A comparison of stability computational methods for periodic solution of nonlinear problems with application to rotordynamics

Loïc Peletan, Sébastien Baguet, Mohamed Torkhani, Georges Jacquet-Richardet

Nonlinear Dyn - Author's accepted manuscript
The final publication is available at Springer via http://dx.doi.org/10.1007/s11071-012-0744-0

A comparison of stability computational methods for periodic solution of nonlinear problems with application to rotordynamics

Loïc Peletan · Sébastien Baguet · Mohamed Torkhani · Georges Jacquet-Richardet

Received: date / Accepted: date

Abstract In this paper a comparative study of five different stability computational methods based on the Floquet theory is presented. These methods are compared in terms of accuracy and CPU performance. Test are performed on a set of nonlinear problems relevant to rotating machinery with rotor-to-stator contact and a variable number of degrees of freedom, whose periodic solutions are computed with the Harmonic Balance Method (HBM).

Keywords Harmonic Balance Method · stability · Floquet theory · rotor nonlinear dynamics · rotor-stator contact

1 Introduction

One of the main challenges for turbomachinery designers is to improve the efficiency of machines, whilst ensuring the safety during operation. The steady state response of the system as well as its stability are thus of prime interest.

As turbomachinery often operates at a constant rotational speed, time-domain methods such as the shooting method [1] and frequency-domain methods such as the trigonometric collocation method [2] or the harmonic balance method (HBM) [3] are commonly used to predict the steady state response of the system. It is well established that such techniques offer a significantly reduced computational time, when compared to transient time marching solutions.

L. Peletan · S. Baguet · G. Jacquet-Richardet
Université de Lyon, CNRS, INSA-Lyon, LaMCoS UMR5259, F-69621, Villeurbanne cedex, France
E-mail: loic.peletan@insa-lyon.fr
E-mail: sebastien.baguet@insa-lyon.fr
E-mail: georges.jacquet@insa-lyon.fr

M. Torkhani
LaMSID UMR EDF-CNRS-CEA 2832, EDF R&D, F-92141 Clamart Cedex, France
E-mail: mohamed.torkhani@edf.fr

Concerning stability the choice for a method is not so clear. The computation of rotor stability has yielded a vast amount of literature in the past, with emphasis on linear systems with time periodic coefficients such as asymmetric rotors, rotors subjected to aerodynamic effects or on-board rotors with periodic motion of the support [4–8]. In the case of nonlinear systems (systems with rotor-stator or blade-casing contact, nonlinear bearings, ...) several stable and unstable periodic responses can coexist at a given rotational speed and appropriate numerical methods must be used in order to obtain all solutions and distinguish between stable and unstable ones.

The Floquet theory and the Lyapunov exponents are certainly the most popular approaches used to assess the local stability of periodic solutions. Several different implementations of these theories have been developed and used in the past, as reviewed by Friedmann [5]. More recently, these methods have been applied to nonlinear systems. Liaw [9] studied the stability of a one-degree-of-freedom oscillator using a piecewise linear nonlinearity and the Lyapunov exponents. Later, Raghothama and Narayanan [10] and Zhen and Hasebe [11] applied this technique to rotor systems. Different variants of the Floquet theory have also been frequently used. Kim and Noah [12] studied the stability of a nonlinear Jeffcott rotor by constructing and analyzing the so-called Floquet transition or monodromy matrix, whereas Shen et al. [13] chose to approximate the monodromy matrix in their study of a Mathieu-Duffing oscillator. Hill's method, which is also a variant of the Floquet theory, was used by Rook [14] to study the stability of a Duffing oscillator and by Von Groll and Ewins [3] and Sinha [15] in rotordynamics applications.

Although the aforementioned techniques are not at all equivalent in terms of accuracy and CPU performance, such differences are rarely debated in the literature. In this paper, a comparative study of five different implementations of

the Floquet theory for nonlinear problems is presented. The Harmonic Balance Method is used to compute the nonlinear response curves and the associated CPU time serves as a reference for further comparisons. Following a brief presentation of the HBM, several stability computational method are reviewed. Then the accuracy and performance for five of them are quantitatively evaluated and discussed. For this purpose, a set of nonlinear problems relevant to rotating machinery with rotor-to-stator contact and a variable number of degrees of freedom is considered. Techniques based on the computation of the Lyapunov exponents are not discussed in this paper, as their results are not directly comparable with those given by the techniques based on the Floquet theory.

2 Harmonic Balance Method (HBM)

The Harmonic Balance Method (HBM) is a well known technique, used to compute periodic solutions for dynamic systems. This method consists in solving the equations of movement in the frequency domain, rather than in the time domain. The general time-domain equation for a dynamic mechanical system is given by:

$$\mathbf{M}\ddot{\mathbf{q}}(t)+\mathbf{C}\dot{\mathbf{q}}(t)+\mathbf{K}\mathbf{q}(t)+\mathbf{f}(\mathbf{q}(t),\dot{\mathbf{q}}(t))-\mathbf{p}(t)=\mathbf{0} \tag{1}$$

where $\mathbf{q}$ represents the displacement vector for all of the n degrees of freedom (DOFs); $\mathbf{K}$, $\mathbf{C}$ and $\mathbf{M}$ are the generalized $n\times n$ stiffness, damping and mass matrices; $\mathbf{f}$ is the nonlinear force vector and $\mathbf{p}$ is the external excitation force vector. When the external excitations are periodic, it is valid to assume that a steady state solution for Eq. (1) exists, and that this solution is also periodic. The displacements, the external and the nonlinear forces can thus be written as truncated Fourier series:

$$\mathbf{q}(t)=\mathbf{Q}_0+\sum_{j=1}^{N}\left(\mathbf{Q}_{2j-1}\cos(\frac{j}{\nu}\omega t)+\mathbf{Q}_{2j}\sin(\frac{j}{\nu}\omega t)\right) \tag{2}$$

$$\mathbf{p}(t)=\mathbf{P}_0+\sum_{j=1}^{N}\left(\mathbf{P}_{2j-1}\cos(\frac{j}{\nu}\omega t)+\mathbf{P}_{2j}\sin(\frac{j}{\nu}\omega t)\right) \tag{3}$$

$$\mathbf{f}(t)=\mathbf{F}_0+\sum_{j=1}^{N}\left(\mathbf{F}_{2j-1}\cos(\frac{j}{\nu}\omega t)+\mathbf{F}_{2j}\sin(\frac{j}{\nu}\omega t)\right) \tag{4}$$

$\mathbf{Q}=\left[\mathbf{Q_0}^T,\mathbf{Q_1}^T,\ldots,\mathbf{Q}_{2N}^T\right]^T$, $\mathbf{F}=\left[\mathbf{F}_0^T,\mathbf{F}_1^T,\ldots,\mathbf{F}_{2N}^T\right]^T$ and $\mathbf{P}=\left[\mathbf{P}_0^T,\mathbf{P}_1^T,\ldots,\mathbf{P}_{2N}^T\right]^T$ are the vectors of the Fourier coefficients for displacements, nonlinear forces and external excitations, respectively. ω is the fundamental frequency of the external excitation, and ν is a positive integer used for sub-harmonic responses calculation.

As described in [16], Eqs. (2) (3) and (4) can be substituted into Eq. (1), following which a Galerkin procedure is applied to transform the nonlinear differential Eq. (1), of dimension n, into an algebraic nonlinear system of equations, of dimension $n_{\mathrm{HBM}}=n(2N+1)$:

$$\mathbf{R}(\mathbf{Q},\omega)=\mathbf{Z}(\omega)\mathbf{Q}+\mathbf{F}(\mathbf{Q})-\mathbf{P}=\mathbf{0} \tag{5}$$

where $\mathbf{Z}=\mathrm{diag}\left(\mathbf{K},\mathbf{Z}_1,\ldots,\mathbf{Z}_k,\ldots,\mathbf{Z}_N\right)$ with:

$$\mathbf{Z}_k=\begin{bmatrix}\mathbf{K}-\left(\frac{k}{\nu}\right)^2\omega^2\mathbf{M} & \left(\frac{k}{\nu}\right)\omega\mathbf{C}\\ -\left(\frac{k}{\nu}\right)\omega\mathbf{C} & \mathbf{K}-\left(\frac{k}{\nu}\right)^2\omega^2\mathbf{M}\end{bmatrix}$$

Equation (5) needs to be solved for $\mathbf{Q}$. As this equation is still nonlinear, an appropriate nonlinear solver must be used to derive a correct solution. In the present paper, a Newton-Raphson iterative solver was used. It has been demonstrated that the combined use of HBM and a Newton-Raphson solver is equivalent to the Incremental Harmonic Balance Method (IHBM) [17].

The Alternating Frequency Time (AFT) algorithm [18] proves to be an efficient way to compute the nonlinear term $\mathbf{F}(\mathbf{Q})$ and its derivative $\partial\mathbf{F}(\mathbf{Q})/\partial\mathbf{Q}$ involved in the Newton-Raphson iterations . The AFT scheme uses fast direct and inverse Fourier transforms to quickly compute the nonlinear forces in the time domain and then switch back to the frequency domain. The nonlinear forces are indeed usually much easier to evaluate in the time domain than in the frequency domain. The above-mentioned algorithm is generally combined with a pseudo-arc length continuation method [1] in order to obtain both stable and unstable solutions of response curves.

3 Stability determination using the Floquet theory

There are different definitions for the stability of a dynamic solution. In the present paper we investigate what is referred to as the local stability of the solution. Assessment of the local stability consists in applying a small perturbation to the equilibrium solution, and then checking whether or not this perturbation subsides with time. The HBM converges to stable as well as unstable solutions, and there is no way to make an *a priori* evaluation of its stability. However, it can be computed a posteriori using the Floquet theory. There are currently several different algorithms based on the Floquet theory. Some of these are based on a description of the problem in the time domain, and consist in computing the eigenvalues of the so-called monodromy matrix. Others are based on a description of the problem in the frequency domain.

Some of the most commonly used algorithms are described here. It is important to emphasize that these algorithms can also be used with other solving techniques (such as the shooting method, for instance). However, the programming effort required and the CPU performance can differ widely from one method to another.

3.1 Time domain methods

3.1.1 Monodromy matrix computation by 2n-pass numerical integration (referred to as the '2n-pass' method)

In this algorithm, the equation of motion (1) is rewritten in the following state form:

$$\dot{\mathbf{x}}(t) = \mathbf{L}\mathbf{x}(t) - \tilde{\mathbf{f}}(\mathbf{x}(t)) + \tilde{\mathbf{p}}(t) \tag{6}$$

where $\mathbf{x}(t) = [\mathbf{q}(t), \dot{\mathbf{q}}(t)]^T$; $\tilde{\mathbf{f}}(\mathbf{x}(t)) = \left[\mathbf{0}, \mathbf{M}^{-1}\mathbf{f}(\mathbf{q}, \dot{\mathbf{q}})\right]^T$; $\tilde{\mathbf{p}}(t) = \left[\mathbf{0}, \mathbf{M}^{-1}\mathbf{p}(t)\right]^T$ and:

$$\mathbf{L} = \begin{bmatrix} \mathbf{0} & \mathbf{Id} \\ -\mathbf{M}^{-1}\mathbf{K} & -\mathbf{M}^{-1}\mathbf{C} \end{bmatrix}$$

where $\mathbf{Id}$ represents the identity matrix. $\mathbf{x}_0(t)$ is defined as a solution for Eq. (6), and $\mathbf{y}(t)$ as a small perturbation. $\mathbf{x}(t) = \mathbf{x}_0(t) + \mathbf{y}(t)$ is then introduced into Eq. (6) and using first order approximation the following first order differential equation is eventually obtained:

$$\dot{\mathbf{y}}(t) = \mathbf{J}(\mathbf{x}_0, t)\mathbf{y}(t) \tag{7}$$

where:

$$\mathbf{J}(\mathbf{x}_0, t) = \mathbf{L} - \left(\frac{\partial \tilde{\mathbf{f}}}{\partial \mathbf{x}}\right)_{\mathbf{x}=\mathbf{x}_0}(t)$$

$\mathbf{J}$ is T-periodic and a $2n \times 2n$ matrix $\mathbf{H}$ can be defined such that:

$$\mathbf{y}(T) = \mathbf{H}\mathbf{y}(0)$$

$\mathbf{H}$ is called the monodromy or Floquet transition matrix. It is a linear operator describing the evolution of the perturbation over one period. Its eigenvalues are called the Floquet multipliers. If any of the Floquet multipliers has a module greater than one, then the solution is unstable, otherwise it is stable.

Thus, the first approach to the computation of $\mathbf{H}$ is to do proceed column by column. The linear equation (7) has to be solved by proceeding to $2n$ time integrations over one period, using $2n$ linearly independent initial conditions, namely $\mathbf{y}_i(0) = \mathbf{e}_i, \quad i \in [1, \ldots, 2n]$, where $\mathbf{e}_i$ is a vector of zeros with a one at the i^{th} position.

$$\mathbf{H} = [\mathbf{y}_1(T), \mathbf{y}_2(T), \ldots, \mathbf{y}_{2n}(T)]$$

This algorithm requires relatively little programming effort, since there are numerous existing tools in the scientific libraries of most programming languages, which can be used to integrate first order linear ordinary differential equations. As long as the numerical integration is performed correctly, this method produces very accurate results. In the present study, the results provided by the $2n$-pass algorithm are used as a reference. However, the downside of this approach is that it can require an enormous computational effort. It should also be noted that in the case where the problem is solved using a shooting method, the monodromy matrix is a subproduct of the solving process, such that no additional operation is required to obtain this matrix.

A single-pass version of this algorithm exists, in which Eq. (7) is rewritten such that $\mathbf{J}$ is a $4n^2 \times 4n^2$ matrix and the vectors $\mathbf{y}_i(0) = \mathbf{e}_i, \quad i \in [1, \ldots, 2n]$ of size $2n$ are put together to form a $4n^2$ vector. Then the time integration is performed on a larger system but only once instead of $2n$ times. This single-pass scheme can save about 50% of CPU time as reported by Gaonkar et al. [6], but only in the case of single precision computation. As shown by Sinha et al. [7], it does not turn out to be significantly more effective for double precision computations which are the rule with modern computers.

3.1.2 Monodromy matrix computation by matrix exponentials approximation (referred to as the 'exponentials' method)

Following the pioneer work of Hsu [19], Friedmann et al. in [4] and later in [5] developed a method to approximate the monodromy matrix in order to reduce CPU time, with very little loss in accuracy. To do so, $\mathbf{J}(\mathbf{x}_0, t)$ is assumed to be piecewise constant. The time interval $[0, T]$ is divided into K subintervals ($[0, t_1, \ldots, t_K]$). Within each subinterval, $\mathbf{J}(\mathbf{x}_0, t)$ is assumed to be constant and $\mathbf{H}$ can thus be approximated by the following formula:

$$\mathbf{H} \approx \tilde{\mathbf{H}} = \prod_{k=K}^{1} \mathrm{e}^{[\mathbf{J}_k](t_k - t_{k-1})}$$

where $\mathbf{J}_k$ is the mean value of $\mathbf{J}(\mathbf{x}_0, t)$ over the time interval $[t_{k-1}, t_k]$. In practical applications, as $\mathbf{J}(\mathbf{x}_0, t)$ is evaluated only at $t_k \quad k \in [0..K]$, we can write $\mathbf{J}_k = (\mathbf{J}(\mathbf{x}_0, t_k) - \mathbf{J}(\mathbf{x}_0, t_{k-1}))/2$. The quality of the approximation depends on K, since:

$$\lim_{K \to +\infty} \tilde{\mathbf{H}} = \mathbf{H}$$

In practical applications however, the matrix exponentials must be evaluated. Among the many different methods used to compute a matrix exponential, one can cite the Padé approximation, the Taylor approximation, or the calculation based on singular values decomposition. In [20], Cardona et al. give upper bound formulae for the accuracy of the matrix exponential when the Taylor approximation is used.

3.1.3 Monodromy matrix computation by Runge-Kutta single-pass numerical integration (referred to as the 'RK 1-pass' method)

As already mentioned the $2n$-pass integration method can be modified in order to perform only one numerical time in-

tegration instead of $2n$. Friedmann et al. [4] suggested an improved single-pass algorithm based on the fourth order Runge-Kutta integration scheme. In this algorithm $\mathbf{H}$ is computed with the following formula:

$$\mathbf{H} = \prod_{k=1}^{K} \mathbf{G}(T - kh)$$

with $h = T/K$ being the time step size and $\mathbf{G}$ being a matrix (dependant on time) calculated from the matrix $\mathbf{J}$ and other parameters (Gill coefficients). From a programming point of view, this method is very similar to the 'exponentials' methods as the monodromy matrix is evaluated from the product of precomputed matrices. The two methods simply differ by the way theses matrices are calculated.

3.1.4 Monodromy matrix computation by Newmark single-pass numerical integration (referred to as the 'Nm 1-pass' method)

Many time integration schemes can be used to perform the calculation of the monodromy matrix. For instance Gaonkar et al. [6] compare different algorithms (including Runge-Kutta, Hamming and Gear schemes) both in single-pass and multi-pass. In the field of rotordynamics and linear structural dynamics, the implicit Newmark algorithm with constant average acceleration is commonly used due to interesting properties such as unconditional stability. The combination of the nonlinear version of the Newmark algorithm with a shooting method to compute the periodic solution of nonlinear systems can be found in [21,22]. It is shown that the calculation of the monodromy matrix can be advantageously performed as a step of the Jacobian involved in the Newton-Raphson solver. It results in solving a linear system of differential equations in a similar way as described in Sect. 3.1.1 but with a system of size n instead of $2n$ since it is not necessary to put the equation of motion in state form.

3.1.5 Implicit monodromy matrix calculation by k-pass numerical integration

In [23] Bauchau and Nikishkov developed a method that does not require the explicit calculation of the monodromy matrix. Only its k ($k \leq 2n$) largest eigenvalues are approximated thanks to the Arnoldi algorithm. Starting with normalised random intial conditions, k numerical integrations over one period are performed. At the end of each one of the k integrations, a column of $\tilde{\mathbf{H}}$ is determined. $\tilde{\mathbf{H}}$ is a Hessenberg matrix of size k that represents the real monodromy matrix $\mathbf{H}$. The eigenvalues of $\tilde{\mathbf{H}}$ are approximations of the largest eigenvalues of $\mathbf{H}$. If $2n$ integrations over one period are performed then the eigenvalues of $\tilde{\mathbf{H}}$ are exactly the same as the ones of $\mathbf{H}$. In this case this method is equivalent to the $2n$-pass method.

This method is faster that the $2n$-pass method because only k integrations are performed instead of $2n$. However, this method is still slower than a single-pass scheme while introducing an additional level of approximation. Thus, this method shows little interest and for this reason, the results provided by this method will not be given in this paper.

3.1.6 Monodromy matrix calculation with the use of Chebyshev polynomials

Sinha and Wu in [7] introduced an efficient technique based on the idea that the state system can be expanded in terms of Chebyshev polynomials over one period. Thus, instead of performing a numerical integration over one period of a set of linear differential equations, one has to solve a system of linear algebraic equations. This method has been tested on systems with relatively small number of DOFs (up to 10 DOFs in [8]). Even though this method seems promising in terms of CPU performance, it introduces an additional level of approximation as the accury of the result depends on the number of terms retained in the Chebyshev expansion. This method has not been tested in the scope of this paper.

3.1.7 Monodromy matrix calculation by wavelet-Galerkin procedure

In [24] and later in [25], Pernot and Lamarque introduced a wavelet-Galerkin procedure in order to obtain transient and periodic solutions of linear and nonlinear multi-DOFs dynamical systems with time-periodic coefficients. Due to good time-frequency localization properties, the wavelet-based procedure was found to be reliable even in the case of problems involving both smooth or non-smooth parametric excitations and a relatively large number of DOFs. As reported by the authors, benefit is made of the wavelet-Galerkin procedure to compute efficiently the monodromy matrix in one pass and consequently construct stability diagrams characterizing stable/unstable parameters areas and estimators for stability/instability levels. This method however requires an important programming effort. For this reason it has not been applied in this paper.

3.2 Frequency domain methods

3.2.1 Hill's method (referred to as the 'Hill1' method)

The methods described above are based on a time-domain description of the problem. However, as Eq. (5) is solved in the frequency domain, it would seem more natural to compute the stability of the solution in the frequency domain too. In Hill's method, the perturbation is written as the combination of a periodic term and a exponentially decreasing term [3]. This perturbation is introduced into the equation of

motion and a procedure similar to that used by the HBM is applied, leading to a quadratic eigenvalues problem which can be transformed into a linear eigenvalues problem of size $2n(2N+1) \times 2n(2N+1)$.

However, as the perturbation is written as a truncated Fourier series, the accuracy of this method depends on the number of harmonics retained. Although using an infinite number of harmonics would lead to the exact result, it would also lead to a matrix $\hat{\mathbf{H}}$ of infinite dimension. For obvious numerical reasons, only a limited number of harmonics are retained in the Fourier series. This method is known to provide inaccurate results most of the time, especially for larger systems [5,7]. This method is also known to be CPU intensive for large systems. For these reasons, the results provided by this method will not be shown in this paper.

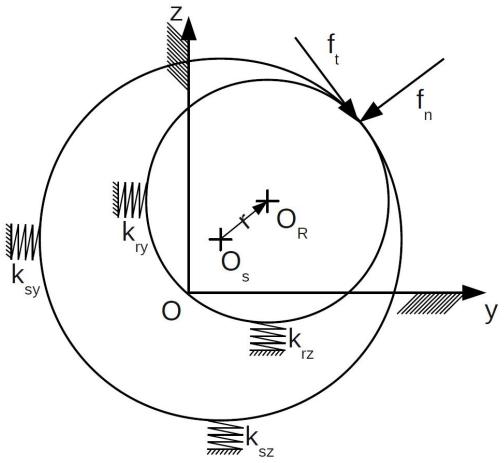

Fig. 1 Jeffcott rotor

3.2.2 Improvement of Hill's method (referred to as the 'Hill2' method)

When using the Hill1 method, one obtains $2n(2N+1)$ Floquet multipliers, of which only $2n$ have a physical meaning. The existence of the remaining non-physical Floquet multipliers can lead to an inappropriate stability evaluation. To address this issue, Lazarus and Thomas [26] proposed a technique to discriminate, and then eliminate the non physical multipliers. Their approach is slightly different from that used in the Hill1 method.

The eigenvalues of the monodromy matrix are filtered on the basis of a criterion related to the associated eigenvectors. The reader may refer to [26] for further details. This approach suffers from the same shortcomings as the Hill1 method (i.e. the choice of harmonics), but is supposed to yield more accurate results as a consequence of the filtering.

4 Test cases

All aforementioned algorithms were developed in the Code_Aster [27] environment, with extensive use of python and the Scipy library [28]. Two different test cases were considered for the analysis of accuracy and performance. It is important to note that these techniques rely on several control parameters, which can have a considerable influence on both accuracy and CPU time.

Contrary to the other techniques, the evaluation of the solution's stability using the 2n-pass technique has no relevant control parameter, since the built-in time integration scheme has several automatic routines to determine well suited parameter values. Indeed, the classical time integration schemes provided with Scipy have a set of default parameters which generally allow accurate integration. In particular, the number of time steps is handled automatically.

The exponentials technique has two main parameters. The most important of these is the chosen number of intervals. Since from numerical experiments a total of 256 intervals was found to produce sufficiently accurate results in every situation, this number was used in all of our tests. The second parameter is the accuracy with which the matrix exponentials are computed. When computing these exponentials using the Padé or Taylor approximation, the built-in Scipy functions use default parameters which are found to provide sufficiently accurate results in every situation.

The Nm 1-pass technique also used a total of 256 time intervals and used the average acceleration scheme with the Newmark parameters $\beta = 0.25$ and $\gamma = 0.5$ (see [33]). For convergence reasons, the RK 1-pass technique used 256 time intervals for the first test case and 1024 intervals for the second test case (except the last variant of the second test case where 2048 intervals were used).

The Hill2 technique used every harmonic between 0 and the highest harmonic present in the HBM response computation.

4.1 Nonlinear Jeffcott rotor

The first test case is inspired by Von Groll and Ewins [3]. It is a modified Jeffcott rotor which can come into contact with the stator (see Fig. 1). Two variants of this model were tested. The first variant features a fixed stator, and has only two DOFs. In the second variant, the stator is suspended and acts like an oscillator interacting with the rotor. This variant has four DOFs. The mountings of the rotor are orthotropic. When the mountings are isotropic, the rotor's orbit remains circular at all times. Thus, only one harmonic is necessary for the HBM calculation and stability analysis. Conversely, when the mountings are orthotropic, the orbits are not circular and contact (when it occurs) is *a priori* non-permanent.

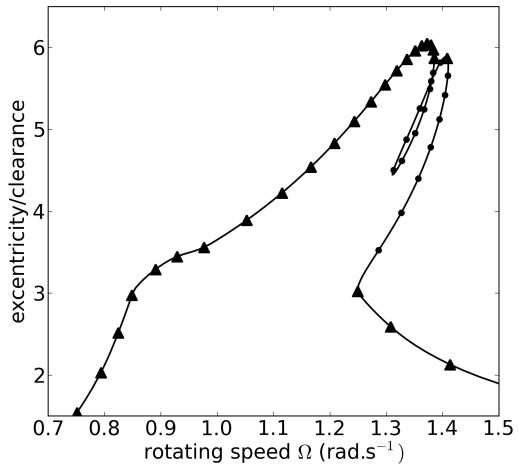

Fig. 2 Jeffcott rotor. Response curve. Stability calculated with the 2n-pass method. (▲) stable, (.) unstable

A greater number of harmonics is thus needed, to compute the HBM solution with good accuracy. In the present example, 24 harmonics were used for the HBM calculation in the first variant and 32 harmonics were used in the second variant. These two variants are summarized in Table 1.
The equations of motion are shown below (for variant 2):

$$\mathbf{m}_r\ddot{\mathbf{q}}_r(t)+\mathbf{c}_r\dot{\mathbf{q}}_r(t)+\mathbf{k}_r\mathbf{q}_r(t)+\mathbf{f}_c(t)=\mathbf{p}(t) \tag{8}$$

$$\mathbf{m}_s\ddot{\mathbf{q}}_s(t)+\mathbf{c}_s\dot{\mathbf{q}}_s(t)+\mathbf{k}_s\mathbf{q}_s(t)-\mathbf{f}_c(t)=\mathbf{0} \tag{9}$$

where $\mathbf{m}_r$ and $\mathbf{m}_s$ represent the mass matrix of the rotor and the stator, respectively. Similarly, $\mathbf{c}_r$ and $\mathbf{c}_s$ are the damping matrices and $\mathbf{k}_r$ and $\mathbf{k}_s$ are the stiffness matrices. $\mathbf{q}_r(t)=[q_{ry}(t),q_{rz}(t)]^T$ is the displacement of the rotor and $\mathbf{q}_s(t)=[q_{sy}(t),q_{sz}(t)]^T$ is the displacement of the stator. $\mathbf{f}_c(t)$ represents the contact force and $\mathbf{p}(t)$ is the external excitation force. In the present example, a penalty contact law coupled with Coulomb friction is used:

$$\mathbf{f}_c(t)=\mathbf{f}_n(t)+\mathbf{f}_t(t)=-k\delta(t)-\mu||k\delta(t)||\frac{\mathbf{v_{rel}}(t)}{||\mathbf{v_{rel}}(t)||} \tag{10}$$

where $\mathbf{f}_n$ and $\mathbf{f}_t$ are respectively the normal and tangential contact forces.

$$\delta(t)=\begin{cases}(r(t)-\delta_0)\mathbf{n_{norm}}(t) & \text{if } r>\delta_0,\\ [0,0]^T & \text{otherwise.}\end{cases} \tag{11}$$

where

$$r(t)=\sqrt{(q_{ry}(t)-q_{sy}(t))^2+(q_{rz}(t)-q_{sz}(t))^2}$$

and

$$\mathbf{n_{norm}}(t)=\left[\frac{(q_{ry}(t)-q_{sy}(t))}{r(t)},\frac{(q_{rz}(t)-q_{sz}(t))}{r(t)}\right]^T$$

k is the contact stiffness and δ_0 is the initial gap between rotor and stator. μ is the Coulomb friction coefficient and

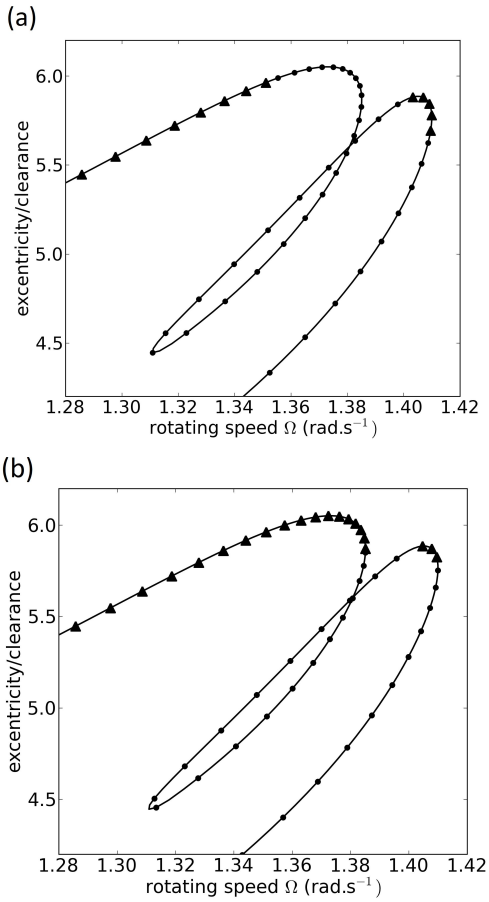

Fig. 3 Jeffcott rotor. Response curve (close up). (a) Stability calculated with the Hill2 method ; (b) Stability calculated with the other methods (2n-pass, exponentials, RK 1-pass, Nm 1-pass). (▲) stable, (.) unstable

$\mathbf{v_{rel}}$ is the relative velocity between the rotor and the stator at the contact point. In this case, the external excitation is generated by an imbalance:

$$\mathbf{p}(t)=p_0\omega^2\left[\cos(\omega t),\sin(\omega t)\right]^T \tag{12}$$

When the first variant is treated, Eq. (9) is ignored and q_s is assumed to be zero in the other equations.

The solutions were calculated for varying values of ω. Figs. 2 and 3 show the results of the simulation for the five different techniques (2n-pass, exponentials, RK 1-pass, Nm 1-pass, Hill2). All the results are summarized in Table 2. The CPU performance is given for the five algorithms, with respect to the time spent without making a stability computation.

The 2n-pass method is very slow, since the computation times is multiplied by 23 compared to the same simulation without stability assessment. The exponentials, RK 1-pass and Nm 1-pass methods remain comparatively fast (between 1.3 and 1.7 times the time needed without stability calculation), while giving accurate results. The Hill2 technique proved to be quite slow compared to the three pre-

rotor model	n_{ele}	n	N	n_{HBM}
Jeffcott v.1	N/A	2	24	98
Jeffcott v.2	N/A	4	32	260
FE rotor v.1	4	24	12	600
FE rotor v.2	6	34	12	850
FE rotor v.3	9	49	12	1225
FE rotor v.4	13	69	12	1725
FE rotor v.5	17	89	12	2225

Table 1 The variants of the rotor models. n_{ele} = number of elements. n = number of DOFs. N = number of harmonics for the HBM calculation. $n_{\text{HBM}} = n(2N+1)$ = dimension of the HBM algebraic system.

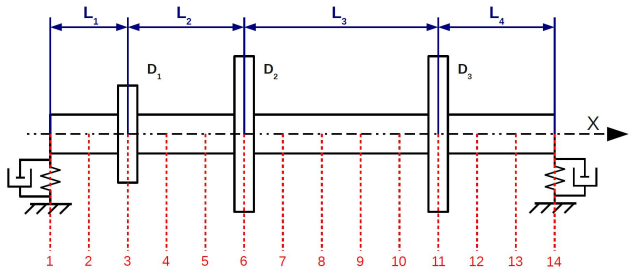

Fig. 4 Multi-DOFs finite element rotor

viously mentioned techniques while yielding inaccurate results. Contrary to a duffing oscillator test case proposed by Lazarus and Thomas (see [26]) the filtering technique is not effective in the present test case.

4.2 Multi-DOF nonlinear rotor

A finite element nonlinear rotor model [29] was also tested (see Fig. 4). This rotor is originaly composed of thirteen Timoshenko beam elements. Two linear orthotropic mountings are located at both ends of the rotor, and three disks (modeled by additional mass and inertia) are added to nodes 3, 6 and 11. The rotor is excited by an imbalance force located on disk 2. The rotor makes contact with a circular, static, and rigid stator located in the vicinity of disk 2. The contact model is identical to that described in section 4.1. The equation of motion (13) is written in a finite element form.

$$\mathbf{M}\ddot{\mathbf{q}}(t)+\mathbf{C}(\omega)\dot{\mathbf{q}}(t)+\mathbf{K}\mathbf{q}(t)+\mathbf{f}_c(q)=\mathbf{p}(\omega,t) \tag{13}$$

where $\mathbf{M}$, $\mathbf{C}(\omega)$ and $\mathbf{K}$ represent the mass, damping and stiffness matrices respectively. $\mathbf{C}(\omega)$ includes the gyroscopic matrix, which varies with ω. In its original form, this model has 69 DOFs. For the purposes of the present study five variants of this model were tested, in four of which the number of elements has been modified (see Table 1) to illustrate the influence of the number of DOFs on the relative performance of the stability computation techniques.

In the first and second variants, the system was reduced to 24 and 34 DOFs respectively. In the third and last variants, the system was modified to 49 DOFs and 89 DOFs, respectively. The number of DOFs was always sufficient to avoid any significant change in the system response over the range of tested rotational speeds. Twelve harmonics were retained for the HBM calculation.

The response curve of the system can be seen in Fig. 5, and the results are summarized in Table 2. The Nm 1-pass method gives accurate results, and is by far the fastest of the five methods. However, it can be noted that the overhead of this method increases with n. The computation time is multiplied by 1.4 in the case of the Jeffcott rotor with $n = 2$ but it is multiplied by 10 in the case of the finite element rotor with $n = 89$. The exponentials method and the RK 1-pass method also become slower when n increases but at a much faster rate than the Nm 1-pass method. The Hill2 method is as slow as the 2n-pass method and yields inaccurate results.

Additional information, together with a discussion of these performance issues, is provided in the following section.

5 Discussion

The results from the two test cases are summarized in Table 2. Although the accuracy of the Hill2 method is unsatisfactory, close analysis of the CPU performance achieved with the five techniques provides considerable insight into these processes. Independently from the accuracy of the results, it can be seen that there are vast differences in terms of CPU requirements, depending on the test case and the algorithm used. The additional cost of the stability computation can be less than 30% of the basic computational time requirement (without the stability calculation) in the case of the RK 1-pass method for the Jeffcott rotor. Conversely, in the case of variant 4 of the FE rotor test case with the Hill2 method, the stability computation multiplies the CPU time requirement by almost sixty thousand. The general trend for the computation times is to increase when n increases (and also when N increases for the Hill's method). In order to analyze this trend, it should be noted that for each of the five methods studied here, the stability analysis can be separated into two main steps:

- constructing a matrix $\mathbf{H}$,
- computing its eigenvalues.

However, the nature of $\mathbf{H}$ and/or the way in which it is constructed is different for each method. When using the Hill's methods, the dimension of $\mathbf{H}$ is $d = 2n(2N_H+1)$, whereas it is $d = 2n$ when using a time domain method (where N_H represents the number of harmonics used in the Hill's method). In the test cases presented here, $N_H = 2N-1$ for the Hill2 method. Depending on the situation, most of the CPU time is spent on either one or the other of these two steps. Tables 3 and 4 indicate the proportion of time spent computing each of the two steps for the Hill2 method and the exponentials

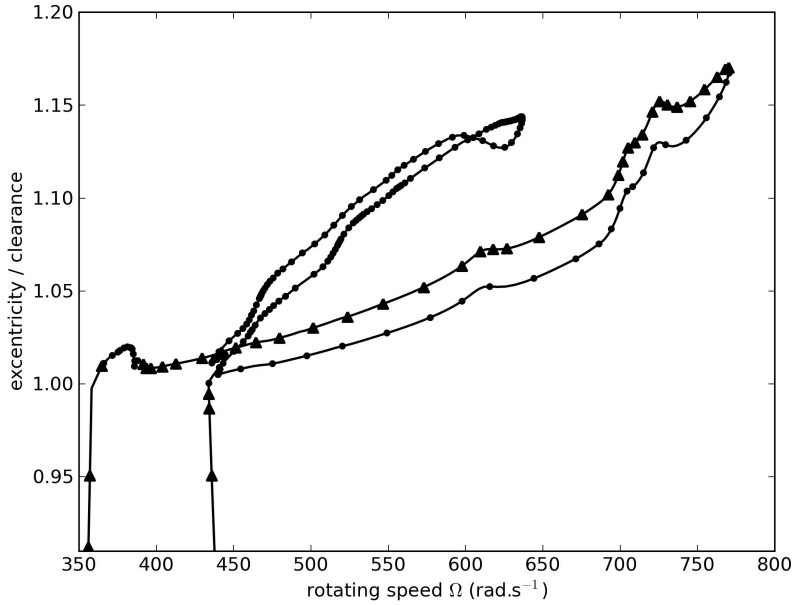

Fig. 5 Multi-DOFs finite element rotor (variant 1). Response curve and stability calculated using the 2n-pass method. (▲) stable, (.) unstable

		Frequency domain	Time domain			
rotor model	No stab.	Hill2	2n-pass	exponentials	RK 1-pass	Nm 1-pass
Jeffcott v.1	1	9.5*	23	1.7	1.3	1.4
Jeffcott v.2	1	151*	45	2.0	1.6	1.5
FE rotor v.1	1	2.4×10^3*	1.4×10^3	10	7.9	1.8
FE rotor v.2	1	4.6×10^3*	5.3×10^3	22	16	2.3
FE rotor v.3	1	1.1×10^4*	1.3×10^4	51	34	3.5
FE rotor v.4	1	5.9×10^4*	3.7×10^4	170	81	5.7
FE rotor v.5	1	N/A	9.9×10^4	260	300	10

Table 2 Relative CPU time. The time spent to compute the response curve without stability determination is normalized to 1 as indicated in the 'No stab.' column. Numbers with an asterisk (*) indicate unsatisfactory accuracy.

method respectively. The data relative to the other time domain methods (2n-pass method, RK 1-pass method and Nm 1-pass method) are not shown here as they are very similar to the exponentials method data.

In the case of the Hill2 method, it should be noted that, except for the Jeffcott test case, most of the CPU time (>99.9%) is spent computing the eigenvalues of **H**. In the first variant of the Jeffcott test case, constructing the matrix **H** represents 17% of the time spent assessing the stability. Classical algorithms for the computation of matrix eigenvalues have an $O(d^3)$ algorithmic complexity, and constructing **H** has an $O(d^2)$ algorithmic complexity. Thus, the portion of time computing the eigenvalues of **H** is is expected to increase as the size of the system increases. The bad overall performance of the Hill2 method can be explained by the fact that the size of the matrix **H** is large ($d = 2n(2N_H + 1)$) thus leading to large eigenvalues computation times.

Contrary to the case of the Hill's methods, most of the time required by the time domain methods is spent computing **H**. This is due to the fact that the dimension of **H** in these methods depends only on n (not N), and is therefore considerably smaller than for the case of the Hill's methods. The eigenvalue computation thus remains relatively fast. Moreover, with these methods the construction of **H** is, by nature, considerably more demanding in terms of CPU than with Hill's methods. However, in absolute terms, it can be noticed that these methods becomes relatively CPU inefficient when n grows. Among the time domain methods, large disparities

Hill2			
rotor model	dimension of **H**	construction of **H**	compute eigenvalues
Jeffcott v.1	380	17%	83%
Jeffcott v.2	1016	2%	98%
FE rotor v.1	2256	0.1%	99.9%
FE rotor v.2	3196	<0.1%	>99.9%
FE rotor v.3	4606	<0.1%	>99.9%
FE rotor v.4	6486	<0.1%	>99.9%
FE rotor v.5	8366	<0.1%	>99.9%

Table 3 Proportion of time spent during stability computations with the Hill2 method.

exponentials			
rotor model	dimension of **H**	construction of **H**	compute eigenvalues
Jeffcott v.1	4	99.8%	0.2%
Jeffcott v.2	8	99.8%	0.2%
FE rotor v.1	48	99.9%	0.1%
FE rotor v.2	68	99.9%	0.1%
FE rotor v.3	98	>99.9%	<0.1%
FE rotor v.4	138	>99.9%	<0.1%
FE rotor v.5	178	>99.9%	<0.1%

Table 4 Proportion of time spent during stability computations with the exponentials method. The 2n-pass, RK 1-pass and Nm 1-pass results are very similar.

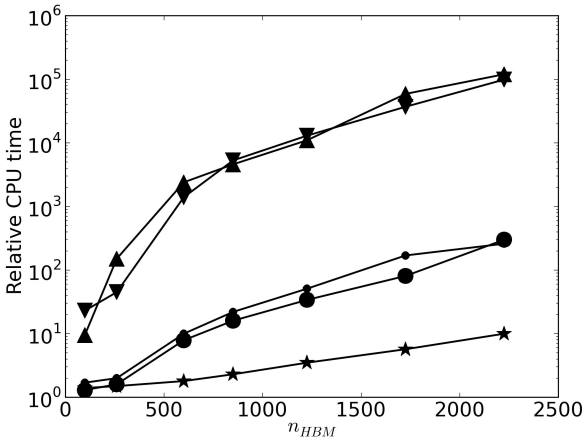

Fig. 6 Relative CPU time as a function of n_{HBM}. Time spent to compute the response curve with stability with respect to the time spent to compute it without stability assessment. ▲ Hill2, ▼ 2n-pass, . exponentials , ● RK 1-pass, ★ Nm 1-pass.

in terms of CPU efficiency can be observed. The 2n-pass method is by far the most inefficient technique as $2n$ numerical integrations over one period have to be performed to compute **H**. This property obviously becomes problematic as n increases. Single-pass methods (exponentials method, RK 1-pass method, Nm 1-pass method) are indeed much faster. However, it can be noted that the Nm 1-pass technique outperforms the exponentials and RK 1-pass techniques. The exponentials and RK 1-pass techniques are similar in their implementation since the matrix **H** is computed as the product of precalculated matrices in both techniques. In the exponentials method, the precalculated matrices are evaluated by a massive use of a matrix exponential calculation function. Matrix exponential computation has an $O(d^3)$ algorithmic complexity, and this operation becomes predominant as n increases. It can thus be concluded that the CPU efficiency of this method can only be expected to decrease, when more complex problems are dealt with.

On one hand, the RK 1-pass technique does not require matrix exponentials calculations but on the other hand, an increasing number of time steps is necessary to obtain convergence. The period T was divided in 256 intervals in the Jeffcott test case. As the Runge-Kutta-Gill method is not inconditionnaly stable, 256 time steps are not enough for the finite element rotor model. 1024 time steps were necessary for the first four variants and 2048 time steps were used for the last variant. The exponentials method required only 256 time steps for all test cases. For the above-mentioned reasons, both the exponentials and the RK 1-pass techniques become very slow when n increases.

Fig. 6 shows the relative CPU time as a function of the total size of the HBM system for the five techniques presented in this paper. The Nm 1-pass technique has proven to be the most efficient one. As there is no need to transform the equation of motion into the state form, the size of the system remains equal to n instead of $2n$. Moreover, only 256 time steps per period were sufficient to obtain good accuracy in all cases, due to the good stability of the implicit Newmark average acceleration scheme for the considered nonlinear problems. However, in absolute numbers, this technique still multiplies the computational time by 10 in the last test case. The algorithmic complexity of this method is still greater than the algorithmic complexity of the HBM method. Thus, this technique is also expected to become increasingly inefficient as the size of the system increases.

6 Conclusion

Five techniques based on the Floquet theory, which are commonly used to compute the stability of periodic solutions, have been compared in terms of accuracy and CPU performance on different test cases. One technique is based on a modified version of the Hill's method in the frequency domain and the four others are based on the computation of the monodromy matrix in the time domain. The five techniques require the computation of a specific matrix **H**, followed by the calculation of its eigenvalues, and differ mainly in the nature of **H** and/or the way in which it is constructed.

It has been shown that the four techniques based on the monodromy matrix calculation yield substantially more accurate results, and contrary to the Hill's methods, most of the CPU time is spent constructing **H**. All factors being considered, for the present nonlinear test cases the monodromy matrix calculation using the single-pass Newmark algorithm appears to provide the best compromise between accuracy and CPU performance. Its accuracy is satisfactory in all of the situations tested in this paper while being by far the fastest of the tested methods.

However, this technique is also expected to become increasingly less efficient as the number of degrees of freedom increases. This inefficiency can nevertheless be avoided with the use of a model reduction technique. For systems with lo-

calized nonlinearities a condensation procedure [30,31] is a good candidate among many others such as component mode synthesis procedure [1] or invariant manifolds construction [32].

Acknowledgements This work was partially supported by the French National Agency (ANR) in the framework of its Technological Research COSINUS program (IRINA, project ANR 09 COSI 008 01 IRINA)

References

1. Sundararajan, P., Noah, S.T.: An algorithm for response and stability of large order non-linear systems – application to rotor systems. J. Sound Vib. **214**(4), 695 – 723 (1998). DOI 10.1006/jsvi.1998.1614
2. Nataraj, C., Nelson, H.D.: Periodic solutions in rotor dynamic systems with nonlinear supports: A general approach. ASME J. Vib. Acoust. **111**(2), 187–193 (1989). DOI 10.1115/1.3269840
3. Von Groll, G., Ewins, D.J.: The harmonic balance method with arc-length continuation in rotor/stator contact problems. J. Sound Vib. **241**(2), 223 – 233 (2001). DOI 10.1006/jsvi.2000.3298
4. Friedmann, P., Hammond, C.E., Woo, T.H.: Efficient numerical treatment of periodic systems with application to stability problems. Int. J. Numer. Methods Eng. **11**(7), 1117–1136 (1977). DOI 10.1002/nme.1620110708
5. Friedmann, P.: Numerical methods for determining the stability and response of periodic systems with applications to helicopter rotor dynamics and aeroelasticity. Comp. and Maths. with Appls. **12**(1, Part A), 131 – 148 (1986). DOI 10.1016/0898-1221(86)90091-X
6. Gaonkar, G.H., Simha Prasad, D.S., Sastry, D.: On computing floquet transition matrices of rotorcraft. J. Am. Helicopter Soc. **26**(3), 56–61 (1981). DOI doi:10.4050/JAHS.26.56
7. Sinha, S., Wu, D.H.: An efficient computational scheme for the analysis of periodic systems. J. Sound Vib. **151**(1), 91 – 117 (1991). DOI 10.1016/0022-460X(91)90654-3
8. Wu, D.H., Sinha, S.: A new approach in the analysis of linear systems with periodic coefficients for applications in rotorcraft dynamics. Aeronaut. J. R. Aeronaut. Soc. **1**, 8–16 (1994)
9. Liaw, C.Y., Koh, C.G.: Dynamic stability and chaos of system with piecewise linear stiffness. J. Eng. Mech. **119**, 1542–1558 (1993)
10. Raghothama, A., Narayanan, S.: Bifurcation and chaos in geared rotor bearing system by incremental harmonic balance method. J. Sound Vib. **226**(3), 469 – 492 (1999). DOI 10.1006/jsvi.1999.2264
11. Zheng, T., Hasebe, N.: An efficient analysis of high-order dynamical system with local nonlinearity. ASME J. Vib. Acoust. **121**(3), 408–416 (1999). DOI 10.1115/1.2893995
12. Kim, Y.B., Noah, S.T.: Quasi-periodic response and stability analysis for a non-linear jeffcott rotor. J. Sound Vib. **190**(2), 239 – 253 (1996). DOI 10.1006/jsvi.1996.0059
13. Shen, J., Lin, K., Chen, S., Sze, K.: Bifurcation and route-to-chaos analyses for mathieu-duffing oscillator by the incremental harmonic balance method. Nonlinear Dyn. **52**(4), 403–414 (2008)
14. Rook, T.: An alternate method to the alternating time-frequency method. Nonlinear Dyn. **27**, 327–339 (2002). DOI 10.1023/A:1015238500024
15. Sinha, S.K.: Dynamic characteristics of a flexible bladed-rotor with coulomb damping due to tip-rub. J. Sound Vib. **273**(4-5), 875 – 919 (2004). DOI 10.1016/S0022-460X(03)00647-3
16. Petrov, E.P., Ewins, D.J.: Analytical formulation of friction interface elements for analysis of nonlinear multi-harmonic vibrations of bladed disks. J. Turbomach. **125**(2), 364–371 (2003). DOI 10.1115/1.1539868
17. Lau, S.L., Cheung, Y.K., Wu, S.Y.: A variable parameter incrementation method for dynamic instability of linear and nonlinear elastic systems. ASME J. Appl. Mech. **49**(4), 849–853 (1982). DOI 10.1115/1.3162626
18. Cameron, T.M., Griffin, J.H.: An alternating frequency/time domain method for calculating the steady-state response of nonlinear dynamic systems. ASME J. Appl. Mech. **56**(1), 149–154 (1989)
19. Hsu, C.S.: Impulsive parametric excitation: Theory. ASME J. Appl. Mech. **39**(2), 551–558 (1972)
20. Cardona, A., Lerusse, A., Gradin, M.: Fast fourier nonlinear vibration analysis. Comput. Mech. **22**(2), 128–142 (1998)
21. Peeters, M., Vigui, R., Srandour, G., Kerschen, G., Golinval, J.C.: Nonlinear normal modes, part ii: Toward a practical computation using numerical continuation techniques. Mech. Syst. Sig. Process. **23**(1), 195 – 216 (2009). DOI DOI: 10.1016/j.ymssp.2008.04.003
22. Ibrahim, S., Patel, B., Nath, Y.: Modified shooting approach to the non-linear periodic forced response of isotropic/composite curved beams. Int. J. Non Linear Mech. **44**(10), 1073 – 1084 (2009). DOI 10.1016/j.ijnonlinmec.2009.08.004
23. Bauchau, O.A., Nikishkov, Y.G.: An implicit transition matrix approach to stability analysis of flexible multi-body systems. Multibody Sys.Dyn. **5**, 279–301 (2001). 10.1023/A:1011488504973
24. Pernot, S., Lamarque, C.H.: A wavelet-galerkin procedure to investigate time-periodic systems: Transient vibration and stability analysis. J. Sound Vib. **245**(5), 845–875 (2001). DOI 10.1006/jsvi.2001.3610
25. Pernot, S., Lamarque, C.H.: A wavelet-balance method to investigate the vibrations of nonlinear dynamical systems. Nonlinear Dyn. **32**, 33–70 (2003). DOI 10.1023/A:1024263917587
26. Lazarus, A., Thomas, O.: A harmonic-based method for computing the stability of periodic solutions of dynamical systems. C.R. Mec. **338**(9), 510 – 517 (2010). DOI 10.1016/j.crme.2010.07.020
27. EDF R&D: Code_Aster: A general code for structural dynamics simulation under gnu gpl licence. (2001–). URL http://www.code-aster.org
28. Jones, E., Oliphant, T., Peterson, P.: Scipy: Open source scientific tools for Python. (2001–). URL http://www.scipy.org/
29. Lalanne, M., Ferraris, G.: Rotordynamics Prediction in Engineering. John Wiley and Sons (1998)
30. Hahn, E.J., Chen, P.Y.P.: Harmonic balance analysis of general squeeze film damped multidegree-of-freedom rotor bearing systems. ASME J. Tribol. **116**(3), 499–507 (1994). DOI 10.1115/1.2928872
31. Sarrouy, E., Sinou, J.J.: Advances in Vibration Analysis Research, chap. Non-Linear Periodic and Quasi-Periodic Vibrations in Mechanical Systems - On the use of the Harmonic Balance Methods, pp. 419–434. InTech (2011). DOI 10.5772/15638
32. Gabale, A.P., Sinha, S.: Model reduction of nonlinear systems with external periodic excitations via construction of invariant manifolds. J. Sound Vib. **330**(11), 2596–2607 (2011). DOI 10.1016/j.jsv.2010.12.013
33. Newmark, N.M.: A Method of Computation for Structural Dynamics. ASCE J. Eng. Mech. Div. **85**(EM3), 2596–2607 (1959).