JOURNAL OF COMPUTATIONAL PHYSICS 96, 277–296 (1991)

# The Use of the Karhunen–Loève Procedure for the Calculation of Linear Eigenfunctions

KENNETH S. BREUER* AND LAWRENCE SIROVICH

*Center for Fluid Mechanics, Turbulence and Computation, Box 1966, Brown University, Providence, Rhode Island 02912*

Received December 19, 1989; revised October 17, 1990

It is shown that the Karhunen–Loève decomposition may be used to determine the eigenfunctions of a general class of linear operators from an ensemble of realizations that are derived from that system. Given a moderate size data set (either numerical or experimental) from a linear system, good approximations to the eigenfunctions that characterize the underlying equations can be computed by performing the Karhunen–Loève procedure. Two numerical examples are presented: the vibration of a thin membrane in a rectangular domain and in a stadium. These are used to determine the convergence and accuracy of the method. It is found that this method yields accurate results for the first few eigenfunctions with relatively few realizations. Eigenfunctions with less energy are accurately resolved as the size of the ensemble increase. The method is shown to be an efficient and practical procedure for determining the eigenfunctions of systems in complex geometries and in cases where the governing equations are not known a priori. The effect of random noise contamination of the data set is also investigated and it is found that the Karhunen–Loève procedure can still achieve accurate results despite the presence of substantial background noise. 

## 1. INTRODUCTION

The determination of the eigenvalues and eigenfunctions of a linear operator is fundamental in physics and the mathematical sciences. Analytically, the determination of the eigenfunctions is restricted to the simplest of geometries and the most elementary of operators. Consequently, for the vast majority of non-trivial problems, one must resort to numerical approximations of the eigenfunctions associated with the equation of interest. However, even with the powerful computational resources available, the eigenvalue problem can be extremely difficult. Often the geometry of the system is complex, requiring high resolution, which in turn demands large computers for solution. A second case in which the determination of the eigenfunctions is all but impossible is when the governing equations of motion are unknown, but for which experimental data from the system in question are

* Present address: Department of Aeronautics and Astronautics, Room 33-214c, MIT, Cambridge, MA 02139.



0021-9991/91 $3.00

available. One such example might be the response of a complex structure to external vibrations.

In this paper, we develop a method by which the eigenfunctions from a linear system may be computed from a set of data generated by that system. This method is based on the Karhunen–Loève (K–L) decomposition by which any ensemble of data may be decomposed into an optimal set of *empirical eigenfunctions* which are orthogonal and achieve the maximal capture of the energy of the system in the minimum number of eigenfunctions. We shall demonstrate that this procedure can be used to reproduce the analytic eigenfunctions for a general class of linear systems. The method is shown to be computationally efficient—involving far fewer operations than the QR algorithm which is often employed in the solution of the eigenvalue problem [1]. It should be noted that the K–L procedure is usually associated with nonlinear and chaotic behavior (cf. [2–4]). Thus our presentation demonstrates that the procedure naturally extends *down* to linear, organized behavior.

The rest of the paper is divided into two main sections. In the first section, we derive the basic equations and discuss the analysis behind the procedure. In the second section, we illustrate the procedure using two sample problems. The first example is a trivial one—the vibration of a thin membrane in a rectangular domain; however, its simplicity allows us to examine in some detail the convergence and accuracy of the procedure, the effect of changes in the resolution of the data, the size of the ensemble, and the level of background noise. The second example computes the eigenfunctions in a stadium-shaped domain, a representative problem for complex geometries.

## 2. Derivation of Equations

The question at hand is the determination of the eigenfunctions of an operator $L_x$:

$$L_{\mathbf{x}}[\Psi(\mathbf{x})] = \mu\Psi(\mathbf{x}). \tag{1}$$

In general $L_{\mathbf{x}}$ is a linear matrix valued operator and $\mathbf{x} \in R^n$. For problems of interest $L_{\mathbf{x}}$ is a differential operator (though it may be more general) and the eigenfunctions satisfy suitable homogeneous boundary conditions. For the following remarks, we shall require that the spectrum of $L$ is real although we do not necessarily assume that $L$ is Hermitian.

Consider the two evolution equations

$$i\frac{\partial u}{\partial t} = L[u] \tag{2}$$

$$-i\frac{\partial v}{\partial t} = L^{\dagger}[v], \tag{3}$$

where $L^\dagger$ refers to the adjoint operator. We solve (2) and (3) subject to initial data which are sufficiently general so as to excite all the modes of the corresponding operators.

Next, we form the cross-correlation:

$$K(\mathbf{x}, \mathbf{x}') = \lim_{T \to \infty} \frac{1}{T} \int_0^T u(\mathbf{x}, t)\, v(\mathbf{x}', t)\, dt$$

$$= \langle u(\mathbf{x})\, v(\mathbf{x}') \rangle. \tag{4}$$

Generally, the K–L eigenfunctions, $\Psi(\mathbf{x})$, are defined [5] as the eigenfunctions of the auto-correlation tensor, $\langle u(\mathbf{x})\, u(\mathbf{x}') \rangle$:

$$\int \langle u(\mathbf{x})\, u(\mathbf{x}') \rangle\, \Psi(\mathbf{x}')\, d\mathbf{x}' = \lambda \Psi(\mathbf{x}). \tag{5}$$

Thus, by considering the cross-correlation, $\langle u(\mathbf{x})\, v(\mathbf{x}') \rangle$, we are extending the K–L procedure. To associate the K–L eigenfunctions of $K(\mathbf{x}, \mathbf{x}')$ with those of the linear operator, $L$, we assert that modulo invariant subspaces, $K$ and $L$ have the same eigenfunctions.

This can be shown by applying $L$ to the eigenvalue relation, (5):

$$\int L_{\mathbf{x}}[K(\mathbf{x}, \mathbf{x}')]\, \Psi(\mathbf{x}')\, d\mathbf{x}' = \lambda L_{\mathbf{x}}[\Psi(\mathbf{x})]. \tag{6}$$

Using the definition of $K(\mathbf{x}, \mathbf{x}')$, (4), and the equation of motion, (2), we obtain

$$\int \langle (iu_t(\mathbf{x}))\, v(\mathbf{x}') \rangle\, \Psi(\mathbf{x}')\, d\mathbf{x}' = \lambda L_{\mathbf{x}}[\Psi(\mathbf{x})] \tag{7}$$

and, after integration by parts, we find

$$\int \langle u(\mathbf{x})(-iv_t(\mathbf{x}')) \rangle\, \Psi(\mathbf{x}')\, d\mathbf{x}' = \lambda L_{\mathbf{x}}[\Psi(\mathbf{x})]. \tag{8}$$

Now, using the second equation of motion, (3), and after a second integration by parts, we obtain

$$\int \langle u(\mathbf{x})\, L^\dagger_{\mathbf{x}'}[v(\mathbf{x}')] \rangle\, \Psi(\mathbf{x}')\, d\mathbf{x}'$$

$$= \int \langle u(\mathbf{x})\, v(\mathbf{x}') \rangle\, L_{\mathbf{x}'}[\Psi(\mathbf{x}')]\, d\mathbf{x}' = \lambda L_{\mathbf{x}}[\Psi(\mathbf{x})]. \tag{9}$$

$K$ and $L$ commute and thus have the same eigenfunctions. Note that the eigenvalues, $\mu$, associated with $L$ will not in general be the same as the K–L eigenvalues, $\lambda$. The eigenvalues calculated by the K–L procedure represent the average value of

each eigenfunction over the entire ensemble. If, for example, $u^2$ represents an energy, then $\lambda_n$ represents the average energy of the $n$th mode in the system. For a linear system, each eigenmode evolves without any intersection with the rest of the system, and thus the value of $\lambda_n$ is uniquely determined by the initial conditions. In contrast, the eigenvalues of the operator $L$ have no such meaning and are intrinsic to the equations, not dependent on the initial conditions. The eigenvalues of the operator $L$ may be found by projecting the original data set, $u(\mathbf{x}, t)$, onto the computed eigenfunctions and analyzing the time behavior of the resulting coefficients.

The strategy adopted above is in no way unique. The main purpose of the above is to show, in a general way, how to obtain the eigenfunctions by treating an evolutionary system. Certain operators with special properties will allow other techniques. For example, a negative, self-adjoint operator can be treated in a simpler fashion, solving only one time dependent system: $u_{tt} = L_{\mathbf{x}}[u(\mathbf{x})]$. This is the approach used for the application considered in Section 3.

### 2.1. *Numerical Considerations*

In ref. [6], two different approaches are suggested for computing the K–L eigenfunctions: the direct method and the snapshot method. For the purposes of completeness, we briefly summarize each of the two methods. The goal is to determine the eigenfunctions of $K(\mathbf{x}, \mathbf{x}')$, defined by (4), or equivalently,

$$K = \frac{1}{P} \sum_{k=1}^{P} u^{(k)}(\mathbf{x})\, v^{(k)}(\mathbf{x}') \tag{10}$$

where, e.g.,

$$u^{(k)}(\mathbf{x}) = u(\mathbf{x}, t_k), \qquad k = 1 \cdots P \tag{11}$$

and $t_k$ represents uniformly sampled times.

#### 2.1.1. *Direct Method*

For the implementation of this approach, we immediately introduce the spatial grid on which the data is represented. For data resulting from numerical simulation this will be the collocation grid, while for experimental data it will consist of the sampling locations. For example, if the problem has support in two dimensions,

$$\mathbf{x} = (x, y), \tag{12}$$

we would write $u(x, y, t)$ at the discrete grid points:

$$u(x_n, y_m, t), \qquad n = 1 \cdots N_x, m = 1 \cdots N_y. \tag{13}$$

It follows that

$$K(x, y, x', y') \approx K_{nm;n'm'}. \tag{14}$$

The eigenfunction problem is therefore equivalent to the diagonalization of an $N_x N_y \times N_x N_y$ matrix. More generally, in $d$-dimensions, we will have an $O(N^d)$ matrix to deal with. The operation count for the direct method scales like $O(PN^{2d})$ for the computation of $K_{nm;n'm'}$, and like $O(N^{3d})$ for the diagonalization. Storage requirements scale as $O(N^{2d})$ for both parts of the process. While this approach is feasible when $N^d$ is not too large, it becomes impractical for highly resolved, multi-dimensional problems. In that case, the computational demands and the storage requirements necessitate an alternative approach.

2.1.2. *Snapshot Method*

The snapshot method is based on the fact that the cross-correlation, (10), is a degenerate kernel. From this, the eigenvalue problem, (5), has a solution which can be written as

$$\Psi = \sum_{n=1}^{P} \alpha_n u^{(n)}(\mathbf{x}), \tag{15}$$

i.e., an admixture of snapshots. Thus if we define

$$C_{nm} = \int u^{(n)}(\mathbf{x})\, v^{(m)}(\mathbf{x})\, d\mathbf{x}, \tag{16}$$

then the problem is solved by performing the eigen-analysis of the $P \times P$ matrix $C_{nm}$, i.e.,

$$C\boldsymbol{\alpha} = \lambda \boldsymbol{\alpha} \tag{17}$$

where $\boldsymbol{\alpha} = (\alpha_1, ..., \alpha_P)$ and $\lambda$ is the same as in (5). Using this approach, the operations count scales like $O(N^d P^2)$ operations for the computation of $C_{nm}$, $O(P^3)$ operations for the diagonalization of $C_{nm}$, and $O(N^d P^2)$ operations to compute $\Psi(\mathbf{x})$ using (15). Storage requirements also scale like $O(P^2)$.

Whereas the direct method proved to be unusable for large $N^d$, the snapshot method remains viable and only becomes impractical for large ensemble size, $P$. The choice of the approach will usually be determined by the form of the particular data set. If the data consists of a long time history with only moderate spatial resolution (typical of experimental data or a coarse numerical simulation) the direct method will be appropriate, whereas if the data consists of a moderate time history with high resolution (common for large-scale numerical simulations of complex systems) the snapshot method becomes favorable.

## 3. Numerical Examples

As an illustration of this procedure, we will present two illustrative examples in which the K–L procedure has been used to find the eigenfunctions of the Laplacian operator

$$\nabla^2 \Phi = \mu \Phi. \tag{18}$$

We consider two differently shaped domains: (i) a rectangular domain and (ii) a "quarter-stadium" domain. In both cases, the K–L procedure has been applied to data from the numerical solution of the two-dimensional wave equation:

$$\frac{\partial^2 u}{\partial t^2} = \nabla^2 u. \tag{19}$$

Since the Laplacian is a negative operator, we can employ a simplified version of the analysis from Section 2, solving only one equation of motion with $u_{tt}$ as the left-hand side. Since the operator is self-adjoint, the cross-correlation, (10), will be a Hermitian matrix, further simplifying the computational procedure.

The first of these examples can be considered as the vibration of a membrane in a rectangular domain and seems at first to be a trivial application of the procedure. However, it is chosen for that reason, since it affords us the opportunity of investigating the convergence properties and overall accuracy of the method. Since the eigenvalues and eigenfunctions for this problem are easily found analytically and since the problem is simple enough to be repeated for many different resolutions and ensemble sizes, we can thoroughly explore the application of the procedure for different choices of the parameters. In addition to varying the resolution ($N$) and ensemble size ($P$), we shall also investigate the effect of various levels of random noise on the accuracy of the eigenfunctions, an issue of some importance when applying the procedure to experimentally obtained data.

The second example, which can be interpreted as the vibration of a thin membrane in a "stadium," is presented as an example of the application of the procedure in a complex geometry. The stadium geometry, achieved by appending two semicircles of unit radius onto the ends of a rectangle of length $L$, is a classic geometry that has been extensively studied [7–10] and is known to possess spatially chaotic eigenfunctions when $L > 0$.

We should note that once the data and the correlation matrix (or inner product matrix) has been obtained, the numerical procedure is blind as to both the nature of the data and the system that produced it. Thus the K–L decomposition does not know, or care, whether the geometry of the domain is a simple rectangle, a stadium, or any other arbitrary shape. In addition, the complexity of the dynamical system that generated the data, whether it be linear, nonlinear, ordered or chaotic, does not affect the computation of the eigenfunctions and eigenvalues. Thus, the use of the rectangular geometry as a test case is not a numerically insignificant problem, since we would expect that the convergence and accuracy data that we obtain from this system will be generally applicable to many systems.

### 3.1. *Vibration of a Membrane in a Rectangular Domain*

We choose a domain bounded by $x = [0, L]$, $y = [0, 1]$, where $L$ is the aspect ratio of the rectangle. It is easy to show that the analytic eigenfunctions are

$$\Phi_{nm}(x, y) = \sin\left(\frac{n\pi x}{L}\right)\sin(m\pi y). \tag{20}$$

To apply the K–L procedure, we need an ensemble of snapshots from the system, and in general, we would have to utilize some procedure to numerically solve the equations of motion. However, for this case, since the equations are so simple we can generate our ensemble analytically. If the initial conditions state that at $t=0$ the membrane has some initial, stationary, configuration, then the complete solution $u(x, y, t)$ may be represented by

$$u(x, y, t) = \sum_{n=1}^{N} \sum_{m=1}^{M} A_{nm} \cos(k\pi t) \sin\left(\frac{n\pi x}{L}\right) \sin(m\pi y); \tag{21}$$

$N$ and $M$ are the number of modes used in the $x$ and $y$ directions, respectively, and $k = \sqrt{n^2/L^2 + m^2}$. The coefficients $A_{nm}$ are determined by the initial conditions and are the eigenvalues that will be approximated by the K–L procedure. Once we have specified these, we can quickly calculate $u(x, y)$ at any time. In addition, the matrix $C_{ij}$, required for the snapshot method, can be calculated simply by making use of the orthogonality of the trigonometric functions:

$$C_{ij} = \sum_{n=1}^{N} \sum_{m=1}^{M} A_{nm}^2 \cos(k\pi t_i) \cos(k\pi t_j). \tag{22}$$

Sirovich [11] has discussed extensively the extensions to the original data set that may be found by making use of symmetries inherent in the geometry of the problem. For the present case, any single realization may be extended into four independent realizations by reflections about the two axis of symmetry, and 180° rotation. The data extension has two purposes. First, when the ensemble is of limited size, it increases that size dramatically (some geometries allow as many as 16 new realizations from a single snapshot), thus improving the statistical quality of the results. Second, the data extension endows the computed eigenfunctions with the symmetries that we know the analytic eigenfunctions possess, thus improving the geometric fidelity of the computed eigenfunctions. For the present case, new snapshots are not difficult to generate and so statistical considerations do not require any data extension. However, the geometric advantages can still be enjoyed and, for all the results presented, we have made use of the fourfold symmetry that the rectangular domain allows.

For the present example, the aspect ratio was chosen as the Golden Mean, $L = (1+\sqrt{5})/2 \approx 1.61803$, while the initial conditions $u_0(x, y)$ were specified as

$$u_0 = \sin\left(\frac{x\pi}{L}\right) \sin(y\pi)\, e^{-10[(2x-L-0.1)^2 + (2y-1-0.2)^2]} \tag{23}$$

$$\frac{\partial u_0}{\partial t} = 0. \tag{24}$$

The first twenty coefficients, $A$ (for these initial conditions), computed for $N = M = 16$, along with the mode numbers in the $x$ and $y$ directions, $n$ and $m$, are given in Table I. Since these coefficients are approximated by the K–L eigenvalues,

TABLE I

Analytic Values for the K–L Eigenvalues and Mode Numbers for Initial Conditions

| Rank | $n$ | $m$ | Eigenvalue, $\Lambda$ |
|---|---|---|---|
| 1 | 1 | 1 | 0.27907109 |
| 2 | 3 | 1 | 0.18179162 |
| 3 | 5 | 1 | 0.07670603 |
| 4 | 1 | 2 | 0.05969925 |
| 5 | 1 | 3 | 0.05850719 |
| 6 | 1 | 4 | 0.04820078 |
| 7 | 3 | 2 | 0.03888910 |
| 8 | 3 | 3 | 0.03811257 |
| 9 | 3 | 4 | 0.03139880 |
| 10 | 7 | 1 | 0.02069319 |
| 11 | 4 | 1 | 0.01878738 |
| 12 | 5 | 2 | 0.01640905 |
| 13 | 6 | 1 | 0.01626344 |
| 14 | 5 | 3 | 0.01608140 |
| 15 | 5 | 4 | 0.01324856 |
| 16 | 2 | 1 | 0.00832501 |
| 17 | 8 | 1 | 0.00757156 |
| 18 | 1 | 6 | 0.00672115 |
| 19 | 7 | 2 | 0.00442672 |
| 20 | 3 | 6 | 0.00437827 |

*Note.* $u_0(x, y) = \sin(x\pi/L)\ \sin(y\pi)\ \exp(-10[(2x - L - 0.1)^2 + (2y - 1 - 0.2)^2])$ with $N = M = 16$. The eigenvalues are normalized so that $\sum \Lambda_n = 1$.

we shall refer to them as the *analytic eigenvalues* (we reiterate that they are the true values of the K–L eigenvalues and *not* the eigenvalues of the Laplacian operator).

Using the above equations, a data set consisting of 100 realizations, each separated by 0.5 time units, was constructed. This was done for three different resolutions: $4 \times 4$ ($N = 16$), $8 \times 8$ ($N = 64$), and $16 \times 16$ ($N = 256$). Here $N$ is defined as the *total* size of the system. With the described data extensions, this yielded 400 separate realizations at each resolution.

The K–L procedure was applied to various subsets of the entire data ensemble. Since the resolutions used in this example are not too large, both the direct method and the snapshot method were used to calculate the K–L eigenvalues and eigenfunctions. This also allows us to see how the different methods affect the results.

### 3.1.1. *Accuracy and Convergence*

Figure 1 shows the eigenvalue spectrum computed using the snapshot method. The resolution of the calculation was $N = 256$ ($16 \times 16$), and the solid line plots the analytic eigenvalues at this resolution for comparison. The different lines represent different ensemble sizes, ranging from 24 to 400 (including symmetry extensions).

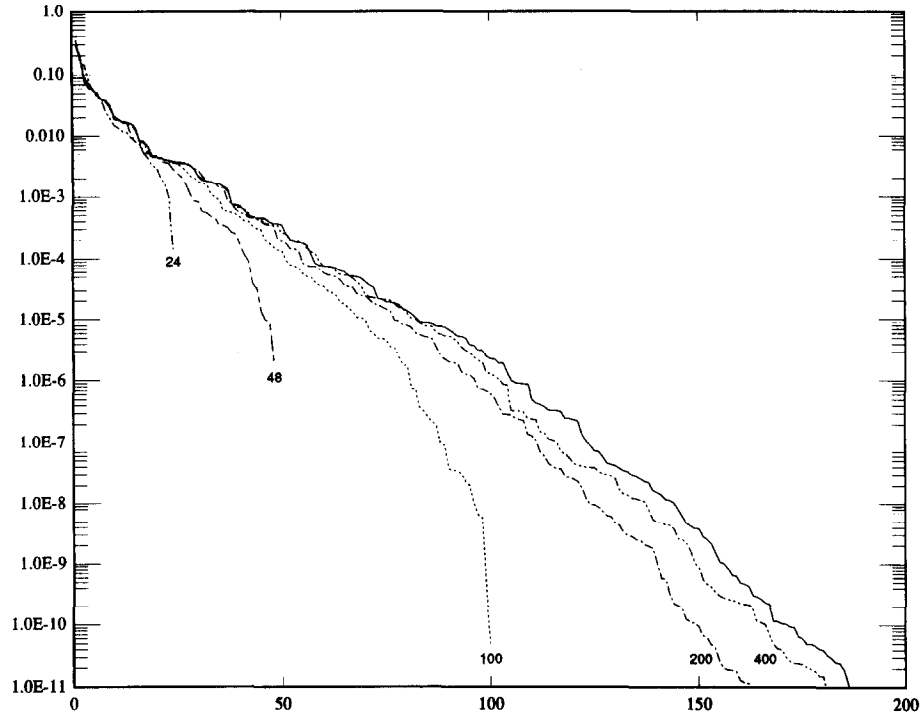


FIG. 1. Eigenvalue spectrum for wave equation in a rectangular domain. Resolution of computation was $16 \times 16$. Solid line represents the analytic spectrum while dotted lines show K–L computed spectrum with ensemble sizes of 24, 48, 100, 200, and 400.

As we would expect, as $P$ increases, the computed spectrum approaches the analytic spectrum and the fidelity of the K–L procedure improves. For $P = 24$, only the first few eigenvalues are well approximated before the spectrum begins to fall off, while for $P = 400$ the computed and analytic spectra are essentially indistinguishable until about $k = 50$, and even then, the K–L spectrum produces a reasonably faithful estimation of $\lambda_k$ throughout 12 orders of magnitude.

A better indication of the accuracy of the K–L estimation is shown in Fig. 2, which plots the running average of the error in the eigenvalues,

$$E_\lambda(k) = \frac{1}{k} \sum_{i=0}^{k} \frac{|\lambda_i - \Lambda_i|}{\Lambda_i}, \tag{25}$$

where $\lambda_i$ is the $i$th empirical eigenvalue, while $\Lambda_i$ is the $i$th analytic eigenvalue, based on the initial data. Figure 2 plots this error for the five different ensemble sizes ranging from 24 to 400. As expected, the average error decreases as the ensemble size increases, but we also see that the convergence of the eigenvalues is not completely uniform. For example, for $P = 400$, the jump in the average error at $k = 4$ indicates

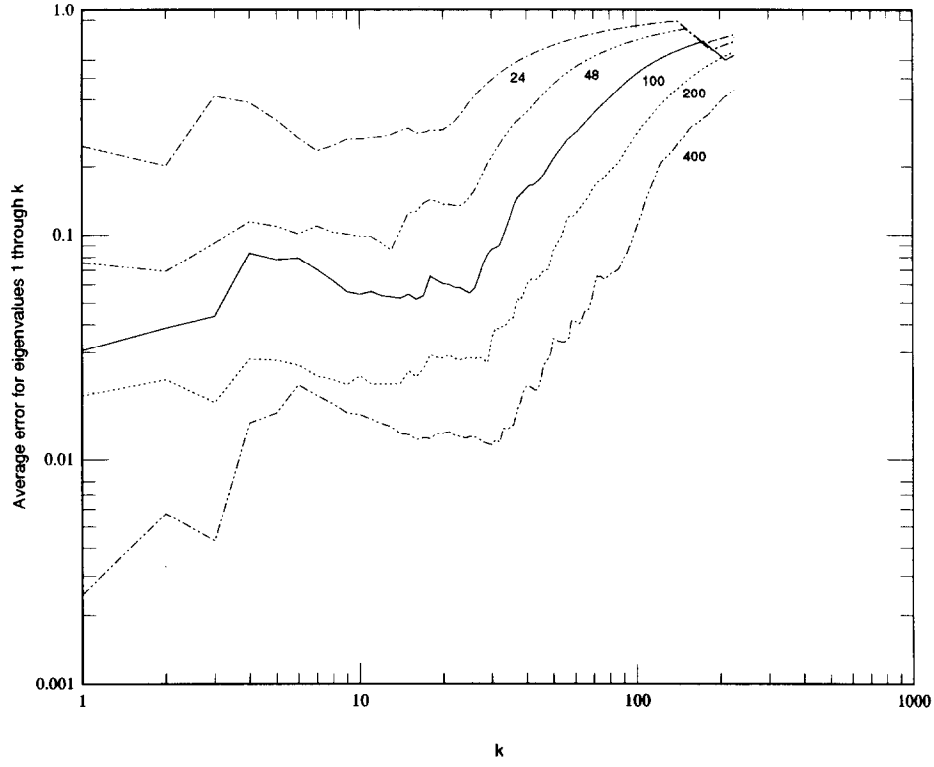


FIG. 2. Running average of error of computed eigenvalues for $N = 256$. Lines indicate different sizes of the ensemble.

that $\lambda_4$ is not accurately resolved. However, for the range of $k$ between 6 and about 40, the average error decreases, indicating that these eigenvalues are better resolved than at lower values of $k$. At high values of $k$, the running average becomes smooth and rises rapidly. This corresponds to the spurious eigenvalues at values of $k$ larger than the size of the ensemble. At these values of $k$, the error is always 1, and the average error rises accordingly. One should comment that the running average does present the results in the most severe manner, since it treats each eigenvalue with equal weight despite their wildly disparate sizes.

The use of the snapshot method yields almost identical results to those of the direct method. Whereas the direct method gives $N$ eigenvalues regardless of the ensemble size, $P$, the snapshot method gives $P$ eigenvalues, regardless of the size of $N$. Thus for $P < N$ all of the eigenvalues computed are valid approximations to the analytic spectrum. However, when the ensemble size exceeds the resolution of the problem, spurious eigenvalues are generated. These spurious eigenvalues have a characteristic magnitude comparable to the machine accuracy of the computer and thus are easily distinguished from the true ones. Apart from this, the snapshot and

direct methods yield almost identical results and unless there is a comment to the contrary, the results reported could be from either method.

Of course, the primary application of the K–L procedure in this context is to approximate the analytic eigenfunctions. Figure 3 shows the first 12 eigenfunctions computed using an ensemble size of 400. The eigenfunctions are normalized so that their maximum value is 1. The lines plotted represent the zero-crossings and the $\pm 0.5$ contour levels. At a glance, the eigenfunctions seem to have reproduced the analytic eigenfunctions remarkably well. The zero-crossings are generally straight lines and are aligned well with the side boundaries. There are some nonuniformities in the eigenfunctions which are especially noticeable when looking at the 0.5 level contours, but in general the comparison is quite favorable. The somewhat haphazard ordering of the eigenfunction is determined by the initial membrane displacement, and comparison of the computed modes with the analytic modes (given in Table I) shows that the mode numbers of the first 12 eigenfunctions are correctly captured.

As a quantitative measure of the accuracy of the eigenfunctions, we have calculated the root-mean-square error of each eigenfunction, $\Psi_k$, as compared with the known analytic eigenfunction, $\Phi_k$. As with the eigenvalues, we then calculate the running average of the error in the eigenfunctions, which is now defined as

$$E_{\Psi}(k) = \frac{1}{k} \sum_{i=0}^{k} \left[ \int (\Psi_k(\mathbf{x}) - \Phi_k(\mathbf{x}))^2 \, d\mathbf{x} \right]^{1/2}. \tag{26}$$

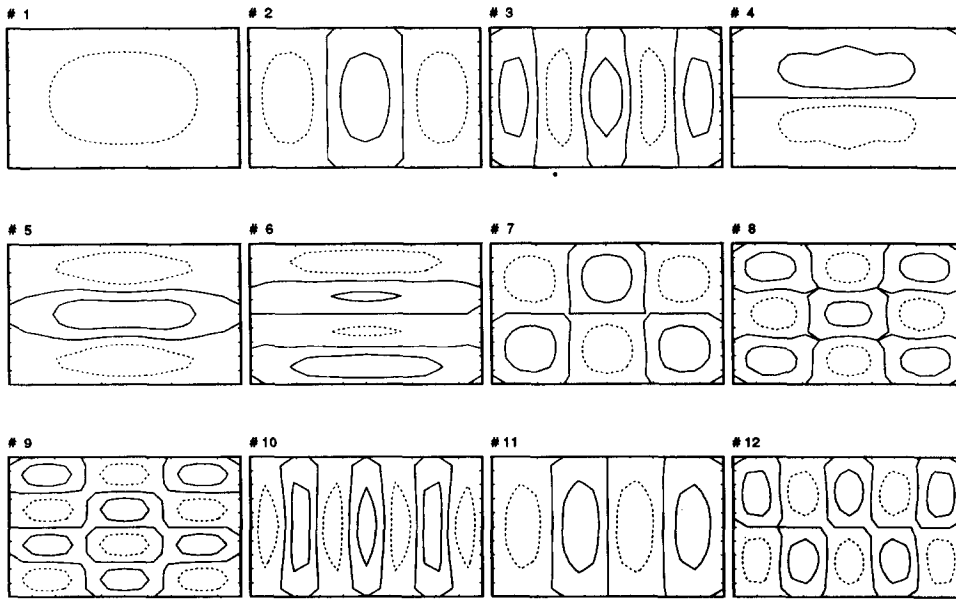


FIG. 3. Contour plot of the first ten eigenfunctions computed using the K–L procedure for $N = 256$, $P = 400$. Eigenfunctions are normalized to have a maximum of 1. Contour levels show zero crossings and $\pm 0.5$.

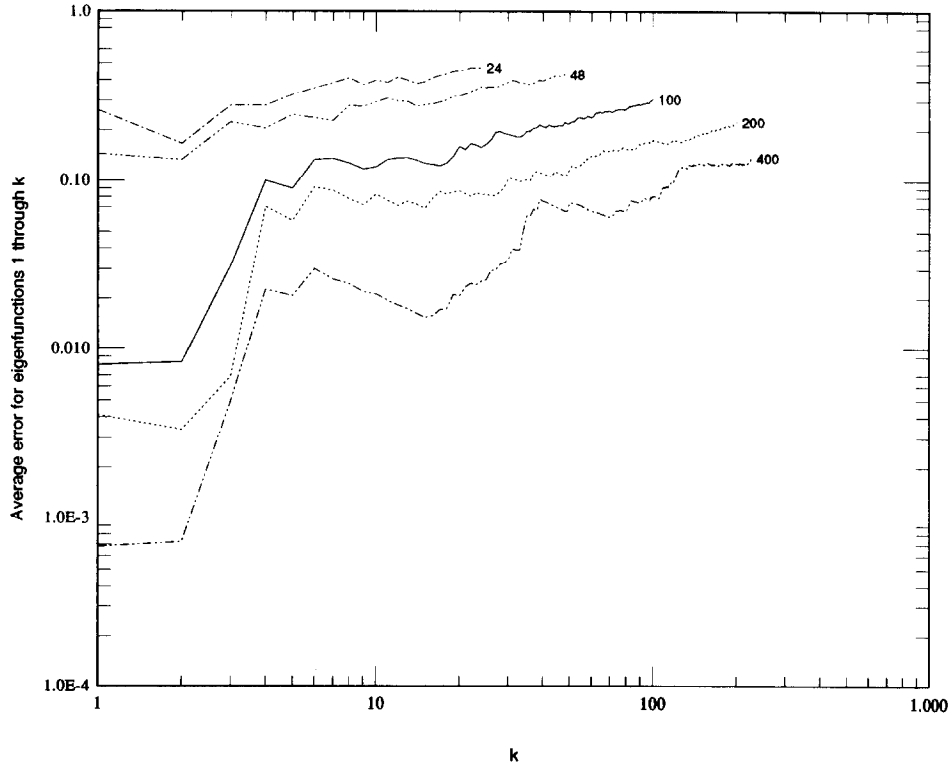


FIG. 4. Running average of error of computed eigenfunction for $N = 256$. Lines show different sizes of the ensemble.

The results of this computation for a resolution of 256 are shown in Fig. 4, where each line represents a different ensemble size. As with the eigenvalues, the error, for any fixed value of $k$, decreases when the ensemble size is increased. While the first two eigenfunctions are resolved with good accuracy, the resolution of $\Psi_3$ is not so good, as indicated by the sudden jump in the average error. However, from $k = 3$ onwards, the accuracy of the computation is quite consistent, rising slowing as $k$ increases. Nevertheless, for $P = 400$, the average error for the first 100 eigenfunctions is only 0.1, indicating that on average, 90% of each analytic eigenfunction was correctly captured by the corresponding empirical eigenfunction.

### 3.1.2. *The Effect of Resolution*

Surprisingly, changes in the resolution of the data set had very little effect on the accuracy of the eigenfunction calculation. Three different resolutions were considered, with 16, 64, and 256 total grid points in the domain. For a large ensemble size ($P = 400$) in which the average error in $\Psi$ ranges from 0.001 to 0.1 (Fig. 4) the lowest resolution did show a decrease in the accuracy of the first few eigenfunctions,

although even at this coarse level the average error for all $\Psi_k$ was still only 0.1 and comparable with the higher resolutions. For the two finer resolutions, the accuracy of the computed $\Psi_k$ was almost unchanged for the first ten eigenfunctions. For higher values of $k$, the errors for the $N=64$ calculation, while larger than the $N=256$ case, were still of the same order. For a small ensemble size (for example, $P=48$), the average errors in $\Psi_k$ ranged between 0.1 and 0.5, but the change in the resolution of the data set had almost no consistent effect on the accuracy of the computed eigenfunctions. This was the case even for the $N=16$ resolution.

### 3.1.3. *The Effect of Noise*

In the previous section, the convergence of the K–L procedure was examined using data that was accurate and clean. This was afforded by the choice of a problem which enabled us to generate the realizations very accurately, since we have the complete solution to the problem in closed form. The nature of the data thus allowed us to concentrate exclusively on the performance of the K–L method without any concern for issues regarding the data itself. In practical situations, however, the data is unavoidably imperfect. In the case of numerically simulated

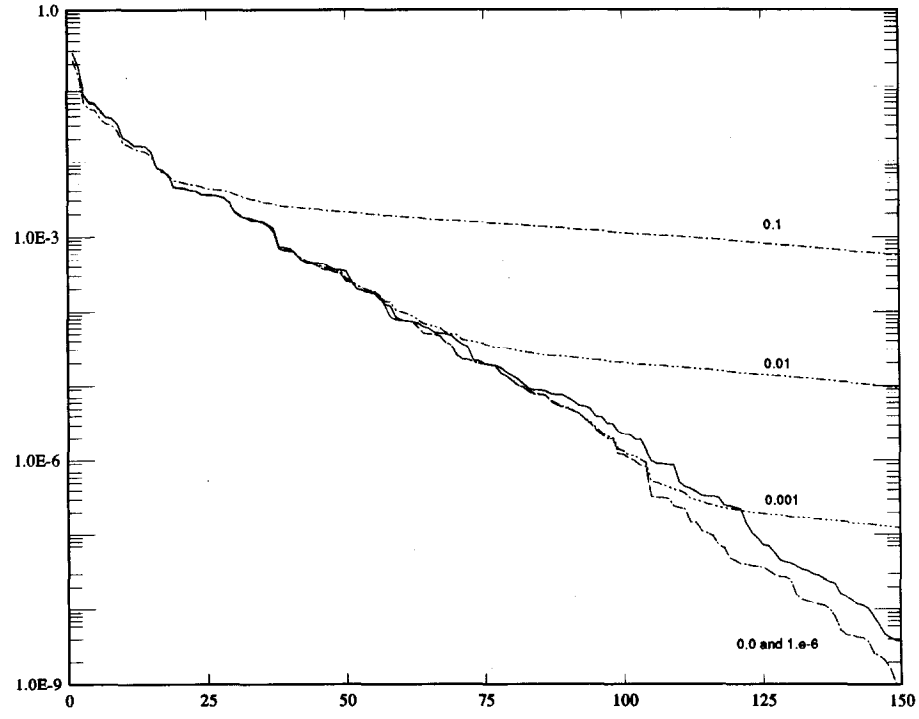


FIG. 5. Eigenvalue spectrum for rectangular geometry, $N=256$, $P=400$, illustrating the effect of random noise. The solid line depicts the analytic spectrum while the dotted lines show the spectra for noise levels of $10^{-1}$, $10^{-2}$, $10^{-3}$, and $10^{-6}$.

data, there may be high-frequency oscillations arising from the particular numerical scheme, or, conversely, a suppression of high-frequencies through the use of a numerical filtering technique. With experimental data, random background noise and experimental inaccuracies often leave the data mildly contaminated.

In this section, we examine the effect that random noise has on the accuracy and convergence of the K–L procedure in the determination of the linear eigenfunctions. We examine this issue using the same test problem as before, namely the vibration of a thin membrane in a rectangular domain. The procedure is identical to that described above with the one exception that at each point of the domain, and to every realization, we add random noise. The noise consisted of pseudo-random numbers ranging from $-A$ to $+A$, where $A$ is a specified amplitude level. Four different levels of noise were added, with amplitudes $A = 10^{-6}$, $10^{-3}$, $10^{-2}$, and $A = 10^{-1}$ (compared with the membrane displacement, which is $O(1)$). Thus at the highest level, the signal-to-noise ratio was 10. From this new set of "noisy" data, the K–L eigenvalues and eigenfuncions were computed as before, again using both the direct and the snapshot methods and performing the calculation for a variety of resolutions and ensemble sizes.

Figure 5 shows the eigenvalue spectra for $N = 256$, $P = 400$ with several noise levels. The solid line shows the analytic spectrum for comparison. The effect of the random noise is evident. For eigenvalues larger than a given threshold (which

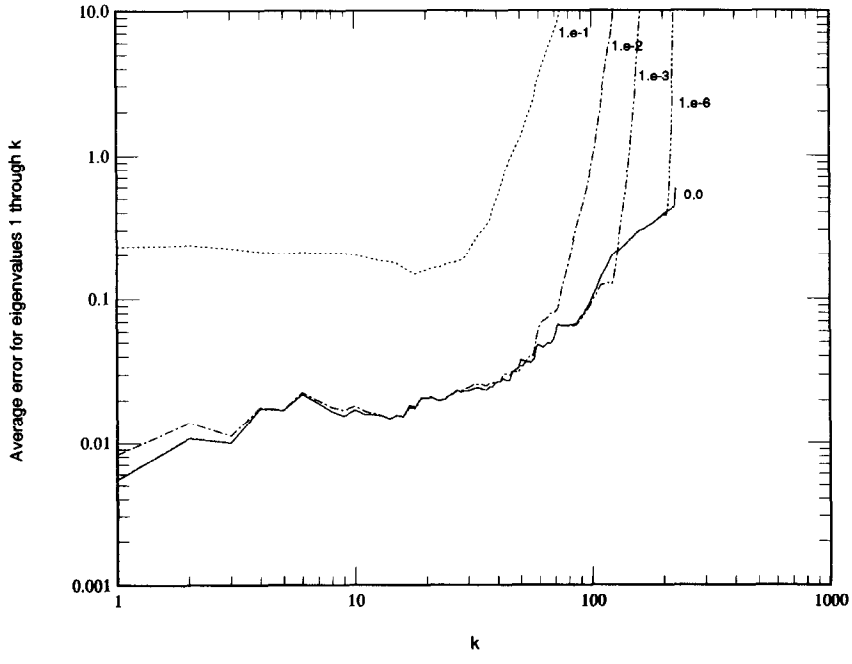


FIG. 6. Running average of error of computed eigenvalues for $N = 256$, $P = 400$. Lines represent different levels of background noise.

depends on the noise level) the noise has no discernible effect and the eigenvalue spectra lie on top of one another. As $k$ increases and the magnitude of the eigenvalue decreases, the effect of the contamination becomes apparent as the computed spectrum peels off and departs from the zero-noise spectrum. It is interesting and encouraging to note that the magnitude of the eigenvalue where the accuracy of the computation breaks down is considerably smaller than the level of the noise injected into the raw data. Thus, for the case in which $O(10^{-2})$ noise was introduced into $u$, the eigenvalues are still essentially correct until $\lambda_k \approx O(10^{-4})$. The level at which the noise-distorted spectrum begins to dominate over the true spectrum does not appear to change with smaller ensemble size or lower resolution. In fact, the overall accuracy of the calculation is almost completely unaffected by the noise level for values of $k$ smaller than the peel-off value. This is illustrated in Fig. 6, which shows the running error of the computed eigenvalues for the $N=256$, $P=400$ case. The solid line plots the zero-noise case while the dotted lines plot the running error for increasing levels of noise. For the 10% noise level, the entire calculation is adversely affected and the errors in the eigenvalues are roughly one order of magnitude higher than the zero-noise case. However, for the smaller noise levels, the errors are essentially indistinguishable from the zero-noise case until the point at which the eigenvalue spectrum (Fig. 5) peels off due to noise level. The result is not quite so good for the eigenfunctions (Fig. 7) and one sees that the average error

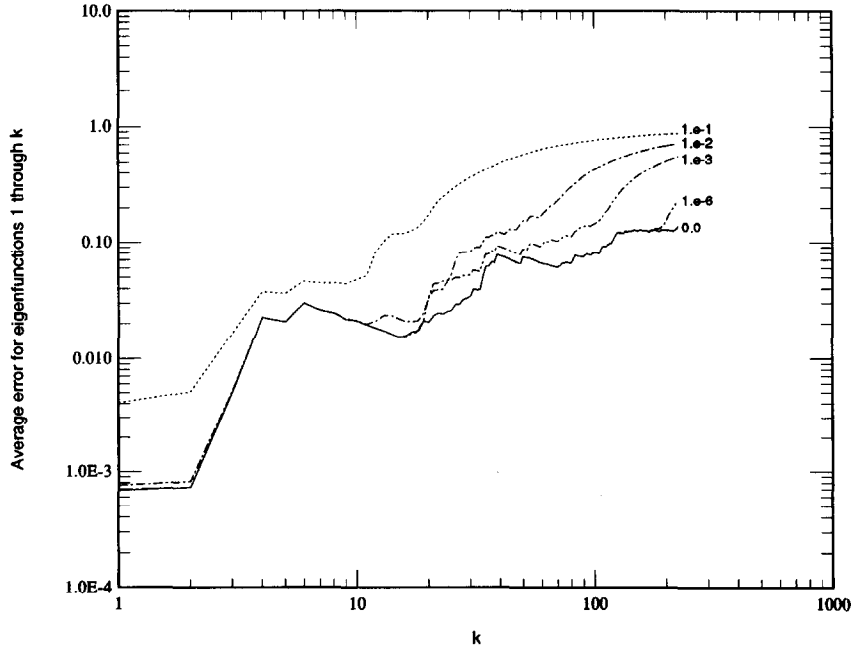


FIG. 7. Running average of error of computed eigenfunctions for $N=256$, $P=400$. Lines indicate different levels of background noise.

in the eigenfunctions begins to rise somewhat before the peel-off point occurs in the corresponding eigenvalue spectrum.

### 3.2. *The Vibration of a Membrane in a Stadium*

The second numerical example considered applies the K–L procedure to a complex geometry, namely that of a stadium. It is known [7, 8] that a stadium geometry is ergodic. In addition, the lack of any simple geometry makes the analytic determination of these eigenfunctions impossible. Using the K–L procedure, however, we demonstrate that the computation of the linear eigenfunctions may be achieved relatively easily. The stadium geometry does have fourfold symmetry, meaning that all of the eigenfunctions will have either odd or even symmetry with respect to the $x$ and the $y$ axes. In order to simplify the present analysis, we have chosen to examine only the odd–odd eigenfunctions. This means that we need only study one quarter of the stadium geometry, applying homogeneous boundary conditions along the lines of symmetry. The remaining symmetries can be studied by changing the boundary conditions along the lines of symmetry appropriately.

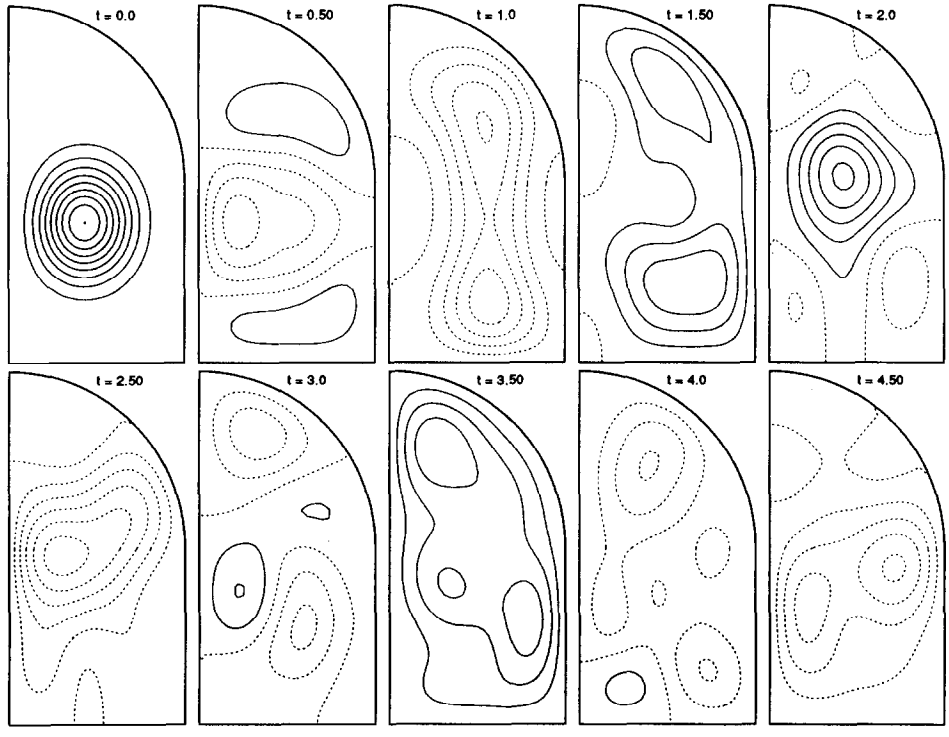


FIG. 8. Ten representative realizations of the membrane height in the stadium geometry. At $t = 0$ the initial condition is specified as a Gaussian-shaped distortion of height 1. Solid lines indicate positive heights while dotted lines indicate negative membrane heights. Contour spacing = 0.1.

In order to apply the K–L method, we must first obtain an ensemble of realizations of the dynamical system. This was achieved by numerical simulation of the wave equation (19) in the quarter-stadium geometry. The numerical scheme employed was a spectral collocation method in which the quarter-stadium was divided into two domains: a square and a quarter circle. Each domain was then integrated separately, matching the solution and its first derivative at the interface at each time step. The computations were carried out using a $32 \times 64$ grid. Since we do not know the analytic solution for the eigenfunctions, we do not know to what extent our results are accurate. However, based on the previous section, we can estimate the limits of validity for our computation.

Ten successive individual realizations from an ensemble of 427 are shown in Fig. 8 as examples of the data set. With the resolution used, the direct method would require the inversion of a $2145 \times 2145$ matrix which is computationally impractical. Thus the results presented were achieved using the snapshot method. Figure 9 shows the first 10 eigenfunctions. The contour lines plot the zero-crossings and the $\pm 0.5$ levels. The first few eigenfunctions look like what one might expect and can be described in the same terms as the rectangular geometry: $\Psi_1$ is the (1,1)

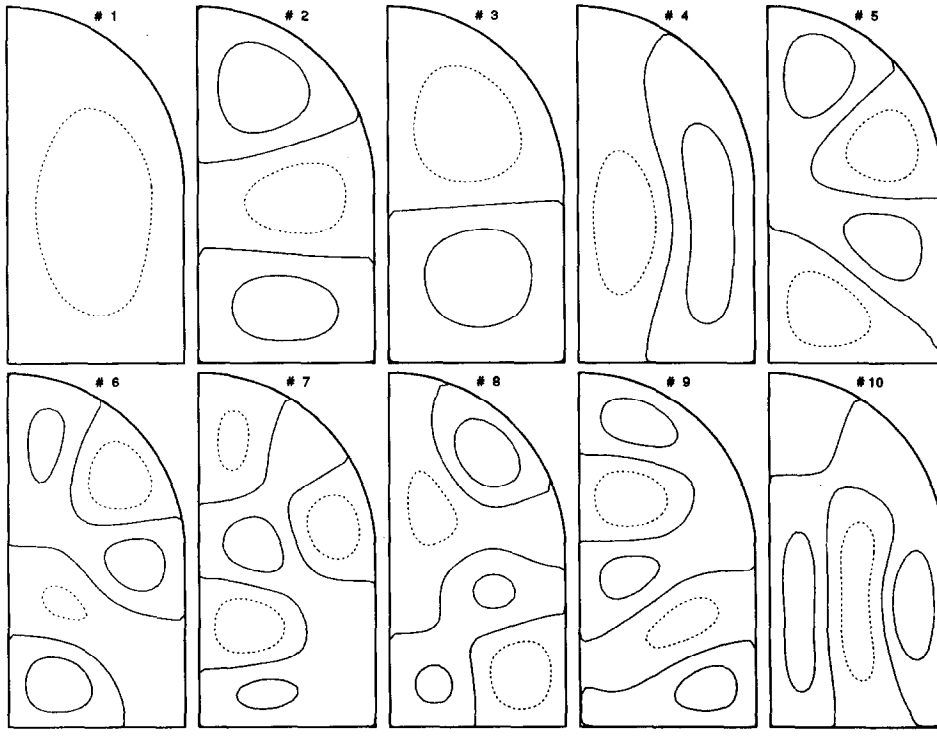


FIG. 9. First ten eigenfunctions for the stadium geometry. Contour lines indicate levels of 0.0, $\pm 0.5$.

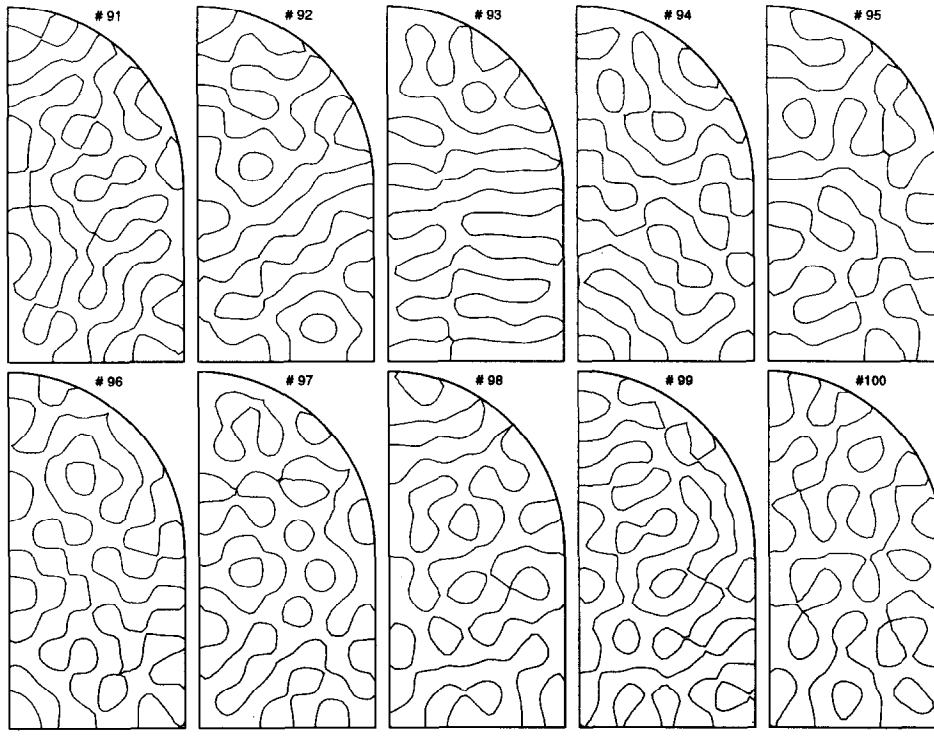


FIG. 10. Eigenfunctions 91–100 for the stadium geometry illustrating spatial incoherence of the higher eigenfunctions. Contour lines show zero-crossings.

mode; $\Psi_2$, the (1,3) mode; etc. However, the irregular geometry quickly breaks this pattern, and then next few eigenfunctions do not follow any easily describable pattern. This is even more evident in the higher eigenfunctions, a few of which are shown in Fig. 10. Occasionally, strong patterns, similar to the rectangular geometry, are clearly visible (for example, $\Psi_{93}$). However, for the most part, the zero-crossings are randomly oriented with respect to the boundary and each other. This is in good qualitative agreement with McDonald and Kaufman's [12] example of a stadium eigenfunction. Unlike the eigenfunctions in the rectangular geometry, the nodal lines here do not intersect. However, as Berry notes [13], this alone does not imply any chaotic structure, since the crossing of the nodal lines of eigenfunctions is not a generic property of Hamiltonian systems.

## 4. CONCLUSIONS

The results presented here indicate that the eigenfunctions of a general class of linear operators in an arbitrary geometry may be accurately and easily obtained by

using the K–L procedure. The method is computationally efficient and the choice between the two approaches, the direct method and the snapshot method, ensure that the procedure may be efficiently used both when the size of the ensemble is large but the spatial resolution is limited (typical in an experimental setup) or when the spatial resolution is large but the ensemble size is limited (typical for a numerical simulation).

By applying the technique to a problem in which the analytic solution is well known, we have been able to determine the convergence and accuracy characteristics of the procedure. Of course, these will differ from problem to problem, depending on the eigenvalue spectrum and the spatial complexity of the eigenfunctions. A very encouraging result is the insensitivity of the results to moderate levels of random noise and the fact that the most energetic eigenfunctions are accurately resolved with very moderately-sized ensembles.

The ability of the K–L procedure to compute the linear eigenfunctions also opens a new possibility in the study of complex, nonlinear systems. The procedure has been applied to complex nonlinear systems (for example, to a turbulent boundary layer [3]), and this indicates that the procedure has physical meaning down in a linear regime. Thus is would be possible to follow a system from a linear state through various stages of complexity. At every stage one could obtain a basis of eigenfunctions that optimally describe the system. At the linear stage, these will be the eigenfunctions of the linear system. However, as the flow becomes nonlinear and more complex, these eigenfunctions will gradually distort, reflecting the different phenomena that govern the flow at each stage. A similar approach has been reported by Deane and Sirovich [14] in their study of low-Rayleigh-number turbulent convection.

## ACKNOWLEDGMENTS

This work was supported by DARPA-URI Grant N00014-86-K0754. Some the calculations presented were performed at the John von Neumann Supercomputing Center which is supported by the National Science Foundation.

## REFERENCES

1. G. H. GOLUB AND C. F. VAN LOAN, *Matrix Computations* (Johns Hopkins Univ. Press, Baltimore, 1983).
2. L. SIROVICH AND D. RODRIGUEZ, *Phys. Lett. A* **120**, 211 (1987).
3. N. AUBRY, P. HOLMES, J. L. LUMLEY, AND E. STONE, *J. Fluid Mech.* **192**, 115 (1988).
4. K. S. BALL, L. SIROVICH, AND L. R. KEEFE, *Int. J. Numer. Methods Fluids* **12**, 585 (1991).
5. M. LOÈVE, *Probability Theory* (Van Nostrand, Princeton, NJ, 1955).
6. L. SIROVICH, *Q. Appl. Math.* **45**, No. 3, 561 (1987).
7. L. A. BUNIMOVICH, *Funct. Anal. Appl.* **8**, 254 (1974).
8. L. A. BUNIMOVICH, *Commun. Math. Phys.* **65**, 295 (1979).

9. G. BENETTIN AND J.-M. STRELCYN, *Phys. Rev. A* **17**, 773 (1978).
10. M. BERRY, *Eur. J. Phys.* **2**, 91 (1981).
11. L. SIROVICH, *Q. Appl. Math.* **45**, No. 3, 573 (1987).
12. S. W. MCDONALD AND A. N. KAUFMAN, *Phys. Lett. Rev.* **42**, No. 18, 1189 (1979).
13. M. BERRY, "Semiclassical Mechanics of Regular and Irregular Motion," Lectures given at the Les Houches Summer School, 1981.
14. A. E. DEANE AND L. SIROVICH, *J. Fluid Mech.* **221**, 251 (1991).