Available online at www.sciencedirect.com
ScienceDirect
Journal of Sound and Vibration 310 (2008) 448–458
JOURNAL OF SOUND AND VIBRATION
www.elsevier.com/locate/jsvi

Short Communication

An efficient computational procedure for random vibro-acoustic simulations

Jean-Pierre Coyette[a], Karl Meerbergen[b,*]

[a]Free Field Technologies, 1 rue Francqui, B-1435 Mont–Saint–Guibert, Belgium
[b]Department of Computer Science, Katholieke Universiteit Leuven, Celestijnenlaan 200A, B-3001 Heverlee, Belgium

Received 18 December 2006; received in revised form 26 July 2007; accepted 30 July 2007
Available online 19 September 2007

Abstract

We discuss a new method for handling random acoustic excitations in finite-element models. The idea is to approximate the cross power spectral density matrix of the response by a low rank matrix. We illustrate the low rank approximation can be computed efficiently by the implicitly restarted block Lanczos method. The novelty lies in applying the Lanczos method to the cross power spectral density matrix of the response rather than the excitation. We give a theoretical explanation for finite-element models with modal damping.

1. Introduction

Vibro-acoustic models are often subjected to random excitations. Examples are acoustic diffuse fields (e.g. in reverberant test chambers) and turbulent boundary layer excitations (e.g. in aerodynamic noise studies).

The framework for modelling such distributed excitations is the mathematical concept of (weakly) stationary random process. Such processes are usually characterized, in the frequency domain, by power spectra and are practically defined by referring to a reference power spectrum and a suitable spatial correlation function.

In the time domain, a random process x has a mean zero over one period of time; the auto-correlation function $R(\tau)$ is the mean value of the product $x(t)x(t+\tau)$ over a period of time. The power spectral density (PSD) is the Fourier transform of the auto-correlation function and is the characterization of a random process in the frequency domain. If $\mathbf{x}$ is a vector of excitations, the random process is determined by the cross power spectral density matrix $\mathbf{S}_x(\omega)$. Its (i,j) entry is the cross correlation function for the entries i and j in $\mathbf{x}$. The diagonal elements are the auto correlation functions for the entries of $\mathbf{x}$. The matrix $\mathbf{S}_x(\omega)$ is Hermitian positive semi-definite. See Refs. [1–5] for the theoretical background of random processes.

Let in the frequency domain, $\mathbf{x}$ be the excitation vector and $\mathbf{y}$ the output vector, where $\mathbf{y}=\mathbf{H}\mathbf{x}$ with $\mathbf{H}$ the receptance matrix (the inverse of the dynamic stiffness matrix $\mathbf{Z}$). Then

$$\mathbf{S}_y(\omega)=\bar{\mathbf{H}}(\omega)\mathbf{S}_x\mathbf{H}^{\mathrm{T}}(\omega), \tag{1}$$

*Corresponding author.
E-mail address: karl.meerbergen@cs.kuleuven.be (K. Meerbergen).

0022-460X/$ - see front matter © 2007 Elsevier Ltd. All rights reserved.
doi:10.1016/j.jsv.2007.07.068

which is Hermitian positive semi-definite. In a simulation, we are often interested in a few diagonal entries of $\mathbf{S}_y$ or the sum of the diagonal entries. In general, $\mathbf{Z}$ is a complex symmetric or unsymmetric matrix. In many cases, $\mathbf{Z}(\omega) = \mathbf{K} - \omega^2\mathbf{M}$ where $\mathbf{K}$, and $\mathbf{M}$ are the stiffness and mass matrices, respectively. For the theoretical analysis in Section 5, we assume that $\mathbf{Z}$ takes the form $\mathbf{K} - \omega^2\mathbf{M}$.

In a finite-element context, the random excitation $\mathbf{x}$ is usually defined on a part of the boundary surface. Let n be the number of dofs in the finite-element model and m be the number of dofs along the loaded discrete surface. Usually, m is significantly smaller than n. Then, we could write $\mathbf{S}_x = \mathbf{B}\mathbf{S}_p\mathbf{B}^{\mathrm{T}}$ where $\mathbf{B}$ is an $n \times m$ prolongation matrix of rank m that maps the dofs on the loaded surface onto global dofs, and $\mathbf{S}_p$ is an $m \times m$ positive definite matrix. The PSD matrix $\mathbf{S}_y$ has dimensions $n \times n$, where n can be very large. Note that it is not feasible storing $\mathbf{S}_y$, since it is a dense matrix of very high dimensions: n can be of the order of 100,000. On the other hand, $\mathbf{S}_p$ can be stored explicitly, since its size is typically much lower, e.g. of the order of $m = 1000, \ldots, 10,000$. The goal is to approximate

$$\mathbf{S}_y \approx \mathbf{W}\mathbf{D}\mathbf{W}^H, \quad \mathbf{D} \in \mathbf{R}^{r\times r}, \quad \mathbf{W} \in \mathbf{C}^{n\times r}, \tag{2}$$

with $\mathbf{D}$ a diagonal matrix. The number of columns of $\mathbf{W}$ is the rank of the approximation. Operations on $\mathbf{S}_y$ use the factored form (2). In order to reduce the storage and computational costs, we want r to be as small as possible, i.e. $r \ll n$ and, if possible, $r \ll m$.

In earlier work [6], the pseudo load-case method was proposed that aims at producing a low rank approximation of $\mathbf{S}_y$ from a low rank partial eigendecomposition of $\mathbf{S}_p$. In this particular context, we propose a new method that performs a partial eigendecomposition of $\mathbf{S}_y$. The proposed method is leading to significant improvements versus the pseudo-load case method.

The plan of the paper is as follows. Section 2 overviews the QR and Lanczos methods for eigenvalue problems. In the Sections 3 and 4, we present two groups of methods for computing (2). In Section 3, we present the pseudo-load case method, which has shown good performance for low frequencies and in particular for diffuse fields [6]. Experience for turbulent boundary layers using the Corcos [7] and Goody [8] models has shown that r approaches m, especially for higher frequencies, and the computational and storage costs of the method become high. In Section 4, we present a new method based on the direct decomposition of $\mathbf{S}_y$ using the Lanczos method. The motivation is given by the spectral analysis of $\mathbf{S}_y$ in Section 5. A numerical example using the software ACTRAN [9] is shown in Section 6. The proposed example involves a realistic random excitation.

We denote by $\mathbf{x}^H$ the Hermitian transpose of the vector (or matrix) $\mathbf{x}$. The Euclidean inner product of vectors $\mathbf{x}$ and $\mathbf{y}$ is denoted by $\mathbf{y}^H\mathbf{x}$ and the induced norm by $\|\mathbf{x}\|$.

2. Algebraic eigenvalue solvers

In this section, we give an introduction to solvers to the algebraic eigenvalue problem

$$\mathbf{A}\mathbf{X} = \mathbf{X}\mathbf{\Lambda},$$

where $\mathbf{A}$ is an $n \times n$ Hermitian matrix, $\mathbf{\Lambda}$ is a diagonal matrix with the diagonal elements being the eigenvalues of $\mathbf{A}$ and the columns of $\mathbf{X}$ the associated eigenvectors. Since $\mathbf{A}$ is Hermitian, $\mathbf{X}^H\mathbf{X} = \mathbf{I}$. Hence, $\mathbf{A} = \mathbf{X}\mathbf{\Lambda}\mathbf{X}^H$. When $\mathbf{X}$ has n rows and p columns and $\mathbf{\Lambda}$ is $p \times p$, $\mathbf{A}\mathbf{X} = \mathbf{X}\mathbf{\Lambda}$ is called a partial eigendecomposition.

Theory and practice of the QR method is described in Refs. [10]. It is a method that computes a full decomposition, i.e. we always compute $p = n$ eigenpairs. The matrix $\mathbf{X}$ and the diagonal of $\mathbf{\Lambda}$ need to be stored. The storage of the $n \times n$ complex matrix $\mathbf{X}$ is an important cost if n is large. The matrix $\mathbf{A}$ is first reduced to tridiagonal form. The eigenvalues of this tridiagonal matrix are then computed by the QR method. The work of the reduction to tridiagonal form is of the order of n^3, which is a very high cost when n is large. The matrix $\mathbf{A}$ needs to be available in an explicit dense or banded format for the reduction to tridiagonal form. FORTRAN subroutines for dense linear algebra are available in the LAPACK software package [11]. The corresponding Matlab function is `eig`.

The Lanczos method [12] computes a partial eigendecomposition. It transforms the $n \times n$ matrix $\mathbf{A}$ to a tridiagonal $p \times p$ matrix, which is much cheaper than the QR method, when $p \ll n$. The following is an

algorithm for the Lanczos method:

Algorithm 1 [Lanczos]

1. *Select a random starting vector* $\mathbf{v}_1$ *so that* $\|\mathbf{v}_1\| = 1$.
2. *Let* $\beta_0 = 0$ *and* $\mathbf{v}_0 = 0$.
3. *For* $j = 1, \ldots, k$ *do*:
   - 3.1. *Compute* $\mathbf{w}_j = \mathbf{A}\mathbf{v}_j$.
   - 3.2. *Update* $\tilde{\mathbf{w}}_j = \mathbf{w}_j - \beta_{j-1}\mathbf{v}_{j-1}$.
   - 3.3. *Compute* $\alpha_j = \mathbf{v}_j^H \tilde{\mathbf{w}}_j$.
   - 3.4. *Update* $\hat{\mathbf{w}}_j = \tilde{\mathbf{w}}_j - \alpha_j \mathbf{v}_j$.
   - 3.5. *Compute* $\beta_j = \|\hat{\mathbf{w}}_j\|$.
   - 3.6. *Scale* $v_{j+1} = \hat{\mathbf{w}}_j / \beta_j$.

The most expensive part is the matrix vector product with $\mathbf{A}$ in Step 3.1. Steps 3.2–3.6 compute coefficients α_j and β_j using the Euclidean inner product and the induced norm. The vector $\mathbf{v}_1, \ldots, \mathbf{v}_k$ form an orthogonal basis of the Krylov space

$$\operatorname{span}\{\mathbf{v}_1, \mathbf{A}\mathbf{v}_1, \ldots, \mathbf{A}^{k-1}\mathbf{v}_1\}.$$

The matrix $\mathbf{A}$ does not have to be available in explicit format, since the only operation is a matrix–vector product. The coefficients α_j and β_j are collected in a tri-diagonal matrix

$$\mathbf{T}_k = \begin{bmatrix} \alpha_1 & \beta_1 & & \\ \beta_1 & \ddots & \ddots & \\ & \ddots & \ddots & \beta_{k-1} \\ & & \beta_{k-1} & \alpha_k \end{bmatrix}.$$

The eigenvalues of $\mathbf{T}_k$ are approximate eigenvalues of $\mathbf{A}$.

Other inner products can be used, e.g. Steps 3.3 and 3.5 are often replaced by

Algorithm 2

- 3.3. *Compute* $\alpha_j = \mathbf{v}_j^H \mathbf{C}\tilde{\mathbf{w}}_j$,
- 3.5. *Compute* $\beta_j = \sqrt{\hat{\mathbf{w}}_j^H \mathbf{C}\hat{\mathbf{w}}_j}$,

where $\mathbf{C}$ is a positive definite symmetric matrix. With $\mathbf{C} = \mathbf{I}$, we obtain the Euclidean inner product. Such inner products are, e.g. used in the Lanczos method for computing the eigenmodes of the undamped problem $\mathbf{K}\mathbf{u} = \lambda\mathbf{M}\mathbf{u}$ [13]. The storage cost is of the order np where p is the number of vectors stored.

The storage of the vectors is the major memory cost of the Lanczos method. In this work, we use the implicitly restarted Lanczos method. It is a variation on Lanczos' method, where the basis of Lanczos vectors is compressed from time to time by throwing away a subspace that has a small contribution to the convergence. This is a technique that is used to keep the memory consumption low. We refer to the literature [14,15] for technical details. Excellent software is available in ARPACK [15]. The corresponding Matlab function is `eigs`.

In practice, we often use a block version of the Lanczos method [16,17,13]. In a block version, we multiply several vectors $\mathbf{v}_j$ with $\mathbf{A}$ at once rather than in a sequence. This usually leads to higher performance thanks to the ability to use BLAS3 algebraic kernels [18]. We denote the implicitly restarted block Lanczos method by IRBLM.

Clearly, when n is large and/or $\mathbf{A}$ is not available in an explicit form, only the Lanczos method is a viable alternative to the QR method.

3. Pseudo load case method

The conventional technique approximates $\mathbf{S}_p$ by the partial dominant eigendecomposition $\mathbf{PDP}^H$ where $\mathbf{D} \in \mathbf{R}^{r\times r}$ and $\mathbf{P} \in \mathbf{C}^{m\times r}$. Next, we compute $\mathbf{W} = \bar{\mathbf{H}}(\omega)\mathbf{L}$ as the solution of $\mathbf{Z}\bar{\mathbf{W}} = \mathbf{L}$, where $\mathbf{L} = \mathbf{BP}$ can be considered as a matrix of r pseudo load cases. $\mathbf{S}_y$ is then approximated by Eq. (2).

This procedure is very efficient when $\mathbf{S}_p$ can be approximated by a low rank matrix, and when the computation of $\mathbf{P}$ and $\mathbf{D}$ is cheap. This assumes that there are only a few large eigenvalues of $\mathbf{S}_p$. Unfortunately, the latter is not really true for turbulent layers or higher frequencies, as we will show by examples.

Computing $\mathbf{P}$ and $\mathbf{D}$ is not expensive when $r \ll m$. We can use the (block) Lanczos method with implicit restarting discussed in Section 2. When r approaches m, the Lanczos method is no longer most efficient. In this case, we use the QR method [10], e.g. as implemented in LAPACK [11].

4. Lanczos method for $\mathbf{S}_y$

In this section, we show two algorithms using the Lanczos method for computing a low rank approximation of $\mathbf{S}_y$. Since $\mathbf{S}_y$ cannot be stored in an explicit format, the QR method cannot be used.

We propose two versions of the Lanczos method. First, the direct Lanczos method applies the method to $\mathbf{S}_y$. The reduced Lanczos method solves a smaller equivalent problem, which needs less memory. The drawback, however, is a higher computational cost. This will be explained further in this paragraph and illustrated by numerical examples in Section 6.

4.1. Direct Lanczos method

The implicitly restarted block Lanczos method (IRBLM) discussed in Section 2 is implemented in ACTRAN [9]. This method computes an eigenbasis while keeping the memory consumption relatively low. The most expensive operation is the multiplication $\mathbf{w}_j = \mathbf{S}_y\mathbf{v}_j$ where $\mathbf{v}_j$ and $\mathbf{w}_j$ are vectors. Since $\mathbf{S}_y = \bar{\mathbf{Z}}^{-1}\mathbf{B}\mathbf{S}_p\mathbf{B}^H\mathbf{Z}^{-\mathrm{T}}$, this requires:

- a multiplication with $\mathbf{Z}^{-\mathrm{T}}$ which is implemented as a solve with $\mathbf{Z}^{\mathrm{T}}$,
- a matrix product with $\mathbf{B}^H$,
- a matrix product with $\mathbf{S}_p$,
- a matrix product with $\mathbf{B}$, and
- a solve with $\bar{\mathbf{Z}}$.

The storage requirements can be high, since vectors of size n have to be stored; this number should be at least $2r$ to make the IRBLM efficient. This heuristic choice is motivated by the fact that more than r vectors are needed to compute r eigenvectors and that after implicit restarting more than r vectors should remain in the basis. For more details in the choice of parameters in the implicitly restarted Lanczos method, we refer to the ARPACK manual [15].

4.2. Reduced Lanczos method

The spectrum of $\mathbf{S}_y$ consists of $n-m$ zeroes and m positive eigenvalues. We are not interested in the nullspace of $\mathbf{S}_y$. So, if we find an easy way to work with the m-dimensional range space only, we reduce the dimension of the problem from n to m and also the size of the Lanczos vectors.

Define the $n \times m$ matrix $\tilde{\mathbf{B}} = \bar{\mathbf{Z}}^{-1}\mathbf{B}$, then $\mathbf{S}_y = \tilde{\mathbf{B}}\mathbf{S}_p\tilde{\mathbf{B}}^H$. From Appendix A, we conclude that if

$$\tilde{\mathbf{B}}^H\tilde{\mathbf{B}}\mathbf{S}_p\mathbf{Y} = \mathbf{Y}\mathbf{\Lambda}, \tag{3}$$

with $\mathbf{\Lambda}$ diagonal and $\mathbf{Y}$ full rank, then

$$\mathbf{S}_y\mathbf{U} = \mathbf{U}\mathbf{\Lambda} \quad \text{and} \quad \mathbf{U} = \tilde{\mathbf{B}}\mathbf{S}_p\mathbf{Y}, \tag{4}$$

where $\mathbf{U}$ has full rank. We also derive that all nonzero eigenvalues of $\mathbf{S}_y$ are eigenvalues of $\tilde{\mathbf{B}}^H\tilde{\mathbf{B}}\mathbf{S}_p$.

Since $\mathbf{S}_y$ is Hermitian, $\mathbf{U}^H\mathbf{U}$ is diagonal. The positive-definiteness of $\tilde{\mathbf{B}}^H\tilde{\mathbf{B}}$ and $\mathbf{S}_p$ results in diagonal $Y^H(\tilde{\mathbf{B}}^H\tilde{\mathbf{B}})^{-1}\mathbf{Y}$ and $\mathbf{Y}^H\mathbf{S}_p\mathbf{Y}$.

As a conclusion, Eq. (3) can be viewed as a symmetric eigenvalue problem in a subspace with inner product $(\mathbf{y},\mathbf{x}) = \mathbf{x}^H\mathbf{S}_p\mathbf{y}$. This allows for the use of the Lanczos method with $\mathbf{S}_p$ inner product to compute $\mathbf{\Lambda}$.

Once $\mathbf{\Lambda}$ and $\mathbf{Y}$ have been computed, we can compute $\mathbf{U}$ from the second equation in Eq. (4).

It is important to note that the speed of convergence of the direct and reduced Lanczos methods is the same, since $\mathbf{S}_y$ and $\tilde{\mathbf{B}}^H\tilde{\mathbf{B}}\mathbf{S}_p$ have the same nonzero eigenvalues. The inner product in the Lanczos method using $\mathbf{C} = \mathbf{S}_p$ is rather expensive. In addition, the vectors $\mathbf{U}$ are computed explicitly as $\mathbf{U} = \tilde{\mathbf{B}}\mathbf{S}_p\mathbf{Y}$ in a post-processing step, which requires a linear solve with $\tilde{\mathbf{Z}}$. This shows that the Reduced Lanczos method requires more computation time than the Direct Lanczos method. But the storage cost of the vectors is significantly lower. The vectors have size m rather than n, where m is the number of dofs on the (randomly) loaded surface. Usually $m \ll n$. So, the reduced Lanczos method should only be used when the storage of Lanczos vectors becomes prohibitive in the direct Lanczos method.

5. Spectral analysis

The key argument for using the methods proposed in Section 4 is that $\mathbf{S}_y$ can be well approximated by a matrix of low rank. The theory in this section explains that even if the spectrum of $\mathbf{S}_p$ is dense, so that the pseudo load case method produces $r \approx m$, the direct decomposition of $\mathbf{S}_y$ may lead to small r.

The spectral properties are best illustrated by an example. Consider a steel plate excited by a boundary layer following the Corcos model [7]. Fig. 1 shows the spectra of $\mathbf{S}_x$ and $\mathbf{S}_y$ for two frequencies. We notice the following: the spectrum of $\mathbf{S}_x$ becomes denser for higher frequencies. The spectrum of $\mathbf{S}_y$ drops to zero much faster than the spectrum of $\mathbf{S}_x$.

This shows that r is lower when the eigendecomposition of $\mathbf{S}_y$ is used rather than the one of $\mathbf{S}_p$.

The remainder of this section is devoted to a more formal explanation for this behaviour.

The matrix $\mathbf{S}_p$ becomes more and more diagonally dominant for higher frequencies since the excitation becomes more spatially uncorrelated. This explains why the eigenvalues of $\mathbf{S}_p$ lie closer to each other for higher frequencies: it also makes the pseudo load case method more expensive. There is an interesting physical interpretation here: although the excitation is spatially uncorrelated, the response is highly correlated in space, since $\mathbf{S}_y$ has a few very large eigenvalues. We now explain why $\mathbf{S}_y$ has a few large eigenvalues.

Note that, from linear algebra theory, among all approximations of rank r to $\mathbf{S}_y$, the partial eigendecomposition produces the smallest two-norm of the error. So, in theory it is always better to perform a direct decomposition of $\mathbf{S}_y$ than using a decomposition of $\mathbf{S}_p$.

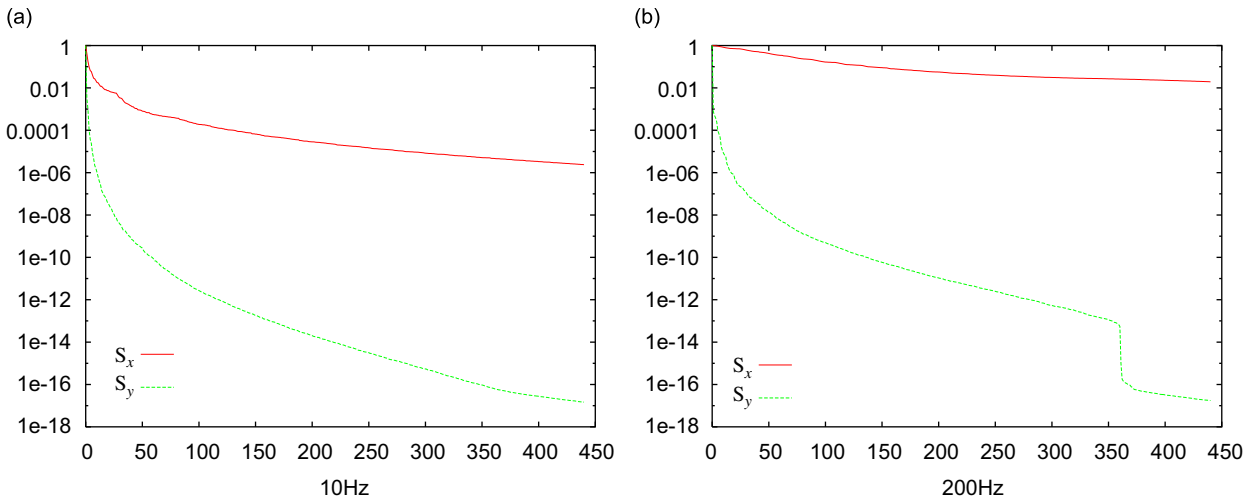

Fig. 1. Spectra of $\mathbf{S}_x$ (dashed line, i.e. lower line) and $\mathbf{S}_y$ (solid line, i.e. upper line) for two frequencies. The figures show the eigenvalue index on the horizontal axis and the value on the vertical axis. (a) shows the eigenvalues for 10 Hz and (b) for 200 Hz.

In this section, we give another argument for finite-element models with modal damping, Rayleigh and proportional damping. We assume that $\mathbf{Z}$ takes the form

$$\mathbf{Z} = \mathbf{U}^{-*}\mathrm{diag}(\zeta_i)\mathbf{U}^{-1},$$

where $\mathbf{U} = [\mathbf{u}_1, \ldots, \mathbf{u}_n]$ are the eigenvectors of the undamped eigenvalue problem

$$\mathbf{KU} = \mathbf{MU}\boldsymbol{\Omega}^2 \quad \text{with} \quad \mathbf{U}^H\mathbf{MU} = \mathbf{I}. \tag{5}$$

The coefficient ζ_j then usually takes the form

$$\zeta_j = \kappa_j + \mathrm{i}\omega\delta_j - \omega^2.$$

In Appendix A, we show that the m nonzero eigenvalues of $\mathbf{S}_y$ and the eigenvalues of $\tilde{\mathbf{B}}^H\tilde{\mathbf{B}}\mathbf{S}_p$ match. So, for the spectral analysis, we study $\tilde{\mathbf{B}}^H\tilde{\mathbf{B}}\mathbf{S}_p$.

The difficulty here is that we do not know the eigenvalues of $\tilde{\mathbf{B}}^H\tilde{\mathbf{B}}$. The eigenvalues of $\tilde{\mathbf{B}}^H\tilde{\mathbf{B}}$ are the squared singular values of $\tilde{\mathbf{B}} = \bar{\mathbf{Z}}^{-1}\mathbf{B}$. Hence, we have

$$\tilde{\mathbf{B}} = \sum_{j=1}^{n} \frac{\mathbf{u}_j\mathbf{u}_j^H\mathbf{B}}{\zeta_j}. \tag{6}$$

Usually, the sum is made over the terms with smallest ζ_j, i.e. for the κ_j's near ω^2. This is also the approach that is followed by the modal truncation method. So, we can find an s so that

$$\tilde{\mathbf{B}} \approx \tilde{\mathbf{B}}_s = \sum_{j=1}^{s} \frac{\mathbf{u}_j\mathbf{u}_j^H\mathbf{B}}{\zeta_j}.$$

Since $\bar{\mathbf{Z}}^{-1}\mathbf{B}$ is dominated by the modes nearest ω, it is to be expected that s is small. Define

$$\mathbf{E}_s = \sum_{j=s+1}^{n} \frac{\mathbf{u}_j\mathbf{u}_j^H\mathbf{B}}{\zeta_j}$$

so that $\tilde{\mathbf{B}} = \tilde{\mathbf{B}}_s + \mathbf{E}_s$.

If $\mathbf{M}$ is a lumped mass, i.e. $\mathbf{M}$ is a diagonal matrix, then $\mathbf{u}_j^H\mathbf{u}_i = 0$ for $j \neq i$. So,

$$\tilde{\mathbf{B}}^H\tilde{\mathbf{B}} = \tilde{\mathbf{B}}_s^H\tilde{\mathbf{B}}_s + \mathbf{E}_s^H\mathbf{E}_s.$$

Hence, we conclude (see Appendix B) that

$$\frac{\|\tilde{\mathbf{B}}^H\tilde{\mathbf{B}}\mathbf{S}_p - \tilde{\mathbf{B}}_s^H\tilde{\mathbf{B}}_s\mathbf{S}_p\|_2}{\|\tilde{\mathbf{B}}^H\tilde{\mathbf{B}}\mathbf{S}_p\|_2} \leqslant \frac{\|\mathbf{E}_s\|_2^2}{\|\tilde{\mathbf{B}}_s\|_2^2} \cdot \frac{\lambda_{\max}(\mathbf{S}_p)}{\lambda_{\min}(\mathbf{S}_p)}$$

from which we derive that if $\tilde{\mathbf{B}}$ can be well approximated by a matrix of rank s, so can $\mathbf{S}_y$.

The spectrum of $\mathbf{S}_y$ is mainly dominated by the spectrum of $\tilde{\mathbf{B}}^H\tilde{\mathbf{B}}$, so the spectrum of $\mathbf{S}_y$ decays as

$$|\kappa_j + \mathrm{i}\omega\delta_j - \omega^2|^{-2}.$$

Let κ_j be ordered following increasing distance to ω. If r is determined so that

$$|\kappa_{r+1} + \mathrm{i}\omega\delta_{r+1} - \omega^2|^{-2} \leqslant \tau|\kappa_1 + \mathrm{i}\omega\delta_1 - \omega^2|^{-2},$$

we have, approximately, the following bound to the approximation error:

$$\|\mathbf{S}_y - \mathbf{W}_r\mathbf{D}_r\mathbf{W}_r^H\|_2 \leqslant \tau\|\mathbf{S}_y\|_2.$$

Since the function $(\kappa - \omega^2)^{-2}$ decays slower when κ is further away from ω, smaller τ may require a much larger r.

The eigenvectors of Eq. (5) play a role in the low rank approximation of $\mathbf{S}_y$ in a similar way as modal truncation. The situation may be somehow better, since only the modes that have a contribution on the faces with the random excitation, will be taken into account.

6. Numerical example

We compare the following methods:

PLQR the pseudo load case method where the QR method is used to decompose $\mathbf{S}_p$,
PLL the pseudo load case method where the Lanczos method is used to decompose $\mathbf{S}_p$,
DL the direct Lanczos method, i.e. the Lanczos method applied to $\mathbf{S}_y$,
RL the reduced Lanczos method, i.e. the Lanczos method applied to $\tilde{\mathbf{B}}^H\tilde{\mathbf{B}}\mathbf{S}_p$.

The rank r was determined so that the error on the rank r approximations to $\mathbf{S}_p$ and $\mathbf{S}_y$, respectively have a relative error bounded from above by $\tau = 10^{-4}$. For the DL and RL methods, we then have

$$\|\mathbf{S}_y - \mathbf{W}_r\mathbf{D}_r\mathbf{W}_r^H\|_2 \leqslant \tau\|\mathbf{S}_y\|_2$$

and for the PLQR and PLL methods, we have

$$\|\mathbf{S}_p - \mathbf{P}_r\mathbf{D}_r\mathbf{P}_r^H\|_2 \leqslant \tau\|\mathbf{S}_p\|_2.$$

The computations were carried out on an Opteron running the Suse 9.2 linux operating system with 4 GB of core memory.

We are computing $\mathbf{S}_y$ for an application whose mesh is shown in Fig. 2. The mesh is a quarter of two connected cylinders. The figure shows only half a single cylinder. The distributed random excitation is related to a turbulent boundary layer acting on the external cylindrical boundary. Such dynamic FE model is targeted for 'low-frequency' computations (i.e. for dynamic responses dominated by a limited number of modes). The example involves a realistic random excitation.

We solve the problem for a coarse and a fine discretization. The coarse discretization is used to illustrate the difference in r depending on whether $\mathbf{S}_p$ or $\mathbf{S}_y$ are decomposed. We therefore only test the methods PLQR and DL, since we only want to show the difference between the decomposition of $\mathbf{S}_p$ and $\mathbf{S}_y$. In fact, PLQR and PLL produce the same r. Similarly, DL and RL produce the same r. The computations on the fine mesh are more expensive and are used for performance testing of the four methods. Also, we compare the difference in r for the coarse and fine meshes, which also leads to interesting conclusions.

6.1. The coarse discretization

The mesh considered for the coarse discretization consists of 690 solid shell elements and 5138 nodes. The number of dofs is $n = 15,414$ and the number of dofs along the randomly loaded boundary is $m = 2193$.

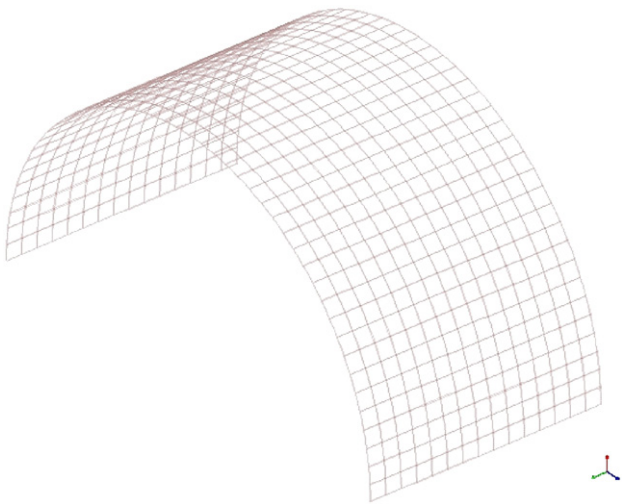

Fig. 2. Coarse mesh of the quarter cylinder.

Table 1
Rank r for the coarse grid quarter cylinder problem

Frequency (Hz)	PLQR	DL	Frequency (Hz)	PLQR	DL
100	364	40	250	1015	10
110	411	41	260	1062	24
120	441	40	270	1109	28
130	474	41	280	1155	55
140	512	41	290	1199	40
150	569	41	300	1244	19
160	616	40	310	1297	57
170	648	39	320	1345	63
180	689	36	330	1389	26
190	741	32	340	1444	50
200	793	24	350	1486	23
210	826	7	360	1528	41
220	873	19	370	1589	4
230	928	22	380	1627	69
240	964	21	390	1670	100
			400	1717	77

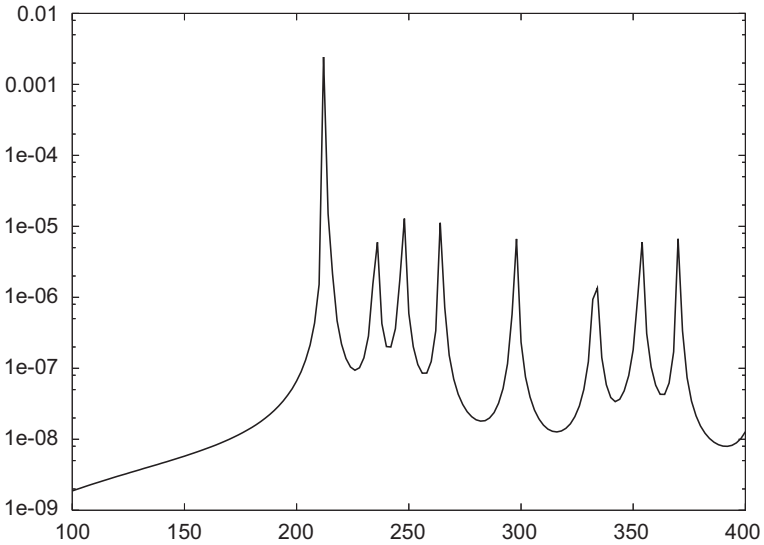

Fig. 3. Spectral density of the mean square velocity. The horizontal axis contains frequency (Hz), the vertical axis the PSD $((m^2/s^2)/Hz)$.

Table 1 shows the rank r for the pseudo load case (PLQR) and the Lanczos (DL) methods in function of the frequency.

From Table 1, we see the impressive difference in r between the two approaches for the same tolerance. Moreover, r approaches m for the higher frequencies for the PLQR method.

Fig. 3 shows the PSD of mean square velocity for this problem. Note that there is no visual difference between the results computed by the two methods.

6.2. The fine discretization

For a comparison in performance, we compare the results for the same problem, but with a finer mesh. The mesh consists of 2760 solid shell elements and 19,933 nodes. The number of dofs is $n = 59,799$ and the number of dofs along the randomly loaded boundary is $m = 8525$. We compared the two methods for the

Table 2
Rank r and computation times for the fine grid quarter cylinder problem

	Pseudo-load methods		Direct methods	
	PLQR	PLL	DL	RL
100 Hz				
rank r	425	425	40	40
time (min)	214	16	2.7	5.5
400 Hz				
rank r	2383	2383	78	78
time (min)	232	248	3.3	6.3

Table 3
Comparison of r and computation times in function of the tolerance τ for the fine grid model

	PLL 100 Hz		DL 400 Hz	
τ	r	time (min)	r	time (min)
10^{-2}	32	2.5	12	2.2
10^{-3}	141	5.0	36	2.8
10^{-4}	425	15.6	78	3.3
10^{-5}	–	–	141	3.3

frequencies 100 and 400. Table 2 summarizes our results. We used $p = 140$ Lanczos vectors in the implicitly restarted block Lanczos method with a block size 5. The reduced Lanczos method is more expensive, since a post-processing step is required to compute the eigenvectors of $\mathbf{S}_y$ from the eigenvectors of $\tilde{\mathbf{B}}^H \tilde{\mathbf{B}} \mathbf{S}_p$, see Section 4.2. The storage of the p Lanczos vectors of length n for the DL method requires $np = 8,371,860$ complex values. For the RL method, the cost is $mp = 1,193,500$. The storage cost of the DL method is about a factor seven of the cost of the RL method.

Another interesting conclusion, when we compare Tables 1 and 2, is that the number r does not change for the DL method when the mesh is refined. The computation time for the pseudo load case methods is mainly due to the QR method for PLQR and the Lanczos method for PLL. Note that the PLL method becomes significantly more expensive for higher frequencies.

6.3. Comparison for different tolerances

We compare for the same problem the influence of the tolerance τ. Table 3 compares the ranks and computation times for the PLL and DL methods. We use a low frequency for the PLL method and a large frequency for the DL method. It follows that the increase of the number of vectors is exponential. Requesting a smaller tolerance implies a larger r and so, a higher storage cost.

7. Conclusions

The evaluation of the random vibro-acoustic response of mechanical structures has been studied by addressing the specific issues related to the distributed nature of the excitation. Both direct and reduced Lanczos methods have been investigated in this context and show excellent convergence properties for the studied problem. In particular, we found that the direct decomposition of $\mathbf{S}_y$ leads to a faster method, requiring a smaller rank than the pseudo load case method.

The reduced Lanczos method (RL) is more expensive in computation time than the direct Lanczos method (DL), but cheaper in memory, as explained in Section 4.2. When memory consumption is not a limiting factor in the simulation, the DL method should be used, since it is significantly faster.

The conclusions only hold for problems where the pseudo-load case method produces a large r. It is an open question whether the proposed method performs better for other types of random excitations than the one used in the numerical example.

Acknowledgements

This paper presents research results of the Belgian Network DYSCO (Dynamical Systems, Control, and Optimization), funded by the Interuniversity Attraction Poles Programme, initiated by the Belgian State, Science Policy Office. The scientific responsibility rests with its author(s).

Appendix A

In this appendix, we give the mathematical proof for the correspondence of the m nonzero eigenvalues of $\mathbf{S}_y$ and the eigenvalues of $\tilde{\mathbf{B}}^H\tilde{\mathbf{B}}\mathbf{S}_p$.

Note that $\tilde{\mathbf{B}}$ is a rank m matrix. Then, the eigenvalue problem $\mathbf{S}_y\mathbf{u}=\lambda\mathbf{u}$ has exactly m non-zero eigenvalues.

It follows that when $\lambda\neq 0$, $\mathbf{u}$ lies in the range of $\tilde{\mathbf{B}}$: there is a unique $\mathbf{v}$ so that $\mathbf{u}=\tilde{\mathbf{B}}\mathbf{v}$. When $\lambda=0$, it follows that $\mathbf{u}$ lies in the nullspace of $\tilde{\mathbf{B}}^H$.

Multiplying $\mathbf{S}_y\mathbf{u}=\lambda\mathbf{u}$ on the left by $\tilde{\mathbf{B}}^H$ produces

$$\tilde{\mathbf{B}}^H\tilde{\mathbf{B}}\mathbf{S}_p\tilde{\mathbf{B}}^H\mathbf{u}=\lambda\tilde{\mathbf{B}}^H\mathbf{u}$$

$$\tilde{\mathbf{B}}^H\tilde{\mathbf{B}}\mathbf{S}_p\mathbf{v}=\lambda\mathbf{v}$$

with $\mathbf{v}=\tilde{\mathbf{B}}^H\mathbf{u}\neq 0$, from which we derive that if $(\lambda,\mathbf{u})$ is an eigenpair of $\mathbf{S}_y$ with $\lambda\neq 0$, then λ is an eigenvalue of

$$\tilde{\mathbf{B}}^H\tilde{\mathbf{B}}\mathbf{S}_p\mathbf{v}=\lambda\mathbf{v}.$$

This is the eigenvalue problem projected on the response from the random excitations and therefore has dimension m.

Appendix B

Let

$$\tilde{\mathbf{B}}^H\tilde{\mathbf{B}}-\tilde{\mathbf{B}}_s^H\tilde{\mathbf{B}}_s=\mathbf{E}_s^H\mathbf{E}_s.$$

In this appendix, we prove that

$$\frac{\|\tilde{\mathbf{B}}^H\tilde{\mathbf{B}}\mathbf{S}_p-\tilde{\mathbf{B}}_s^H\tilde{\mathbf{B}}_s\mathbf{S}_p\|_2}{\|\tilde{\mathbf{B}}^H\tilde{\mathbf{B}}\mathbf{S}_p\|_2}\leqslant\frac{\|\mathbf{E}_s\|_2^2}{\|\tilde{\mathbf{B}}_s\|_2^2}\cdot\frac{\lambda_{\max}(\mathbf{S}_p)}{\lambda_{\min}(\mathbf{S}_p)}$$

For the denominator, note that $\|\tilde{\mathbf{B}}^H\tilde{\mathbf{B}}\|_2\geqslant\|\tilde{\mathbf{B}}_s\|_2^2$ since $\tilde{\mathbf{B}}_s^H\tilde{\mathbf{B}}_s$ and $\mathbf{E}_s^H\mathbf{E}_s$ are both positive (semi) definite terms. Also, $\|\tilde{\mathbf{B}}^H\tilde{\mathbf{B}}\mathbf{S}_p\|_2\geqslant\|\tilde{\mathbf{B}}^H\tilde{\mathbf{B}}\|_2\lambda_{\min}(\mathbf{S}_p)$. For the nominator,

$$\tilde{\mathbf{B}}^H\tilde{\mathbf{B}}\mathbf{S}_p-\tilde{\mathbf{B}}_s^H\tilde{\mathbf{B}}_s\mathbf{S}_p=\mathbf{E}_s^H\mathbf{E}_s\mathbf{S}_p$$

so,

$$\|\tilde{\mathbf{B}}^H\tilde{\mathbf{B}}\mathbf{S}_p-\tilde{\mathbf{B}}_s^H\tilde{\mathbf{B}}_s\mathbf{S}_p\|_2\leqslant\|\mathbf{E}_s\|_2^2\|\mathbf{S}_p\|_2\ ,$$

which concludes the proof.

References

[1] R. Clough, J. Penzien, *Dynamics of Structures*, McGraw-Hill, Kogakusha, Tokyo, 1975.
[2] I. Elishakoff, *Probabilistic Methods in the Theory of Structures*, Wiley, New York, 1983.
[3] Y. Lin, *Probabilistic Theory of Structural Dynamics*, Krieger Publishing Company, Malabar, 1876.
[4] N. Nigam, S. Narayanan, *Applications of Random Vibrations*, Springer, New York, 1994 ISBN 0-387-19861-X.
[5] C. Yang, *Random Vibrations of Structures*, Wiley, New York, 1986 ISBN 0-471-80262-X.
[6] J. Coyette, T. Knapen, G. Lielens, K. Meerbergen, P. Ploumhans, Simulation of randomly excited acoustic insulation systems using finite element approaches, *ICA Conference*, Kyoto, April 4–9, 2004.
[7] G. Corcos, Resolution of pressure in turbulence, *Journal of the Acoustical Society of America* 35 (2) (1963) 192–199.
[8] M. Goody, An empirical spectral model of surface-pressure fluctuations that includes Reynolds number effects, AIAA Paper 2002–2565, *8th AIAA/CEAS Aeroacoustics Conference and Exhibit*, Breckenridge, CO, June 17–19, 2002.
[9] Free Field Technologies, MSC Actran 2006, User's Manual, 2006.
[10] G. Golub, C. Van Loan, *Matrix computations*, third ed., The Johns Hopkins University Press, 1996.
[11] E. Anderson, Z. Bai, C. Bischof, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, S. Ostrouchov, D. Sorensen, *LAPACK Users' Guide*, SIAM, Philadelphia, PA, USA, 1995.
[12] C. Lanczos, An iteration method for the solution of the eigenvalue problem of linear differential and integral operators, *Journal of Research of the National Bureau of Standards* 45 (1950) 255–282.
[13] R. Grimes, J. Lewis, H. Simon, A shifted block Lanczos algorithm for solving sparse symmetric generalized eigenproblems, *SIAM Journal on Matrix Analysis and Applications* 15 (1994) 228–272.
[14] D. Sorensen, Implicit application of polynomial filters in a k-step Arnoldi method, *SIAM Journal Matrix Analysis and Applications* 13 (1992) 357–385.
[15] R.B. Lehoucq, D.C. Sorensen, C. Yang, *ARPACK USERS GUIDE: Solution of Large Scale Eigenvalue Problems with Implicitly Restarted Arnoldi Methods*, SIAM, Philadelphia, PA, USA, 1998.
[16] J. Cullum, R. Willoughby, *Lanczos Algorithms for Large Symmetric Eigenvalue Computations*, Vol. 1, *Theory*, Birkhäuser, Boston, 1985.
[17] J. Cullum, R. Willoughby, *Lanczos Algorithms for Large Symmetric Eigenvalue Computations*. Vol. 2, *Users Guide*, Birkhäuser, Boston, 1985.
[18] J.J. Dongarra, J. DuCroz, I.S. Duff, S. Hammarling, A set of Level 3 Basic Linear Algebra Subprograms, *ACM Transactions on Mathematical Software* 16 (1) (1990) 1–17.