



ELSEVIER

Probabilistic Engineering Mechanics 17 (2002) 293–303

PROBABILISTIC
ENGINEERING
MECHANICS

www.elsevier.com/locate/probengmech

Implementation of Karhunen–Loeve expansion for simulation using a wavelet-Galerkin scheme

K.K. Phoon*, S.P. Huang, S.T. Quek

Department of Civil Engineering, National University of Singapore, Block E1A #07-03, 1 Engineering Drive 2, Singapore, Singapore 117576

Abstract

The feasibility of implementing Karhunen–Loeve (K–L) expansion as a practical simulation tool hinges crucially on the ability to compute a large number of K–L terms accurately and cheaply. This study presents a simple wavelet-Galerkin approach to solve the Fredholm integral equation for K–L simulation. The proposed method has significant computational advantages over the conventional Galerkin method. Wavelet bases provide localized compact support, which lead to sparse representations of functions and integral operators. Existing efficient numerical scheme to obtain wavelet coefficients and inverse wavelet transforms can be taken advantage of solving the integral equation. The computational efficiency of the wavelet-Galerkin method is illustrated using two stationary covariance functions (exponential and squared exponential) and one non-stationary covariance function (Wiener–Levy). The ability of the wavelet-Galerkin approach to compute a large number of eigensolutions accurately and cheaply can be exploited to great advantage in implementing the K–L expansion for practical simulation. © 2002 Elsevier Science Ltd. All rights reserved.

Keywords: Karhunen–Loeve expansion; simulation; Fredholm integral equation; Wavelet-Galerkin; Discrete wavelet transform; Mallat's tree algorithm; Harr wavelets

1. Introduction

Stochastic simulation of practical problems has become increasingly attractive with the rapid advancement in computer technology. A unified procedure based on Karhunen–Loeve (K–L) expansion to simulate stationary and non-stationary, Gaussian and non-Gaussian processes have earlier been proposed by the authors [1–3]. Basically, K–L expansion provides a second-moment characterization of a random process in terms of deterministic orthogonal functions and uncorrelated random variables as follows:

$$\varpi(x, \theta) = \bar{\varpi}(x) + \sum_{k=1}^M \sqrt{\lambda_k} \xi_k(\theta) f_k(x) \quad (1)$$

where $\bar{\varpi}(x)$ is the mean of the process, λ_k and $f_k(x)$ are the eigenvalues and eigenfunctions of the covariance function $C(x_1, x_2)$, $\xi_k(\theta)$ is a set of uncorrelated random variables, and M is the number of K–L terms. The deterministic eigenfunctions $f_k(x)$ are obtained from the spectral decomposition of the covariance function. Hence, the first essential step is to solve for the eigenvalues and eigenfunctions from the homogeneous Fredholm integral equation of

the second kind given by

$$\int_D C(x_1, x_2) f_k(x_1) dx_1 = \lambda_k f_k(x_2) \quad (2)$$

The second important step is the selection of uncorrelated standardized K–L random variables such that the expansion produces the desired marginal distribution. Details are given elsewhere [3].

The efficiency of K–L expansion for simulating random processes hinges crucially on the availability of accurate eigenvalues and eigenfunctions of the covariance function. However, for most covariance functions, numerical methods such as the Galerkin, collocation or Rayleigh–Ritz methods are required, usually employing polynomial or trigonometric bases. Conventional Galerkin methods lead to dense matrices that are very costly to compute and invert. Representation of integral operators using conventional bases, such as polynomials and trigonometrics, also requires approximate integration quadratures that are tedious to evaluate. The difficulty of solving the Fredholm integral equation accurately for higher order eigenvalues and eigenfunctions is quite well known [1,4,5].

It is tempting to dismiss this difficulty as academic since high order eigenvalues are typically very small compared to the first few eigenvalues. For example, the 10th eigenvalue of an exponential covariance function is less than 1% of the

* Corresponding author. Tel.: +65-874-6783; fax: +65-779-1635.
E-mail address: cvepkk@nus.edu.sg (K.K. Phoon).

Table 1
Comparison of eigenvalues for exponential covariance function

No.	Wavelets							Analytical	Polynomial ($P = 10$)	Trigonometrics ($T = 5$)
	$N = 8$	$N = 16$	$N = 32$	$N = 64$	$N = 128$	$N = 256$	$N = 512$			
1	1.16296	1.15272	1.15016	1.14952	1.14936	1.14932	1.14935	1.14935	1.14934	1.14929
2	0.40423	0.39423	0.39176	0.39115	0.39095	0.39095	0.39094	0.39094	0.39094	0.36828
3	0.16926	0.16003	0.15779	0.15723	0.15709	0.15706	0.15706	0.15706	0.15991	0.15704
4	0.09157	0.08240	0.08026	0.07973	0.07957	0.07957	0.07956	0.07956	0.07956	0.07320
5	0.05950	0.04993	0.04781	0.04729	0.04716	0.04714	0.04713	0.04713	0.05004	0.04712
6	0.04408	0.03375	0.03161	0.03175	0.03189	0.03094	0.03193	0.03190	0.03093	0.02821
7	0.03614	0.02464	0.02246	0.02195	0.02182	0.02180	0.02179	0.02179	0.02458	0.02178
8	0.03226	0.01906	0.01682	0.01631	0.01619	0.01615	0.01615	0.01641	0.01598	0.01455
9		0.01543	0.01311	0.01260	0.01247	0.01244	0.01243	0.01243	0.011601	0.01243
10		0.01296	0.01054	0.01003	0.00990	0.00987	0.00986	0.00996	0.008074	0.00800

first eigenvalue (Table 1). It is not unreasonable to assume that a short K–L expansion should suffice for practical applications. Studies involving application of K–L expansion in stochastic finite element method seem to justify such a commonsense point of view [4–6]. A direct examination of the K–L expansion using simulation will however reveal that this is not true in general [1].

For processes with non-smooth covariance functions and long weakly correlated processes, ‘small’ high order eigenvalues cannot be neglected without having a very serious impact on the accuracy of the simulation. Many K–L terms are needed to even reproduce the target variance correctly for these classes of problems. This may partially explain why the K–L expansion has not been widely used for simulation despite its theoretical importance and its obvious advantage of providing an elegant unified framework for both stationary and non-stationary processes. The feasibility of implementing K–L expansion as a practical simulation tool therefore hinges crucially on the ability to compute a large number of K–L terms accurately and cheaply.

This study proposes the wavelet-Galerkin approach to solve Fredholm integral equation to overcome the shortcomings discussed earlier. The proposed method has significant computational advantages over the conventional Galerkin method [7,8]. Wavelet bases provide localized compact support, which lead to sparse representations of functions and integral operators [9–12]. Existing efficient numerical schemes to obtain wavelet coefficients and inverse wavelet transforms can be taken advantage of to solve the integral equation [13,14]. The wavelet transform has been used by various authors for purposes such as representation of random fields and simulation of random processes [15–17], but it has not been applied within the context of developing K–L expansion as a practical simulation tool. The validity and convergence characteristics of the wavelet-Galerkin method for solving the Fredholm integral equation will be illustrated by numerical examples. Results based on the conventional Galerkin method with polynomial bases and trigonometric bases will

be presented for comparison. The ability of the wavelet-Galerkin approach to compute a large number of eigensolutions accurately and rapidly can be exploited to great advantage in implementing the K–L expansion for practical simulation.

2. Method

Wavelets are receiving increased attention as a natural basis with multilevel schemes for solving partial differential equations and integral equations. The Galerkin approach involves approximating each eigenfunction as a linear combination of some basis functions and setting the error in the Fredholm integral equation (Eq. (2)) to be orthogonal to each basis function to obtain a finite order system of linear algebraic equations. The advantage of Galerkin method over integration-formulae based numerical methods (such as quadrature method) is that the complexity and accuracy of the numerical solution is controlled not only by the number of the terms included in the expansion but also the shape of the approximating functions. The choice of basis function for implementing the Galerkin method depends on the application. However, the conventional Galerkin approach leads to a dense matrix due to the global nature of integral equations. For large-scale problem, the solution procedure of the dense matrix equation is prohibitively slow.

The wavelet-Galerkin approach is chosen to overcome this problem. Wavelet basis functions have a number of attractive properties that are well suited for Galerkin solution of integral equations. Wavelet bases provide localized compact support and can generate localization properties in both space and frequency [18]. The use of compactly supported wavelets as basis functions generates strong de-correlation among the expansion coefficients and hence weakens the global coupling effects in the integral equation. This leads to sparse representations of functions and integral operators if the kernel of the integral equation is a rapidly decreasing function of its argument, i.e. if it satisfies the Calderon–Zygmund conditions [9–11,19]. The

solution procedure for sparse matrix equation is much faster than that for dense matrix equation. Existing efficient numerical scheme to obtain wavelet coefficients and inverse wavelet transforms can be taken advantage of solving the integral equation [11,18]. Furthermore, the localized nature and multi-level representation in wavelets imply that the domain of integration can be discretized at different levels or scales. This ability to represent functions at different levels of resolution offers logical means to develop a hierarchy of solutions and an adaptive scheme [13]. This is useful for computing high-order eigenfunctions, where the resolution must be high to capture the rapid fluctuations.

Representation of integral operators using conventional bases, such as polynomials and trigonometrics, requires approximate integration quadratures. The order of convergence of the quadratures determines the computational complexity of the numerical method as a whole. Representation of integral operators in wavelet basis can be performed without numerical integration. For the wavelet-Galerkin scheme presented in this study, replacement of tedious quadratures by the wavelet transform translates into significant cost savings. This important issue will be discussed in Section 2.2 and demonstrated numerically in Section 3.

One difficulty of using wavelets bases for representation of integral operators is that they do not form a basis for functions on a finite interval. Wavelet basis functions overlap in such a way that a periodization must be performed or the basis functions at the boundaries must be modified [9,20,21]. Wavelet properties rely on the assumption that the domain of the integral equation is periodic (which is equivalent to the assumption that the domain is unbounded). Orthogonality, for example, is lost when the basis functions are truncated at the boundary because the domain of integration is a finite interval. This paper adopts the simple approach of treating the integrand as a periodic function. This allows periodic wavelets to be used [22,23]. Other schemes of constructing suitable wavelets for bounded domains are described elsewhere [24–26].

2.1. Orthogonal bases for Galerkin method

The numerical solution of Fredholm integral equation using the Galerkin method is briefly summarized below [1].

(a) Select N basis functions, say $\psi_i, i = 0, \dots, N - 1$

For polynomial basis function, Legendre polynomials up to $P = (N - 1)$ th degree on the interval $[-1, 1]$ are defined as:

$$\begin{aligned} \phi_0 &= 1, & \phi_1 &= x, \dots, \phi_i &= \frac{2i-1}{i}x\phi_{i-1} - \frac{i-1}{i}\phi_{i-2}, \\ i &= 2, \dots, P \end{aligned} \tag{3}$$

which can be normalized and transformed to the interval

$[-a, a]$ by

$$\psi_i(x) = \sqrt{\frac{2i+1}{2a}} \phi_i\left(\frac{x}{a}\right) \tag{4}$$

Numerical solutions obtained using this set of basis functions will be referred to as the P th polynomial solutions, where $P = N - 1$.

For trigonometric basis function, functions up to $(N - 1)/2$ harmonics on the interval $[-a, a]$ are defined as:

$$\begin{aligned} \psi_0 &= 1, & \psi_1 &= \cos\left(\frac{\pi x}{a}\right), \\ \psi_2 &= \sin\left(\frac{\pi x}{a}\right), \dots, \psi_{2i-1} &= \cos\left(\frac{i\pi x}{a}\right), \end{aligned} \tag{5}$$

$$\psi_{2i} = \sin\left(\frac{i\pi x}{a}\right), \quad i = 2, \dots, T$$

Numerical solutions obtained using this set of basis functions will be referred to as the T harmonics solutions, where $T = (N - 1)/2$. For wavelet basis function, Haar wavelets are used as discussed in Section 2.2.

(b) Approximate each eigenfunction as a linear combination of these basis functions:

$$f_k(x) = \sum_{i=0}^{N-1} d_i^{(k)} \psi_i(x) = \Psi^T(x) D^{(k)} \tag{6}$$

where $D^{(k)}$ is a $N \times 1$ coefficient vector and $\Psi(x)$ is the vector of basis functions.

(c) Substitute Eq. (6) into Eq. (2) and set the error in Eq. (2) to be orthogonal to each basis function to obtain the following equation:

$$\begin{aligned} \sum_{i=0}^{N-1} d_i^{(k)} \left[\int_D \int_D C(x_1, x_2) \psi_i(x_1) \psi_j(x_2) dx_1 dx_2 \right] \\ - \lambda_k \sum_{i=0}^{N-1} d_i^{(k)} \left[\int_D \psi_i(x_2) \psi_j(x_2) dx_2 \right] = 0 \end{aligned} \tag{7}$$

or

$$AD^{(k)} = \lambda_k BD^{(k)} \tag{8}$$

where the components in the $N \times N$ matrices are

$$A_{ji} = \int_D \int_D C(x_1, x_2) \psi_i(x_1) \psi_j(x_2) dx_1 dx_2 \tag{9}$$

$$B_{ji} = \int_D \psi_i(x_2) \psi_j(x_2) dx_2 \tag{10}$$

The integrals in Eqs. (9) and (10) are evaluated numerically using the function D01FCF in MATLAB toolbox NAG [27], which is based on multi-dimensional adaptive quadrature over a hyper-rectangle (i.e. with constant limits). The routine returns an estimate of a multi-dimensional integral, and also an estimate of the relative error. The user sets the relative accuracy required and supplies the

integrand as a function subprogram. In this study, the relative accuracy is set to 10^{-4} .

The wavelet-Galerkin method is based on the same general framework except that: (a) basis functions ψ_i are wavelets and (b) integral in Eq. (9) is 2D wavelet transform that does not require quadrature evaluation.

2.2. Wavelet-Galerkin scheme using Haar wavelets

In this study, a simple but efficient and accurate wavelet-Galerkin scheme is proposed. The simplest form of Daubechies' wavelets, Haar wavelets, is selected as basis functions. Haar wavelets are generated from the following Haar mother wavelet function:

$$\psi(x) = \begin{cases} 1 & x \in [0, 1/2); \\ -1 & x \in [1/2, 1); \\ 0 & \text{otherwise.} \end{cases} \quad (11)$$

A family of orthogonal Haar wavelets can be generated by shifting and scaling the mother wavelet:

$$\psi_{j,k}(x) = \alpha_j \psi(2^j x - k) \quad j, k \in \mathbb{Z} \quad (12)$$

where j and k are the dilation and translational (positive) integer constants and α_j is the amplitude of the function. When $\alpha_j = 2^{j/2}$, a family of orthonormal Haar wavelets is obtained. In this study, α_j is taken as 1, and $\psi_{j,k}$ is a set of orthogonal functions with unit amplitude. It has been proven that the wavelets form a basis for $L^2(\mathbb{R})$ [14,23].

Unlike other wavelets in the Daubechies family, Haar wavelets at a given resolution do not overlap which greatly simplifies numerical computation. However, Haar wavelets defined over a bounded domain do not form a set of complete basis functions [23]. To overcome this, consider a bounded domain $[0, 1]$ and define a set of basis function over the domain as

$$\psi_0(x) = 1 \quad (13)$$

$$\psi_i(x) = \psi_{j,k}(x) \quad (14)$$

$$i = 2^j + k; \quad k = 0, 1, \dots, 2^j - 1; \quad j = 0, 1, \dots, m - 1$$

where m is the maximum wavelet level. The extended set $\psi_i(x)$ forms a complete set of orthogonal functions over the domain $[0, 1]$ and the orthogonality condition can be written as

$$\int_0^1 \psi_i(x) \psi_j(x) dx = h_i \delta_{ij} \quad (15)$$

where h_i is a constant and δ_{ij} is Kronecker-delta function.

To solve Eq. (2) numerically, the eigenfunction $f_k(x)$ is approximated as a truncated series of Haar wavelets, written as

$$f_k(x) = \sum_{i=0}^{N-1} d_i^{(k)} \psi_i(x) = \Psi^T(x) D^{(k)} \quad (16)$$

where $N = 2^m$ and $d_i^{(k)}$ are the wavelet coefficients. The covariance function $C(x_1, x_2)$ can be expressed as [23]

$$C(x_1, x_2) = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} \bar{A}_{ij} \psi_i(x_1) \psi_j(x_2) = \Psi^T(x_1) \bar{A} \Psi(x_2) \quad (17)$$

where \bar{A} is an $N \times N$ matrix which is a 2D wavelet transform of $C(x_1, x_2)$ given by

$$\bar{A}_{ij} = \frac{1}{h_i h_j} \int_0^1 \int_0^1 C(x_1, x_2) \psi_i(x_2) \psi_j(x_1) dx_1 dx_2 \quad (18)$$

The double integral is the 2D wavelet transform of the function $C(x_1, x_2)$, which is similar to Eq. (9) for conventional bases. However, the crucial difference is that numerical integration is not required. To perform 2D wavelet transform, the 1D wavelet transform is applied first on the rows and then on the columns of the matrix containing values of $C(x_1, x_2)$ evaluated over a N by N grid. In this paper, it is assumed that the number of discrete points used in the 2D wavelet transform of $C(x_1, x_2)$ (Eq. (17)) is the same as the number of wavelets used to represent $f_k(x)$ (Eq. (16)). It is possible to use more discrete points, which is equivalent to neglecting high frequency wavelets and taking average values in small regions, if the improvement in accuracy can offset the increase in runtime for wavelet transform.

To perform 1D wavelet transform, consider a set of values F_i ($i = 0, 1, 2, \dots, N - 1$) of the function $F(x)$ calculated at the discrete points $x_i = (2i + 1)/(2N)$. A $N \times 1$ vector is first initialized as follows:

$$a_{m,k} = F_k \quad k = 0, 1, 2, \dots, N - 1 \quad (19)$$

The vector is processed using an inverse binary tree, where the topmost (m th) layer contains N nodes with values given by Eq. (19). The nodal values in subsequent layers are computed as:

$$a_{j,k} = \frac{1}{2} (a_{j+1,2k} + a_{j+1,2k+1}) \quad (20)$$

where $k = 0, 1, 2, \dots, 2^j - 1$ and $j = m - 1, \dots, 2, 1, 0$. The wavelet coefficients are evaluated from the nodal values in this binary tree as

$$c_{j,k} = \frac{1}{2} (a_{j+1,2k+1} - a_{j+1,2k}) \quad (21)$$

Finally, the 1D wavelet transform is $[a_{0,0}, c_1, \dots, c_{N-1}]$, where

$$c_i = c_{j,k} \quad (22)$$

$$i = 2^j + k; \quad k = 0, 1, \dots, 2^j - 1; \quad j = 0, 1, \dots, m - 1$$

Details are given elsewhere [11,28]. The matrix \bar{A} can be made sparse by neglecting elements below a threshold value such as 10^{-4} [11,28]. This important feature of wavelet transform has been exploited in image processing [29].

Other Daubechies wavelets can be used by replacing Eqs. (20) and (21) with their respective wavelet transform equations. However, Eq. (17) indicates that 2D wavelet

transform is a similarity transform because $\Psi(x)$ is orthogonal. Hence, the same eigenvalues are obtained at the same resolution (same m), regardless of $\Psi(x)$. The use of a different orthogonal wavelet basis however will affect the sparsity pattern of \bar{A} , which can lead to more efficient eigensolution. This important issue currently is under study.

The discrete wavelet transform based on Mallat’s tree algorithm, as outlined in Eqs. (20) and (21) for Harr wavelets, is extremely efficient computationally. The operation count is approximately N for Harr wavelets, which compares favorably with $N \log_2 N$ in FFT [23]. Another advantage is that a variety of wavelet basis functions can be exploited to optimize matrix sparsity for different covariance functions. In contrast, FFT only works for trigonometric basis functions and the transformed matrix generally is dense.

Substituting Eqs. (16)–(18) into Eq. (2) leads to a finite dimensional eigenvalue problem of the form

$$\lambda_k \Psi^T(x) D^{(k)} = \Psi^T(x) \bar{A} H D^{(k)} \quad (23)$$

where H is a diagonal matrix as

$$H = \begin{bmatrix} h_0 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & h_{N-1} \end{bmatrix} \quad (24)$$

H matrix can be calculated easily without integration because $\psi_i(x)$ is constant in the domain of integration. The value of h_i defined in Eq. (15) is the area of the step function that can be computed by multiplying the amplitude (constant 1) with the interval (2^{-j}). Hence, $h_0 = 1, \dots, h_i = 2^{-j}$, for $i = 2^j + k$; $k = 0, 1, \dots, 2^j - 1$; $j = 0, 1, \dots, m - 1$. Denoting $\hat{D}^{(k)} = H^{1/2} D^{(k)}$, $\hat{A} = H^{1/2} \bar{A} H^{1/2}$, and equating coefficient of $\Psi(x)$, Eq. (23) can be re-written as

$$\lambda_k \hat{D}^{(k)} = \hat{A} \hat{D}^{(k)} \quad (25)$$

which can be solved using an eigenvalue solver. Once the eigenvalues and eigenvectors are known, the eigenfunctions can be obtained by taking the inverse wavelet transform of D . The eigenfunction can be written as

$$f_k(x) = \Psi^T(x) H^{-1/2} \hat{D}^{(k)}. \quad (26)$$

3. Numerical results

To illustrate the validity and merits of the wavelet-Galerkin scheme for solving the Fredholm integral equation and its importance to practical K–L simulation, numerical experiments have been performed on a standard IBM Pentium III, 750 MHz machine. The computer program has been coded in MATLAB using the procedures noted above. The wavelet-Galerkin scheme described in Section 2.2 involves computation of \bar{A} matrix in Eq. (18), solution of eigenvalue problem of Eq. (23), and inverse wavelet

transform of the computed eigenvectors (Eq. (26)). This scheme has been applied to a number of covariance kernels and three representative examples are presented in this section. The performance of the wavelet-Galerkin scheme will also be compared with conventional Galerkin method using polynomial bases and trigonometric bases. The examples considered are:

Example 1. A first-order Markov process defined in $[-a, a]$ with exponential covariance function $C(x_1, x_2) = e^{-|x_1 - x_2|/b}$, where b is the correlation parameter, is considered. In this example, $a = b = 1$.

Example 2. Consider the random process in $[-a, a]$ with squared exponential covariance function $e^{-|x_1 - x_2|^2/b}$ where b is the correlation parameter. In this example, $a = b = 1$. As discussed in Ref. [1], the squared exponential covariance function is smooth in the sense that it is differentiable everywhere. In contrast, the exponential covariance function in Example 1 is non-differentiable at the origin and this feature poses considerable difficulties for K–L expansion [1].

Example 3. Consider the Wiener–Levy process in $[0, a]$ with covariance function $C(x_1, x_2) = \min(x_1, x_2)$. This example is provided to demonstrate that the K–L expansion can provide an efficient and unified simulation tool for both stationary and non-stationary processes if the proposed wavelet-Galerkin scheme is applied.

The performance of the wavelet-Galerkin solution is studied in terms of the accuracy of the eigensolution, rate of convergence, and computational cost (CPU time). Comparisons with conventional Galerkin methods using polynomial bases and trigonometric bases will also be made. In all the examples, seven values of N (number of wavelets) are considered, namely, $N = 8, 16, 32, 64, 128, 256$ and 512 , corresponding to wavelet levels $m = 3, 4, 5, 6, 7, 8$, and 9 , respectively.

3.1. Accuracy of wavelet-Galerkin solution

The wavelet-based eigenvalues for the exponential covariance function (Example 1) are compared with analytical solutions and conventional numerical solutions in Table 1. As to be expected, the accuracy of the eigenvalues increases with the number of wavelets N . For the first 10 eigenvalues, the wavelet solution at $N = 64$ agrees very well with the analytical solution. It is difficult to compare the wavelet-Galerkin approach with conventional Galerkin methods on an equal footing because wavelets are capable of multi-level representation. A rough comparison can be made based on the number of distinct frequencies in the set of basis functions. For example, both $N = 64$ ($m = 6$) wavelets and $N = 11$ ($T = 5$) trigonometrics have five different waveforms, although wavelet and trigonometric

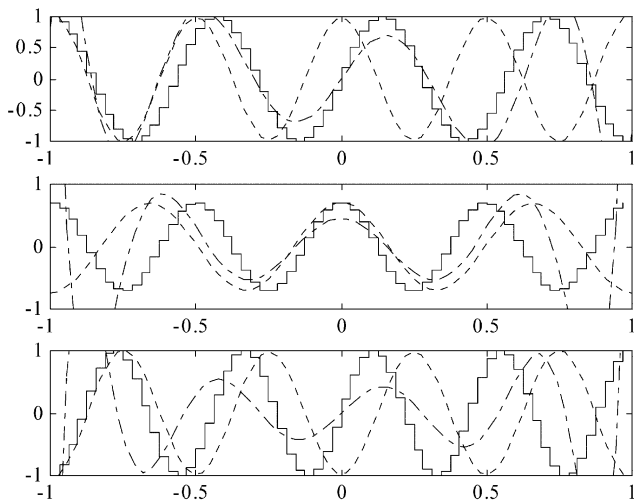


Fig. 1. Eighth- to 10th-order eigenfunctions for exponential covariance function (Example 1)—analytical solution (short dash); wavelet solution using $N = 64$ (solid); degree 10 polynomial solution (dash-dot) and trigonometrics with five harmonics (long dash).

frequencies are different. To maintain consistency between the conventional Galerkin methods in terms of size of coefficient vector D , polynomial basis functions up to degree $P = 10$ (i.e. $N = 11$) are employed. It can be seen that wavelet-based solutions are more accurate for high-order eigenvalues. This is even more evident when the eigenfunctions are compared as shown in Fig. 1. The wavelet solutions for the eighth- to 10th-order eigenfunctions approximate the analytical eigenfunctions significantly better than the polynomial (degree 10) and trigonometric (five harmonics) solutions.

Similar results for the squared exponential covariance function (Example 2) are presented in Table 2. No analytical solutions are available for this example but Fig. 2 shows that the first six eigenvalues would converge when $m \geq 7$. Hence, the wavelet solutions at the maximum level studied ($N = 512$ or $m = 9$) can be used as the benchmark solutions. Using this benchmark, it can be seen that the polynomial solutions (degree 10) are slightly more accurate than the wavelet solutions at $N = 64$ ($m = 6$) for lower

order eigenvalues in this case but comparatively less accurate for higher order eigenvalues. However, more computational time is required for polynomial solutions as discussed below. The trigonometric solutions with five harmonics are least accurate among the three. It is also interesting to note that the eigenvalues decay more rapidly for a smooth covariance function (Example 2) in contrast to one that is non-differentiable at the origin (Example 1). Hence, the number of required terms in the K–L expansion can be quite significant for Example 1 [1]. Wavelet solutions based on, say $N = 128$, would be able to furnish such a lengthy expansion easily for simulation (128 terms to be exact). Even for Example 2, lengthy K–L expansion would be needed as the domain of interest increases because eigenvalues increase in direct proportion to the length of the process [1]. Similar results are obtained for the non-stationary covariance function (Example 3) shown in Table 3. The accuracy of different eigenvalues can be presented more systematically using a measure of relative error such as $|\lambda_{\text{analytical}} - \lambda_{\text{numerical}}|/|\lambda_{\text{analytical}}|$. From Fig. 3, it can be seen that wavelet-based eigenvalues are not always the most accurate but are able to maintain good accuracy consistently over more eigenvalues.

3.2. Rate of convergence

Fig. 4 illustrates the performance of the three different solution methods for Example 1 (exponential covariance function). Both relative error (solid lines) and CPU time (dotted line) are depicted versus the maximum level of wavelet m , the number of harmonics T , and the degree of polynomial P . As expected, the relative error in eigenvalues decreases as m , P or T increases.

It is apparent that the relative error decreases more rapidly with m than P or T . However, it must be noted that the parameters m , P and T are not strictly comparable. A definite advantage of the wavelet-Galerkin approach is that less computational effort is required at a given relative error. In addition, it is also quite clear that significantly more

Table 2
Comparison of eigenvalues for squared exponential covariance function

No.	Wavelets							Polynomial ($P = 10$)	Trigonometrics ($T = 5$)
	$N = 8$	$N = 16$	$N = 32$	$N = 64$	$N = 128$	$N = 256$	$N = 512$		
1	1.30775	1.30508	1.30442	1.30425	1.30421	1.30420	1.30419	1.30419	1.30416
2	0.53783	0.53641	0.53607	0.53599	0.53596	0.53596	0.53596	0.53596	0.49743
3	0.13146	0.133335	0.1338	0.133912	0.133941	0.13395	0.13395	0.13395	0.13393
4	0.02062	0.022199	0.022584	0.022679	0.022703	0.022709	0.02271	0.02271	0.01702
5	0.00218	0.0027	0.00283	0.00286	0.00287	0.00287	0.00287	0.00287	0.00284
6	0.00016	0.00025	0.00028	0.00028	0.00029	0.00029	0.00029	0.00029	0.00014
7	6.87×10^{-6}	1.91×10^{-5}	2.29×10^{-5}	2.39×10^{-5}	2.42×10^{-5}	2.42×10^{-5}	2.42×10^{-5}	2.42×10^{-5}	2.17×10^{-5}
8	1.46×10^{-7}	1.16×10^{-6}	1.58×10^{-6}	1.7×10^{-6}	1.73×10^{-6}	1.74×10^{-6}	1.74×10^{-6}	1.73×10^{-6}	3.28×10^{-7}
9		5.8×10^{-8}	9.44×10^{-8}	1.05×10^{-7}	1.08×10^{-7}	1.09×10^{-7}	1.09×10^{-7}	1.05×10^{-7}	6.06×10^{-8}
10		2.36×10^{-9}	4.92×10^{-9}	5.77×10^{-9}	6×10^{-9}	6.05×10^{-9}	6.07×10^{-9}	3.6×10^{-8}	1.1×10^{-8}

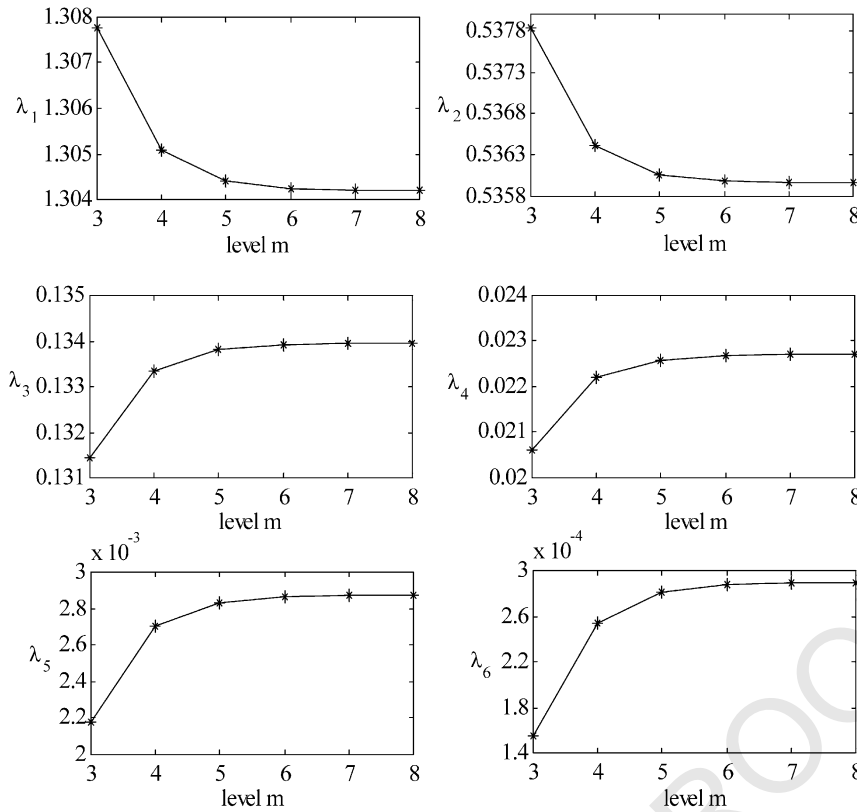


Fig. 2. Convergence of eigenvalues for squared exponential covariance function (Example 2).

eigensolutions can be obtained with less cost using the wavelet-Galerkin approach. For example, $2^7 = 128$ wavelet solutions can be obtained when $m = 7$ while only 15 trigonometric solutions can be obtained when $T = 7$. However, the former requires only 8.13 s in contrast to 210.8 s for the latter, a 26-fold increase.

As noted above, no analytical solutions are available for Example 2. Hence, $N = 512$ wavelet solutions will be considered as $\lambda_{\text{analytical}}$. Fig. 5 illustrates the performance of the three different solution methods for Example 2 (squared exponential covariance function). The results are

comparable to those shown in Fig. 4, but the advantage of the wavelet-Galerkin approach is not as significant for this smooth covariance. It can be seen that the wavelet-Galerkin approach has no apparent CPU time saving for integral equation with globally smooth covariance function, which incurs a small-scale system of equations. This can be explained by the fact that for solution of smooth covariance function, lesser number of discrete points is needed for integration (Eq. (9)) or wavelet transform (Eq. (19)). The advantage of wavelet solution is only apparent when many discrete points are involved.

Table 3
Comparison of eigenvalues for non-stationary Wiener–Levy process

No.	Wavelets							Analytical	Polynomial ($P = 10$)	Trigonometrics ($T = 5$)
	$N = 8$	$N = 16$	$N = 32$	$N = 64$	$N = 128$	$N = 256$	$N = 512$			
1	0.40659	0.40561	0.40537	0.40531	0.40529	0.40529	0.40529	0.40529	0.40515	0.39623
2	0.04636	0.04536	0.04511	0.04505	0.04504	0.04503	0.04503	0.04503	0.04486	0.04401
3	0.01758	0.01654	0.01629	0.01623	0.01622	0.01621	0.01621	0.01621	0.01592	0.01583
4	0.00971	0.00860	0.00835	0.00829	0.00828	0.00827	0.00827	0.00827	0.00732	0.00807
5	0.00654	0.00534	0.00509	0.00502	0.00501	0.00500	0.00500	0.00500	0.00274	0.00487
6	0.00502	0.00369	0.00343	0.0034	0.00336	0.00335	0.00335	0.00335	0.00121	0.00326
7	0.00427	0.00275	0.00248	0.00242	0.00240	0.00240	0.00240	0.00240	3.33×10^{-5}	0.00232
8	0.00394	0.00217	0.00189	0.00182	0.00181	0.00180	0.00180	0.00180	1.08×10^{-6}	0.00174
9		0.00178	0.00149	0.00142	0.00141	0.00140	0.00140	0.00140	8.63×10^{-7}	0.00133
10		0.00151	0.00121	0.00114	0.00113	0.00112	0.00112	0.00112	2.5×10^{-7}	0.00090

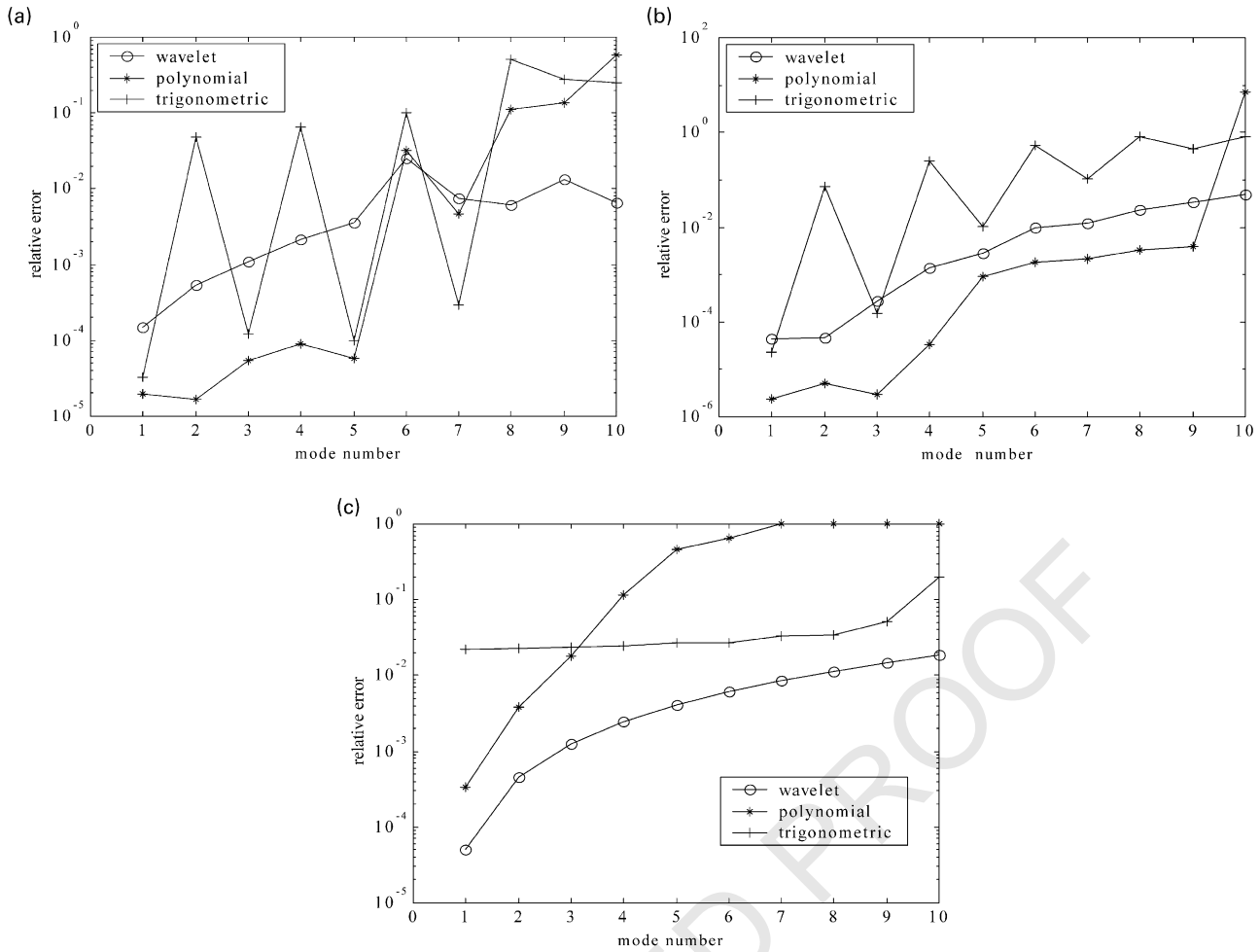


Fig. 3. Comparison of relative error in eigenvalue produced by different Galerkin approaches for (a) exponential covariance function (Example 1), (b) squared exponential covariance function (Example 2), and (c) Wiener–Levy process covariance function (Example 3).

3.3. K–L expansion for simulation

It has been shown that for processes with non-smooth covariance function and long weakly correlated processes, many terms are needed in the K–L expansion for accurate simulation [1–3]. Applying a wavelet-Galerkin scheme to implement Karhunen–Loeve expansion technique for simulation can overcome the shortcomings stated [1]. This section will show that the number of terms needed in some practical problems is easily within reach for the wavelet-Galerkin scheme but very tedious to achieve using conventional methods. The three examples considered previously are used to demonstrate the practicality of performing K–L simulation using the wavelet-Galerkin scheme. For random process with non-smooth covariance kernel, it is advantageous in two ways because both non-smooth covariance and long process entail large number of K–L terms in the expansion for simulation. For smooth covariance, the wavelet-Galerkin scheme is not advantageous for short processes that require only few K–L terms, but it is still necessary for simulation of long processes.

For Example 1, Fig. 6 shows that the covariance function does not converge to the target at the non-differentiable origin even at a fairly large value of $M = 20$. To achieve good agreement at the origin, a large number of terms, say $M = 100$, may be needed. The line for $M = 100$ terms can hardly be discerned as it almost overlaps the target line. As shown in Fig. 4b and c, it is computationally inefficient to compute beyond the first 10 terms using the conventional Galerkin approach. The time required for 100 terms will probably take hours of run-time. On the other hand, 128 wavelet-based eigensolutions can be generated in only

Table 4
Variance of K–L expansion truncated at M terms for various normalized length of the exponential process a/b (eigensolutions from wavelet-Galerkin approach with $N = 128$)

a/b	$M = 5$	$M = 20$	$M = 100$
1	0.927	0.979	0.996
2	0.851	0.958	0.992
5	0.663	0.896	0.980
10	0.458	0.782	0.961

Target variance = 1.

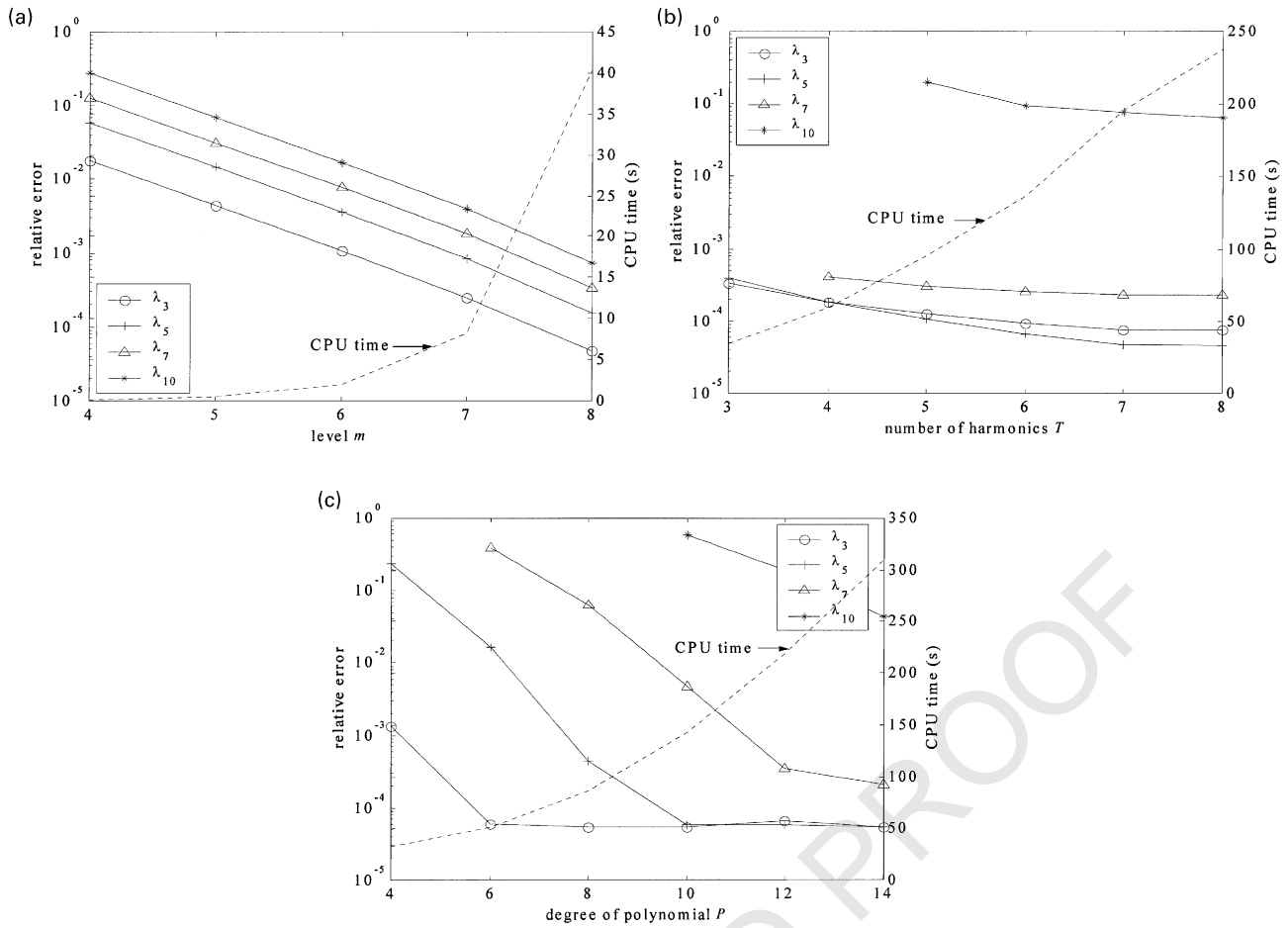


Fig. 4. Relative error in eigenvalue and CPU time versus the: (a) level of resolution of wavelet solution, (b) number of harmonics in trigonometric solution, and (c) degree of polynomial (Example 1—exponential covariance).

8.13 s. on a modest PC computing platform. Table 4 compares the computed variance using 5, 20 and 100 terms in the K–L expansion based on the wavelet-Galerkin approach for various normalized length of the process, a/b . At a fixed M , the accuracy of the simulation, as measured by the variance, clearly degrades as the length of the process ($2a$) increases with respect to the correlation parameter (b). For $a/b = 10$, about 20% of the variance is lost at $M = 20$ while less than 5% is lost at $M = 100$. Again, the ability of the wavelet-Galerkin approach to compute higher order eigen-solutions accurately and rapidly can be applied to great advantage in this instance. This is true even for the

smooth squared exponential covariance function and the non-stationary covariance function as shown in Table 5 and Fig. 7, respectively. Table 5 shows that the error in variance will eventually increase when a/b is large enough for any fixed M . Hence, the wavelet-Galerkin approach is also needed for smooth covariance kernel when long processes are encountered. Fig. 7 shows that the computed variance for Wiener–Levy process converges to the target variance at about $M = 50$. The large number of eigensolutions required would only require 1.98 s to generate using the wavelet-Galerkin approach at $m = 6$.

Table 5
Variance of K–L expansion truncated at M terms for various normalized length of the squared exponential process a/b (eigensolutions from wavelet-Galerkin approach with $N = 128$)

a/b	$M = 5$	$M = 10$	$M = 20$
1	0.999	1.000	1.000
2	0.985	1.000	1.000
5	0.718	0.966	1.000
10	0.417	0.724	0.972

Target variance = 1.

4. Conclusions

Karhunen–Loeve (K–L) expansion has not been widely used for simulation despite its theoretical importance and its obvious advantage of providing an elegant unified framework for both stationary and non-stationary processes. One reason is that there are few analytical solutions to the Fredholm integral equation and numerical solutions using conventional Galerkin methods with polynomial or trigonometric bases are computationally costly. The second reason

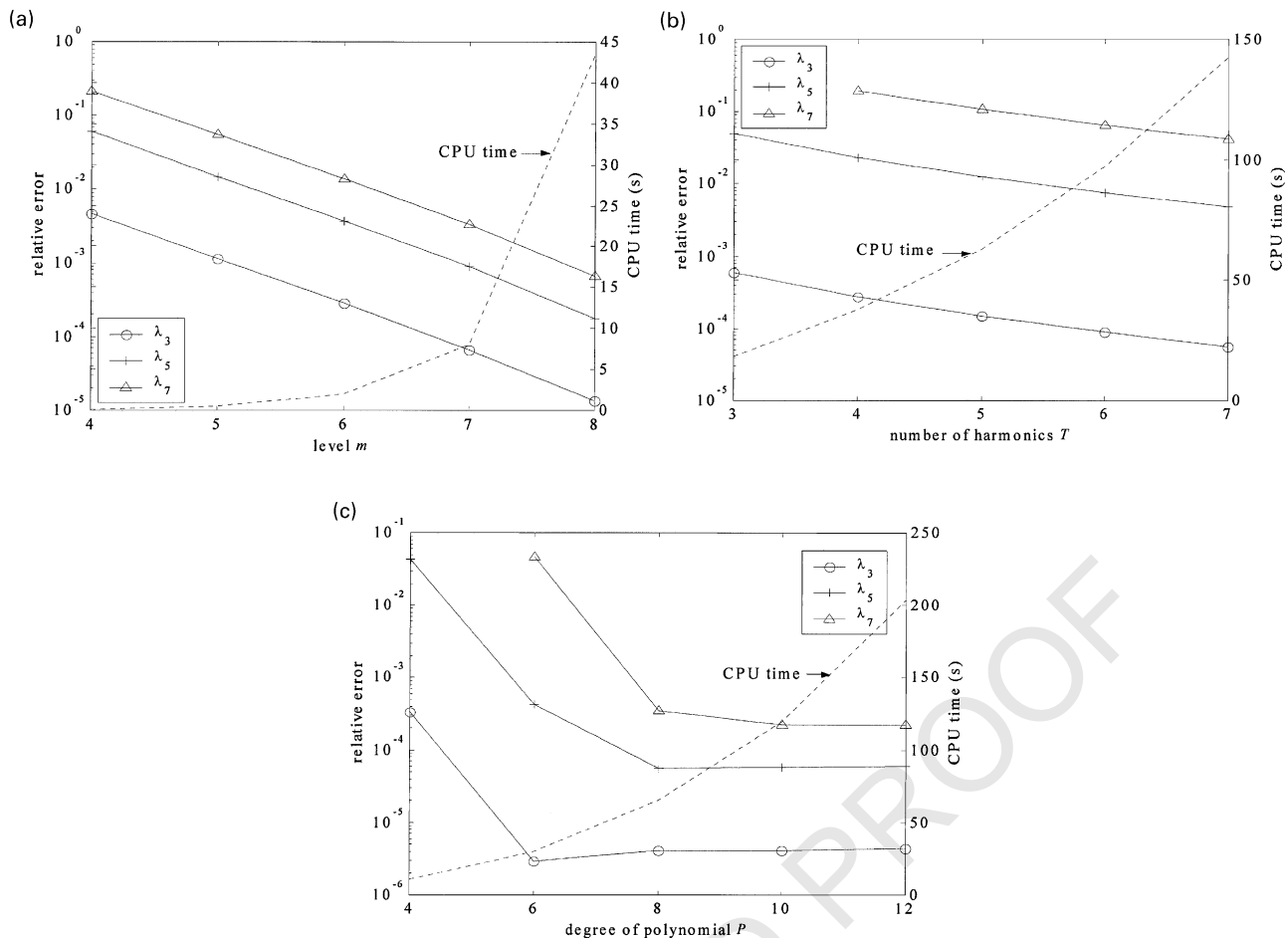


Fig. 5. Relative error in eigenvalue and CPU time versus the: (a) level of resolution of wavelet solution, (b) number of harmonics in trigonometric solution, and (c) degree of polynomial (Example 2—squared exponential covariance).

is that high order eigenvalues cannot be neglected without having a very serious impact on the accuracy of the simulation for processes with non-smooth covariance functions and long weakly correlated processes. The

feasibility of implementing K–L expansion as a practical simulation tool therefore hinges crucially on the ability to compute a large number of K–L terms accurately and cheaply.

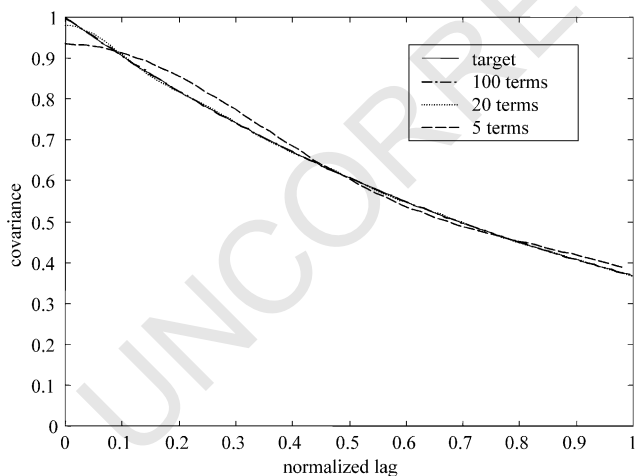


Fig. 6. Computed exponential covariance function (Example 1) using $M = 5, 20$ and 100 terms in K–L expansion (eigensolutions from wavelet-Galerkin approach using $N = 128$).

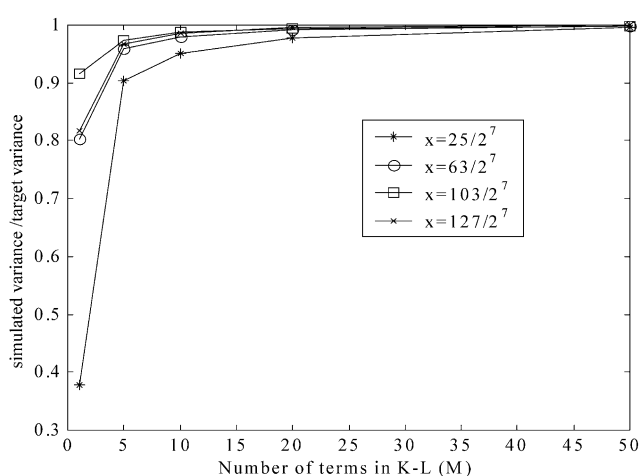


Fig. 7. Normalized variance versus number of terms in K–L at different dyadic points for non-stationary Wiener–Levy process (eigensolutions from wavelet-Galerkin approach with $N = 64$).

This study presents a simple wavelet-Galerkin approach to solve the Fredholm integral equation for K–L simulation. The proposed method has significant computational advantages over the conventional Galerkin method. Wavelet bases provide localized compact support, which lead to sparse representations of functions and integral operators. Existing efficient numerical scheme to obtain wavelet coefficients and inverse wavelet transforms can be taken advantage of solving the integral equation. The computational efficiency of the wavelet-Galerkin method is illustrated using two stationary covariance functions (exponential and squared exponential) and one non-stationary covariance function (Wiener–Levy). The ability of the wavelet-Galerkin approach to compute a large number of eigensolutions accurately and cheaply can be exploited to great advantage in implementing the K–L expansion for practical simulation.

References

- [1] Huang SP, Quek ST, Phoon KK. Convergence study of the truncated Karhunen–Loeve expansion for simulation of stochastic processes. *Int J Num Meth Engng* 2001;52(9):1029–43.
- [2] Huang SP, Phoon KK, Quek ST. Digital simulation of non-Gaussian stationary processes using Karhunen–Loeve expansion. 8th ASCE Specialty Conf Probab Mech Struct Reliab 2000;July:1–6.
- [3] Phoon KK, Huang SP, Quek ST. Simulation of second-order processes using Karhunen–Loeve expansion. *Comput Struct* (special issue on probabilistic mechanics and structural reliability) 2002; in press.
- [4] Ghanem R, Spanos PD. *Stochastic finite element: a spectral approach*. Berlin: Springer; 1991.
- [5] Zhang J, Ellingwood B. Orthogonal series expansions of random processes in reliability analysis. *J Engng Mech, ASCE* 1994;120(12):2660–77.
- [6] Schenk, C.A., Schueller, G.I. On the analysis of cylindrical shells with random imperfections. 4th International Colloquium on Computation of Shell and Spatial Structures, June 2000, Chania-Crete, Greece.
- [7] Baker CTH. *The numerical treatment of integral equations*. Oxford: Oxford University Press; 1977.
- [8] Agrawal OP, Sonti VR. Application of wavelets in modeling stochastic dynamic systems. *J Sound Vibrat* 1998;120(3):763–9.
- [9] Alpert B. A class of bases in L2 for the sparse representation of integral operators. *J Math Anal, SIAM* 1993;24:246–62.
- [10] Alpert B, Beylkin G, Coifman R, Rokhlin V. Wavelet-like bases for the fast solution of the second kind integral equations. *J Sci Comput, SIAM* 1993;14(1):159–84.
- [11] Beylkin G, Coifman R, Rokhlin V. Fast wavelet transforms and numerical algorithm I. *Commun Pure Appl Math* 1991;44(2):141–83.
- [12] Greenshields IR, Rosiene JA. Spectral decomposition by wavelet approximation to the Karhunen–Loeve transform. *Ophthal Technol II, SPIE* 1992;1644:282–6.
- [13] Mallat SG. A theory for multiresolution signal decomposition: the wavelet representation. *IEEE Trans Pattern Anal Mach Intell* 1989;11(7):674–93.
- [14] Daubechies I. Orthonormal bases of compactly supported wavelets. *Commun Pure Appl Math* 1988;41:909–96.
- [15] Zeldin BA, Spanos PD. Random field simulation using wavelet bases. *J Appl Mech, ASME* 1996;63(4):946–52.
- [16] Gurley K, Tognarelli MA, Kareem A. Analysis and simulation tools for wind engineering. *Prob Engng Mech* 1997;12(1):9–31.
- [17] Gurley K, Kareem A. Applications of wavelet transforms in wind, earthquake and ocean engineering. *Engng Struct* 1999;21(2):149–67.
- [18] Micchelli CA, Xu YS, Zhao YH. Wavelet Galerkin methods for second-kind integral equations. *J Comput Appl Math* 1997;86:251–70.
- [19] Greenshields IR, Rosiene JA. A fast wavelet-based Karhunen–Loeve transform. *Patten Recogn* 1998;31(7):839–45.
- [20] Chen MQ, Hwang C, Shin YP. The computational of wavelet-Galerkin approximation on a bounded interval. *Int J Num Meth Engng* 1996;39:2921–44.
- [21] Wang GF. Application of wavelet on the interval to numerical analysis of integral equation in electromagnetic scattering problems. *Int J Num Meth Engng* 1997;40:1–13.
- [22] Daubechies I. *Ten lectures on wavelets*. Philadelphia, PA: SIAM; 1992.
- [23] Newland DE. *An introduction to random vibrations, spectral & wavelet analysis*. New York: Wiley; 1993.
- [24] Meyer Y. *Wavelets, algorithms and applications*. Philadelphia, PA: SIAM; 1993.
- [25] Franklin P. A set of continuous orthogonal functions. *Math Ann* 1928;100:522–9.
- [26] Chui C, Quak E. *Wavelets on a bounded interval. Numerical methods of approximation theory*. Boston: Birkhauser; 1992.
- [27] Nag foundation toolbox for use with MATLAB. The Math Works Inc., 1995.
- [28] Nievergelt Y. *Wavelets made easy*. Boston: Birkhauser; 1999.
- [29] Wagner RL, Chew WC. A study of wavelets for the solution of Electromagnetic integral equations. *IEEE Trans Antennas Propag* 1995;43(8):802–10.