem, A., 1997, “Application of Wavelet Transforms in Signal Characterization,” to appear, Engineering Structures.

[27] Gurley, K. and Kareem, A., 1996, “Numerical Experiments in Ringing of Offshore Systems Under Viscous Loads,” 15th International Conference on Offshore Mechanics and Arctic Engineering, ASME, June 16 - 20, Florence, Italy.

[28] Gurley, K. and Kareem, A., 1997, “Analysis, Interpretation, Modelling and Simulation of Unsteady Wind and Pressure Data,” to appear in the Journal of Wind Engineering and Industrial Aerodynamics.

[29] Gurley, K. and Kareem, A., 1997, “A Conditional Simulation of Non-Normal Velocity/Pressure Fields,” The Eighth U.S. National Conference on Wind Engineering, The Johns Hopkins University, Baltimore, Maryland, June.

[30] Gurley, K. and Kareem, A., 1997, “Modelling PDFs of non-Gaussian system response,” 7th International Conference on Structural Safety and Reliability (ICOSSAR), Kyoto, Japan, November.

[31] Gurley, K. and Kareem, A., “Simulation of Correlated Non-Gaussian Pressure Fields,” 2nd European and African Conference on Wind Engineering (2 EACWE), Genova, Italy, June, 1997.

[32] Gurley, K. and Kareem, A.,Tognarelli, M.A., 1996, “Simulation of a Class of Non-Normal Random Processes,“ International Journal of Nonlinear Mechanics, 31(5), 601-617.

[33] Gurley, K., Tognarelli, M., and Kareem, A., 1994, “Extremes of nonlinear stochastic systems,” Research in progress, University of Notre Dame.

[34] Gurley, K., Tognarelli, M., and Kareem, A., 1997, “Analysis and Simulation Tools for Wind Engineering,” Prob. Eng. Mech., 12(1), 9-31.

[35] Gurley, K., Waisman, F., Grigoriu, M., and Kareem, A., 1997, “Probabilistic Modelling of Ringing of Offshore Structures,” 7th International Conference on Structural Safety and Reliability (ICOSSAR), Kyoto, Japan, November, 1997.

[36] Hasselmann, K., 1962, “On the nonlinear energy transfer in a gravity wave spectrum, part I”, J of fluid mechanics, Vol. 12, 481-500.

[37] Hoshiya, M., 1993, “Conditional Simulation of a Stochastic Field,” Proceedings of the 6th International Conference on Structural Safety and Reliability, Innsbruck, Austria. Balkema Publishers, Amsterdam, Netherlands.

[38] Hoshiya, M., 1994, “Conditional Simulation of Stochastic Field,” Struct. Safety and Rel., G.I. Schueller, M. Shinozuka, and J.T.P. Yao, eds.Vol. 1, 349-353.

[39] Hoshiya, M., 1995, “Kriging and Conditional Simulation of a Gaussian Field,” J. Eng. Mech, ASCE, ASCE, 121(2), 181-186.

[40] Hoshiya, M. and Maruyama, O., 1993, “Stochastic Interpolation of Earthquake wave Propagation,” ICOSSAR ‘93.

[41] Hoshiya M. and Yoshida, I., 1994, “Identification of Conditional Stochastic Gaussian Field,” Source unidentified.

[42] Hudspeth, R.T. and Chen, M.C., 1979, “Digital simulation of nonlinear random waves,” J. waterways, port, coastal and ocean division, ASCE. 105: 67-85.

[43] Iyengar, R.N., and Jaiswal, O.R., 1993, “A new model for non-Gaussian random excitations,” Probabilistic Engineering Mechanics, 8: 281-287.

[44] Jaynes, E.T., 1957, “Information Theory and Statistical Mechanics,” Phys. Rev., 106, 620-630.

[45] Jefferys, E.R., and Rainey, R.C.T., 1994, “Slender Body Models of TLP and GBS Ringing,” *BOSS*.

[46] Johnson, G.E., 1994, “Constructions of Particular Random Process,” Proceedings of the IEEE, 82(2): 270-285.

[47] Journel, A.G. and Huijbregts, C.J., 1989, Mining Geostatistics, Academic Press, London, England.

[48] Kameda, H. and Morikawa, H., 1992, “An Interpolating Stochastic Process for Simulation of Conditional Random Fields,” Prob. Eng. Mech. 7(4), 242-254.

[49] Kameda, H. and Morikawa, H., 1993, “Conditioned Stochastic Processes for Conditional Random Fields,” J. Eng. Mech., ASCE, 120(4), 855-875.

[50] Kareem, A., 1993, “Numerical Simulation of Stochastic Wind effects,” Proceedings of the 7th U.S. National Conference on Wind Engineering, Vol. 1, Wind Engineering Research Council, LA, California.

[51] Kareem, A., 1995, “The Next Generation of Tuned Liquid Dampers,” Proc. of the First World Conference on Structural Control, Los Angeles.

[52] Kareem, A., Gurley, K., and Kantor, J.C., 1993, “Time-scale analysis on nonstationary processes utilizing wavelet transforms,” Proc. 6th international conference on structural safety and reliability (ICOSSAR), Schueller, Shinozuka and Yao (eds.), Innsbruck, Austria, Balkema Publishers, Amsterdam, Netherlands.

[53] Kareem, A., and Hsieh, 1991, “Probabilistic dynamic response of offshore platforms to wave loads,” Technical Report No. NDCE91-1, Dept, of Civil Engineering, University of Notre Dame.

[54] Kareem, A., Hsieh, C.C., Tognarelli, M.A., 1994, “Response analysis of offshore systems to nonlinear random waves part I: wave field characteristics,” Proceedings of the special symposium on stochastic dynamics and reliability of nonlinear ocean systems, Ibrahim and Lin (eds.), ASME, Chicago, IL.

[55] Kareem, A. and Li, Y., 1988, “On modelling the nonlinear relationship between random fields by means of higher-order spectra,” Probabilistic Methods in Civil Engineering (P.D. Spanos, editor), ASCE, NY, 384-387.

[56] Kareem, A. and Li, Y., 1994, “Stochastic Response of a Tension Leg Platform to Viscous and Potential Drift Forces,” Probabilistic Engineering Mechanics, 9, 1-14.

[57] Kareem, A. and Murikawa, 1994, “Conditional Simulation of Random Signals With Applications in Wind Engineering,” Technical Note, NDCE 94.

[58] Kareem, A. and Murlidharan, T., 1993, “Conditional Simulation of Gaussian Velocity Field,” Tech. Report, Department of Civil Engineering and Geological Sciences, University of Notre Dame.

[59] Kozin, F., 1988, “Auto-Regressive Moving-Average Models of Earthquake Records,” J. Prob. Eng. Mech., 3(2), 58-63.

[60] Krige, D.G., 1966, “Two-Dimensional Weighted Moving Average Trend Surfaces for Ore Evaluation,” J. South African Inst. of Mining and Metallurgy, Proc. Symposium on Mathematical Statistics and Computer Applications for Ore Evaluation, Johannesburg, South Africa, 13-38.

[61] Kung, S.Y., 1993, Digital Neural Networks, PTR Prentice Hall, Englewood Cliffs, New Jersey.

[62] Lawrence, A.J. and Lewis, P.A.W., 1985, “Modelling and Residual Analysis of Nonlinear Autoregressive Time Series in Exponential Variables,” Journal of the Royal Statistical Society, Series B, 47(2), 162-202.

[63] Li, Y. and Kareem, A., 1990, “ARMA Systems in Wind Engineering,” Probabilistic Engineering Mechanics, 5(2), 50-59.

[64]Li, Y. and Kareem, A., 1990, “Recursive Modeling of Dynamic Systems,” Journal of Engineering Mechanics, 116(3) 660-679.

[65] Li. Y., and Kareem, A., 1991, “Simulation of multivariate nonstationary random processes by FFT,” Journal of engineering mechanics, ASCE. 117: 1037-1058.

[66] Li, Y. and Kareem, A., 1993, “Parametric Modelling of Stochastic Wave Effects on Offshore Platforms,” Applied Ocean Research, 15(2), 63-83.

[67] Li. Y., and Kareem, A., 1993, “Simulation of multivariate random processes: Hybrid DFT and digital filtering approach,” Journal of engineering mechanics, ASCE. 119: 1078-98.

[68] Li, Y. and Kareem, A., 1997, “Simulation of Multivariate Nonstationary Random Processes: Hybrid DFT and Digital Filtering Approach,” to appear, Journal of Engineering Mechanics.

[69] Liu, W.K., Mani, A., and Belytschko, T., 1987, “Finite Element Methods in Probabilistic Mechanics,” Probabilistic Engrg. Mech., 2(4), 201-213.

[70] Mignolet, M.P., and Spanos, P.D., 1987, “Recursive simulation of stationary multivariate random processes,” Journal of applied mechanics, 54: 674-87.

[71] Ochi, M.K., 1986, “Non-Gaussian random processes in ocean engineering,” Probabilistic Engineering Mechanics, 1: 28-39.

[72] Pandey, M.D. and Ariaratnam, S.T., 1996, “Crossing Rate Analysis of Non-Gaussian Response of Linear Systems,” J. Eng. Mech. 122(6), 507-511.

[73] Papoulis, A., 1984, Probability, Random Variables, and Stochastic Processes, McGraw-Hill, Inc.

[74] Poirion, F., 1993, “Numerical Simulation of Homogeneous Non-Gaussian Random Vector Fields,” J. Snd. Vib., 160(1), 25-42.

[75] Priestly, M.B. (1967), “Power Spectral Analysis of Non-Stationary Random Processes,” Journal of Sound and Vibration, 6(1): 86-97.

[76] Samaras, E., Shinozuka, M. and Tsurui, A., 1985, “ARMA Representation of Random Processes,” J. Eng. Mech., ASCE, 111(3), 449-461.

[77] Schetzen, M., 1980, The Volterra and Wiener Theories of Nonlinear Systems, John Wiley & Sons.

[78] Seong, S.H. and Peterka, J.A., 1993, “Computer Simulation of Non-Gaussian Wind Pressure Fluctuations,” Proceedings of the seventh U.S. National Conference on Wind Engineering, 2: 623-632.

[79] Shinozuka, M., 1971, “Simulation of Multivariate and Multidimensional Random Processes,” J. of Acoust. Soc. Am., 49: 357-368.

[80] Shinozuka, M., 1972, “Monte Carlo Solution of Structural Dynamics,” Computers and Structures, 2(5-6), 855-874.

[81] Shinozuka, M., 1974, “Digital Simulation of Random Processes in Engineering Mechanics with the Aid of the FFT technique,” Stochastic Problems in Mechanics, (Eds. S.T. Ariaratnam, H.H.E. Leipholz), University of Waterloo Press, Waterloo, 277-286.

[82] Shinozuka, M., 1987, “Stochastic Fields and their Digital Simulation,” Stochastic Methods in Structural Dynamics, (Eds. G.I. Schueller and M. Shinozuka), Martinus Hijhoff Publishers, Dordrecht, 93-133.

[83] Shinozuka, M. and Deodatis, G., 1988, “Stochastic Process Models for Earthquake Ground Motion,” J. Prob. Eng. Mech., 3(3), 114-123.

[84] Shinozuka, M. and Deodatis, G., 1991, “Simulation of Stochastic Processes by Spectral Representation,” Applied Mechanics Review, 44(4), 191-204.

[85] Shinozuka, M. and Deodatis, G., 1996, “Simulation of Multi-Dimensional Gaussian Stochastic Fields by Spectral Representation,” Appl. Mech. Rev. ASME, 49(1), 29-53.

[86] Shinozuka, M. and Jan, C-M., 1972, “Digital Simulation of Random Processes and its Applications,” J. Snd. Vib., 25(1), 111-128.

[87] Shinozuka, M. and Zhang, R., 1996, “Equivalence between Kriging and CPDF methods for conditional Simulation,” J. of Eng. Mech. ASCE, Vol. 122, No. 6, pp. 530-538.

[88] Siddall, J.N., 1983, Probabilistic Engineering Design, Marcel Dekker, Inc.

[89] Sobczyk, K., 1991, Stochastic Differential Equations, with Applications to Physics and Engineering. Kluwer Academic Publishers, London.

[90] Sobczyk, K., and Trebicki, J., 1990, “Maximum Entropy Principle in Stochastic Dynamics,” Probabilistic Engineering Mechanics, 5(3): 102-110.

[91] Sobczyk, K., and Trebicki, J., 1993, “Maximum Entropy Principle and Nonlinear Stochastic Oscillators,” Physica A, 193, 448-468.

[92] Soong, T.T. and Grigoriu, M., 1993, Random Vibration of Mechanical and Structural Systems, Englewood Cliffs. N.J., Prentice Hall.

[93] Spanos, P.D. and Donley, M.G., 1991, “Equivalent statistical quadratization for nonlinear systems,” Journal of Engineering Mechanics, *ASCE*, 117(6): 1289-1309.

[94] Tagliani, A., 1990, “On the Existence of Maximum Entropy Distributions with Four and More Assigned Moments,” Prob. Eng. Mech., 5(4), 167-170.

[95] Tick, L.J., 1963, “Nonlinear probability models of ocean waves,” Ocean wave spectra. Prentice-Hall, Inc., Englewood Cliffs, N.J., pp. 163-169.

[96] Tognarelli, M.A., Zhao, J. and Kareem, A., 1997, “Equivalent statistical cubicization: A frequency domain approach for nonlinearities in both system and forcing function,” Journal of Engineering Mechanics, ASCE, Vol. 123, No.5.

[97] Vanmarcke, E.H. and Fenton, G.A., 1991, “Conditional Simulation of Local Fields of Earthquake Ground Motion,” Struct. Safety. 10(1-3), 247-263.

[98] Winterstein, S.R., 1988, “Nonlinear vibration models for extremes and fatigue,” Journal of Engineering Mechanics, ASCE, 114(10): 1772-1790.

[99] Winterstein, S.R. and Lange, C.H., 1995, “Moment-Based Probability Models for Wind Engineering Applications,” Engineering Mechanics, Proceedings of the 10th Conference, ASCE, 159-161.

[100] Wittig, L.E. and Sinha, A.K., 1975, “Simulation of Multicorrelated Random Processes Using the FFT Algorithm,” J. Acoust. Soc. Am., 58(3), 630-634.

[101]Yamazaki, F. and Shinozuka, M., 1988, “Digital Generation of Non-Gaussian Stochastic Fields,” J. Eng. Mech., 114(7), 1183-1197.

[102]Yamazaki, F. and Shinozuka, M., 1990, “Simulation of Stochastic Fields by Statistical Preconditioning,” J. Eng. Mech., 116(2), 268-287.

[103]Yang, J.N., 1972, “Simulation of Random Envelope Processes,” J. Snd. Vib., 25(1), 73-85.