A Multi-harmonic Balance Method for Periodic Steady-state Nonlinear Structural Dynamics

Tiago Miguel Simões Martins
tiago.s.martins@tecnico.ulisboa.pt

Instituto Superior Técnico, Universidade de Lisboa, Portugal

December 2022

Abstract

Simulations are used in vibration analysis to appraise the structure's functionality and to determine the loading effects, enabling partial optimization before actual prototyping. Oscillations are fundamental in nature, appearing in practical engineering applications, which are generally nonlinear. These problems hardly have analytical solutions, requiring sophisticated techniques to reach approximate solutions. A multi-harmonic balance method with predictor-corrector numerical path continuation is built in Python to analyze spatially discretized structures, modeled after real nonlinear systems. It relies on the alternating frequency time scheme to evaluate the Fourier coefficients of the nonlinear force, by using the fast Fourier transform. The frequency reformulation of the dynamics restates it as a root-finding problem. Furthermore, the predictor-corrector numerical path continuation generates multiple individual solutions in the system's frequency response curve. The first order predictor, with step length adaptation, is used to initialize the subsequent Newton-Raphson corrector step. The latter includes a constraint that makes the problem well-determined. Moreover, forward-accumulation automatic differentiation retrieves the nonlinear force's derivatives, by employing dual numbers. Finally, the tool was integrated in a co-simulation procedure with a commercial software for finite element analysis, enabling new possibilities in the study of large-scale systems.

Keywords: Harmonic balance, nonlinear vibration, numerical continuation, automatic differentiation, Abaqus co-simulation

1. Introduction

Digital model building and simulation has higher versatility, enhanced repeatability and lower long-term cost comparing to real life prototyping and testing. Computational simulations help identify bottlenecks in product requirements and design also providing insight into the most important variables. Hence, one can partially optimize the design before actual prototyping.

Vibration analysis is necessary to predict the natural oscillatory behavior of a structure and its response to periodic external excitation, serving as means to appraise the structure's functionality and determine the loading effects such as dynamic stresses, fatigue and noise [1]. The most common tool for analyzing the dynamics of forced vibrations of structures is the frequency response curve.

Most real life systems are nonlinear, in this way they might present sub-harmonic or super-harmonic oscillations; combined internal resonance [2]; period doubling, quasi periodic or chaotic motions [3]; and multiple qualitatively different solutions to the same dynamic problem [1]. Nonlinearities can be caused by hardening or softening of materials, geometric constraints on deformation, misalignment of substructures, mechanical backlash, dry friction, and many types of nonlinear hysteretic damping, including aerodynamic drag and damping of shock absorbers [4].

As an alternative to time-domain methods, the frequency domain multi-harmonic balance aims to be a versatile and fast alternative for building the frequency response of nonlinear dynamic systems [1, 5, 6]. There is also the need for synergetic workflows regarding newly created tools and the established modeling and simulation software, hence the demand for functional co-simulation procedures.

With this in mind, the main goals of this work are: (1) to build a multi-harmonic balance method with predictor-corrector numerical path continuation; (2) to integrate this tool with an established commercial software for finite element analysis in a co-simulation synergetic workflow; (3) and to investigate and present its potential for the efficient com- putation of the frequency response curves of nonlinear systems, which come from spatial discretization and modeling of real-life engineering applications.

This work starts with a background on periodic state-state structural dynamics, followed by a state of the art review. In the theory section, the dynamics are

restated as the harmonic balance root-finding problem. Next, the implementation of root-solvers and numerical path continuation is analyzed. It is followed by insight into the tool's functionality, in the results section. Then, the co-simulation procedure is developed and tested. Lastly, there is the conclusion.

2. Background

To spatially discretize the dynamics of structures through the Ritz method [7], one firstly defines a finite set of shape functions that must meet the problem's boundary conditions and then considers the generalized degrees of freedom as the magnitude of the shape functions over time. Subsequently, lagrangian mechanics leads to the system's Euler-Lagrange equation,

$$\mathbf{M}\ddot{\mathbf{q}} + \mathbf{C}\dot{\mathbf{q}} + \mathbf{K}\mathbf{q} + \mathbf{f}^{\mathrm{nl}}(\mathbf{q}, \dot{\mathbf{q}}, \omega, t) = \mathbf{f}^{\mathrm{ext}}(\omega, t)\,. \qquad (1)$$

The d-dimensional vector of the generalized degrees of freedom is $\mathbf{q}$, time is t, the baseline excitation circular frequency is ω, the period is T, the mass matrix is $\mathbf{M}$, the damping matrix is $\mathbf{C}$, the stiffness matrix is $\mathbf{K}$, the nonlinear force is $\mathbf{f}^{\mathrm{nl}}$, and the external (non state dependent) force is $\mathbf{f}^{\mathrm{ext}}$. Time derivatives use the dot notation. In vibration analysis, the focus is on the periodic steady-state of the system, the boundary value problem $\mathbf{q}(t) = \mathbf{q}(t+T)$ and $\dot{\mathbf{q}}(t) = \dot{\mathbf{q}}(t+T)$ [8].

Time integration techniques are often used for the study of initial value problems [5, 8]. However, error propagates with forward time integration often originating visible numerical divergence, or excessive decay due to numerical damping [1]. This prevents reasonable periodic steady-state results. Also, this approach does not allow for a convenient implementation of numerical path continuation to generate solutions of the dynamics' equation under parameter variation – for example, to obtain the frequency response curve.

Difference methods transform the derivatives in the equation of motion into finite differences defined on a finite set of time instants and then solve the new set of algebraic equations. These are not the most accurate or fastest time-domain techniques [8].

Shooting methods are time-domain methods that solve the boundary value problem of periodic steady-state by partitioning the time-period and defining an initial value problem on each time-partition [8]. The goal is to find the set of initial states that leads to a matching set of final states. It is a robust technique that can deal with non-smooth forces. Adversely, to use derivative-based root-finding solvers, one requires the sensitivity of the final state with respect to the initial state, i.e. an approximation of the monodromy matrix [1], which is computationally heavy.

Multi-harmonic balance is a variational method specialized for periodic steady-state problems [8]. It operates in the finite dimensional linear space of complex exponentials [1]. It relates to truncated Fourier series, which can handle a large class of signals, even some with jump discontinuities. It simplifies the time series solution into a comparatively small set of unknowns, thus decreasing the time-dimensionality of the problem. Ideally, this results in faster solving times, facilitates numerical path continuation and speeds the computation of the sensitivity of the dynamics with respect to the unknowns. Harmonic balance has been extensively used to compute the frequency response of systems in engineering [5, 6, 9–11].

Classical harmonic balance, operates entirely in the frequency domain. It applies to when the dynamics' dependence on the system variables can be expressed in a closed form - limiting the use cases to polynomial nonlinearities [11, 12]. This will not be used in this work because it hardly generalizes, especially when handling non-smooth contact.

The alternating time-frequency scheme, AFT, evaluates the nonlinear force in the time domain and relies on the discrete Fourier transform [1, 6, 10]. It is precise and versatile, being a popular choice [12]. It can handle non-smooth nonlinearities such as stick-slip or unilateral contact. Therefore, it is the most adequate choice for this work.

Stemming from high order Taylor expansions of the dynamics at each solution point, the asymptotic numerical method expresses the solution curve as piecewise multivariate polynomial on an arc length parameter [1, 11]. It aims to provide a smoother and, ideally, more faithful prediction of the solution path which allows for larger and automatically chosen predictor step lengths, reducing the total amount of iterative corrections. It only applies if the frequency domain dynamics are Taylor expandable with polynomial order larger than one, and the computation of the Taylor coefficients is a complicated procedure that takes a significant computational toll [12]. It might not be applicable for many engineering applications, particularly those with stick-slip or unilateral contact.

A simpler alternative is the predictor-corrector numerical continuation [1, 6]. The new solution is predicted from the last found solution point(s) by use of a first-order approximation of the solution curve. Then, an iterative root-solver finds the next solution point. When step length is excessive, important solution curve features might be poorly resolved or omitted entirely. Hence, this procedure requires a step-length adaptation algorithm. It does not demand the computation or existence of high order derivatives, which is more versatile and useful when handling non-smooth dynamics [12].

To avoid convergence failure and to reach the same accuracy as the predictor-corrector numerical continuation, the asymptotic numerical method needs a comparatively large harmonic truncation order. The same occurs when comparing the classical harmonic balance to the AFT [12]. For non-smooth nonlinearities (e.g. in contact problems) the AFT predictor-corrector com-

bination is the best suited option and is the one used in this work.

3. Theory

3.1. Fourier Analysis

A Fourier series is a T-periodic signal that can be represented by the orthonormal (countably infinite) basis of T-periodic complex exponentials. The type of signals that will be dealt with are at least piecewise continuous and with piecewise continuous first-order derivative with respect to time [13]. Emphasizing that jump discontinuities are of high engineering relevance; especially in this work, as they occur in some contact mechanical problems, e.g. stick-slip friction or unilateral contact. The Dirichlet's theorem and Carleson-Hunt theorem are enough to safeguard the convergence of the Fourier series of the signals at hand.

A truncated Fourier series only considers frequency multiples (the harmonics) up to a given magnitude. A generic truncated Fourier series $\mathbf{y}$ of truncation order H is

$$\mathbf{y}(t) = \sum_{n=-H}^{H} \mathbf{Y}_n \exp(jn\omega t)\,,$$

where j is the imaginary unit and $\mathbf{Y}_n$ is the nth Fourier coefficient, which can be retrived with the Fourier integral, as such:

$$\mathbf{Y}_n = \mathcal{F}\{\mathbf{y}\}_n = \frac{1}{T}\int_0^T \mathbf{y}(t)\, \exp(-jn\omega t)\, \mathrm{d}t\,.$$

For functions with a jump discontinuity, their truncated Fourier series will always have a non-zero maximum error near the jump, independently on the harmonic truncation order - the Gibbs phenomenon [1]. It does not affect the "position" and "velocity" as they are forcefully continuous in time. However, some nonlinear contact forces might be modeled with jump discontinuities and their truncated Fourier series will present with this undesired phenomenon. The Fourier coefficients of the periodic signals in the differentiability class $\mathcal{C}^p$ decrease algebraically with index $p+1$, for $p \geq 0$ [1].

3.2. Dynamics in the frequency domain

When simplifying the system such that both the response, $\mathbf{q}(t)$, and the forces are truncated Fourier series, one must compensate the dynamics by introducing the residual force, $\mathbf{r}(t)$. By definition, the residue balances the dynamics of the reduced system, as it gathers all the terms not covered by the ansatzes [5]. Therefore, the Fourier coefficients of $\mathbf{r}$ must be null for all harmonics n such that $|n| \leq H$. Defining $\mathcal{F}\{\mathbf{q}\}_n = \mathbf{Q}_n$, $\mathcal{F}\{\mathbf{f}^{\mathrm{nl}}\}_n = \mathbf{F}^{\mathrm{nl}}_n$, $\mathcal{F}\{\mathbf{f}^{\mathrm{ext}}\}_n = \mathbf{F}^{\mathrm{ext}}_n$ and $\mathcal{F}\{\mathbf{r}\}_n = \mathbf{R}_n$ and substituting the ansatzes in the dynamic equation leads to

$$\sum_{n=-H}^{H} \left[\mathbf{A}_n\mathbf{Q}_n + \mathbf{F}^{\mathrm{nl}}_n - \mathbf{F}^{\mathrm{ext}}_n\right] \exp(jn\omega t) = \mathbf{r}(t)\,,$$

where $\mathbf{A}_n = -(n\omega)^2\mathbf{M} + jn\omega\mathbf{C} + \mathbf{K}$. The uniqueness of the Fourier series implies a correspondence between the residue and the forces in the frequency domain. For each harmonic $-H \geq n \geq H$,

$$\mathbf{A}_n\mathbf{Q}_n + \mathbf{F}^{\mathrm{nl}}_n - \mathbf{F}^{\mathrm{ext}}_n = \mathbf{R}_n = \mathbf{0}\,. \tag{2}$$

3.3. Time-discretization

To characterize the system's time-dependent signals, only a finite set of N evenly spaced samples will be considered, allowing for a computationally viable implementation of the Fourier transform. The notation for sampling is $\mathbf{y}_k = \mathbf{y}(t = t_k)$ and $t_k = Tk/N$ where $k \in \{0, \ldots, N-1\}$.

Let $E_{n,k} = \exp(2\pi jnk/N)/N$. The discrete Fourier transform, DFT serves as the discrete-time counterpart to the continuous-time Fourier integral. For $0 \geq n < N$, the nth component of the DFT is

$$\mathbf{Y}_n = \mathrm{DFT}\{(\mathbf{y}_k)_{k=0}^{N-1}\}_n = N\sum_{k=0}^{N-1} \mathbf{y}_k\, E_{-n,k}\,.$$

In the same way time-periodicity occurs, expressly $\mathbf{y}_k = \mathbf{y}_{k+N}$, due to the cyclic nature of the complex exponential, the Fourier coefficients exist in a cycle of length N, namely $\mathbf{Y}_n = \mathbf{Y}_{n+N}$. This issue might cause aliasing [12], which is when two distinct frequency components become indistinguishable under sampling. This leads to the imprecise representation of data and unreliable signal reconstruction. To avoid aliasing in real signals, one must follow the Nyquist criterion, namely to sample at a rate larger than double the highest relevant frequency expected [1]. Hence, if one expects, for the continuous time case, an adequate truncation order of $\widetilde{H}$, then, when using the discrete Fourier transform, the sampling should follow $N \geq 2\widetilde{H} + 1$.

For completeness, the inverse discrete Fourier transform, iDFT, follows

$$\mathbf{y}_k = \mathrm{iDFT}\{(\mathbf{Y}_n)_{n=0}^{N-1}\}_k = \sum_{n=-H}^{H} \mathbf{Y}_n\, E_{n,k}\,.$$

3.4. Root finding problem

Let $\mathbf{Q}$ and $\mathbf{R}$ be Nd-dimensional vectors that group all coefficients $\mathbf{Q}_n$ and $\mathbf{R}_n$, respectively. The goal is to solve $\mathbf{R}(\mathbf{X}) = \mathbf{0}$, where $\mathbf{X} = [\mathbf{Q}^{\mathrm{T}}\,,\, \omega]^{\mathrm{T}}$. This equation has an uncountable set of solutions, since it is solvable for some frequency range. Said solution set generally lacks an explicit analytical description due to the nonlinear term on the dynamical balance [12]. To solve this issue, predictor-corrector numerical path continuation aims to discretize the solution curve by consecutively and individually obtaining separate solution points [1, 14]. In the prediction step, a first-order approximation of the solution curve, either by a tangent (or secant), is used to extrapolate a predicted solution from the last found solution point(s). In the correction step, an iterative root-finding method used the predicted solution as initialization to find a new solution point.

The mismatch between the dimensionality of $\mathbf{X}$ and $\mathbf{R}$ makes the problem underdetermined, i.e. without

a unique solution. Thus, individually obtaining solution points requires the annexation of an independent equation to be solved [1]. To guarantee that the solution is in a sensible region, one might add a constraint to the search space. For example, to translate the requirement of proximity to a initial guess for the solution or a prerequisite of following given directions in the search space, for instance to avoid backtracking or branch switching. Note that, adding multiple independent equations will make the problem overdetermined, undermining the existence of solutions. Without loss of generality, let this constraint be given by the zero-level set of a scalar function g. The joint problem of the dynamic equilibrium - equation (2) - and the g constraint can be represented as

$$\mathbf{f}(\mathbf{X}) = [\mathbf{R}^{\mathrm{T}}(\mathbf{X})\,,\; g(\mathbf{X})]^{\mathrm{T}} = \mathbf{0}\,. \tag{3}$$

The dimensionalities of $\mathbf{X}$ and $\mathbf{f}(\mathbf{X})$ are the same.

4. Implementation

4.1. Root-finding solver

A root-finding algorithm aims to find zeros of continuous functions. When there is no closed form expression to compute the zeros of a function, root-finding algorithms approximate them by providing a numerical estimate. The majority of these algorithms aspire to generate a sequence of solution candidates that converges to one root, being called iterative methods [8, 15].

The fixed-point methods are easily generalizable to higher dimensions and have fast convergence rates [8]. These methods originate from the restatement of root-finding as a fixed-point problem [8]. In structural dynamical problems, the most commonly used root-finding algorithms are those of Newton-type [15]. Application examples are 1. the implicit solver from Exudyn, a C++ based Python library for efficient simulation of flexible multibody dynamics systems, 2. Abaqus/Standard, a general-purpose implicit time-integration finite-element analyzer, and 3. the implementations of harmonic balance in general literature, e.g [1, 6, 10]. Serving as a simple proof of concept, the simple Newton-Raphson was chosen for the corrector step. It is versatile, presents Q-quadratic convergence and has general applicability. Yet, it is simple enough to be later adapted using more specialized Newton-type algorithms.

4.2. Predictor step

Regarding the construction of the discretized solution curve, to avoid random guessing the initialization for each new iterative solver step, one can make an educated approximation, a prediction. The predictor step methods discussed in this work always use the information from the last solution, $\mathbf{X}^{\mathrm{last}}$, to help extrapolate the position of the next solution, $\mathbf{X}^{\mathrm{next}}$, by indicating a predicted solution, $\widetilde{\mathbf{X}}$ as

$$\widetilde{\mathbf{X}} = \mathbf{X}^{\mathrm{last}} + \mathbf{V}\Delta s\,. \tag{4}$$

where Δs is a positive real scalar and $\mathbf{V}$ is a unit vector [1]. To avoid backtracking, $\mathbf{V}$ must be aligned with a reference direction, $\mathbf{V}_{\mathrm{ref}}$, satisfying $\Re(\,\overline{\mathbf{V}}^{\mathrm{T}}_{\mathrm{ref}}\mathbf{V}\,) \geq 0$. The reference direction should approximate the direction of the solution path, for instance it can be the previous predicted direction - $\mathbf{V}_{\mathrm{ref}} = \mathbf{V}^{\mathrm{prev}}$ - or, more accurately, it can be aligned with the secant through the last two solution points - $\mathbf{V}_{\mathrm{ref}} = \mathbf{X}^{\mathrm{last}} - \mathbf{X}^{\mathrm{prev}}$. Also, one must guarantee a real step in frequency, thus $\mathbf{V}_{\omega} \in \mathbb{R}$.

The simplest approach is to recycle the last solution as an initialization for the next corrector step. This type of predictor step undermines the continuation procedure when dealing with sections of constant frequency or saddle-node bifurcations regarding the parameter ω (the turning-points).

Another predictor method is the secant predictor, giving $\mathbf{V} = (\mathbf{X}^{\mathrm{last}} - \mathbf{X}^{\mathrm{prev}})/\|\mathbf{X}^{\mathrm{last}} - \mathbf{X}^{\mathrm{prev}}\|$. Noticeably, it only works after obtaining two solution points. It is a good approximation of the tangent when the path is almost rectilinear, but fails in the presence of significant curvatures.

Finally, there is the tangent predictor, for which it is necessary to determine the direction for which the local change in the residue is null so that the residue remains equal to zero. One must find the non trivial solution to

$$\left.\frac{\mathrm{d}\mathbf{R}}{\mathrm{d}\mathbf{X}}\right|_{\mathbf{X}^{\mathrm{last}}} \mathbf{V} = \mathbf{0}\,,$$

where the derivative is already known from the previous Newton-type corrector step. This is the most accurate first-order predictor method and can handle turning points and sections of constant ω.

In this work, all three predictor step method alternatives were implemented, but the tangent predictor prevails in most applications.

The step length should be as large as possible to minimize the number of dynamic problems to be solved at the corrector steps [14]. Conversely, the step length must be bounded above by the necessity to ensure fast convergence of the corrector step, maintain solution branch and represent with sufficient detail the global solution space. In short, Δs must be bounded above and below, respectively, by the maximum and minimum step lengths adequate for the problem [1]. Both bounds should come from expert knowledge of the specific problem and are influenced by the desired accuracy and speed of the numerical continuation procedure.

The program should be able to adjust the step length to the needs of the solver and this is called step length adaptation. An approach is to change the step length for the next predictor step according to the quality of the last prediction-correction stage, namely the number of iterations needed until convergence, $N^{\mathrm{last}}_{\mathrm{iter}}$. Accordingly, the step length should decrease, if the number of iterations is "large" and the other-way around

if it is "small". Both terms exist in comparison with the goal number of iterations specified by the user, $N_{\text{iter}}^{\text{goal}}$, which in turn comes from expert knowledge of the specific problem, either by self experience or external sources. In general one has,

$$\Delta s^{\text{next}} = \max\{\Delta s^{\min}, \min\{\Delta s^{\max}, \mathfrak{k}\Delta s^{\,\text{last}}\}\}\,.$$

where $\mathfrak{k}$ is computed based on the $N_{\text{iter}}^{\text{last}}$ and $N_{\text{iter}}^{\text{goal}}$. The option suggested in [1] is to double (of half) the step length if $N_{\text{iter}}^{\text{last}}$ subceeds (or exceeds) the goal. The option used in this work is

$$\mathfrak{k} = \left(\frac{N_{\text{iter}}^{\text{goal}} + 1}{N_{\text{iter}}^{\text{last}} + 1}\right)^2 ,$$

which grades the corrector step's performance in a continuous manner, distinguishing "large" and "small" contrasts against the goal. Squaring the ratio serves to intensify the penalty for "large" numbers of iterations since the goal is probably small. Also, the +1 term is to avoid dividing by zero.

4.3. Parameterization of the corrector step

The purpose of parameterizing the search space is to avoid backtracking or having large jumps in the solution. In short, it provides a sensible region for the next solution by establishing a constraint. The constraint can be expressed as the zero-level set of a function g. For the implementation with a Newton-type solver, the relevant directives are

$$\text{compute } g(\mathbf{X}) \text{ and } \frac{\mathrm{d}g}{\mathrm{d}\mathbf{X}} = \left[\frac{\mathrm{d}g}{\mathrm{d}\mathbf{Q}}, \frac{\mathrm{d}g}{\mathrm{d}\omega}\right] .$$

There are multiple types of search space parameterization, nevertheless, taking into account ease of implementation, applicability and computational optimality, only four methods are addressed [1].

1. Local parameterization is a linear constraint that fixes the frequency. It only allows for adjustments in the Fourier coefficients, $\mathbf{Q}$, and hence it can not deal with turning points or sections of constant frequency in the solution curve.

2. Orthogonal parameterization defines a search space orthogonal to the predicted direction. It is a linear constraint defined by $g(\mathbf{X}) = \gamma\overline{\mathbf{V}}^{\mathrm{T}}(\mathbf{X} - \widetilde{\mathbf{X}})$. Here, the complex scalar γ serves to "harden" or "soften" the constraint. The quality of the orthogonal parameterization depends on the type of predictor method. In combination with a robust predictor step, one can deal with turning points and sections of constant frequency.

3. Arc length parameterization establishes the search space as a hyper-sphere centered at the last solution. The constraint is expressed as $g(\mathbf{X})/\gamma = \|\mathbf{X} - \mathbf{X}^{\text{last}}\|^2 - (\Delta s)^2$, which is guaranteed to intersect another solution. It is a nonlinear (quadratic) constraint that demands the computation of the parameterization's gradient at every iteration. It allows for back-tracking.

4. Normal parameterization avoids the search for the solution in directions that do not change the residue. The search is redefined at each iteration by considering the space orthogonal to the local tangent of the residue. This is heavily nonlinear and might undermine the solver's convergence.

Since they balance functionality and computational efficiency, only the orthogonal and the arc length parameterizations were implemented.

4.4. Alternating frequency-time scheme

The linear and external forces can be directly evaluated in the frequency domain, however the Fourier coefficients of the nonlinear force can not be explicitly expressed in terms of $\mathbf{X}$ [12]. In the alternating frequency-time scheme, at each sampling instant, the system's state is computed with the iDFT, serving as argument for the discrete time-evaluation of the nonlinear force and, finally, through the DFT, all the nonlinear force's time samples are used to compute the corresponding Fourier coefficients [10]. In schematic form, this is

$$\mathbf{Q} \xrightarrow{\text{iDFT}} (\mathbf{q}_k)\,,\ (\dot{\mathbf{q}}_k) \xrightarrow{\text{function eval.}} (\mathbf{f}_k^{\text{nl}}) \xrightarrow{\text{DFT}} \mathbf{F}^{\text{nl}}\,.$$

To compute the DFT and the iDFT, one can implement the fast Fourier transform allowing for more computational efficiency. The AFT can handle non-smooth forces, namely those in contact problems.

4.5. Derivatives of the residue

The derivatives of $\mathbf{R}$ with respect to $\mathbf{Q}$ follow

$$\frac{\mathrm{d}\mathbf{R}_n}{\mathrm{d}\mathbf{Q}_{n'}} = \begin{cases} \mathbf{A}_n + \dfrac{\mathrm{d}\mathbf{F}_n^{\text{nl}}}{\mathrm{d}\mathbf{Q}_n}\,, & n = n' \\ \dfrac{\mathrm{d}\mathbf{F}_n^{\text{nl}}}{\mathrm{d}\mathbf{Q}_{n'}}\,, & n \neq n' \end{cases} ,$$

where, due to the fact that $\mathbf{F}^{\text{nl}}$ is obtained through the DFT , one has

$$\frac{\mathrm{d}\mathbf{F}_n^{\text{nl}}}{\mathrm{d}\mathbf{Q}_{n'}} = \text{DFT}\left\{\left(\frac{\mathrm{d}\mathbf{f}_k^{\text{nl}}}{\mathrm{d}\mathbf{Q}_{n'}}\right)_{k=0}^{N-1}\right\}_n .$$

And the derivative of the nonlinear force in the time domain is obtained by the chain rule,

$$\frac{\mathrm{d}\mathbf{f}_k^{\text{nl}}}{\mathrm{d}\mathbf{Q}_n} = \left[\frac{\partial\mathbf{f}_k^{\text{nl}}}{\partial\mathbf{q}_k} + \frac{\partial\mathbf{f}_k^{\text{nl}}}{\partial\dot{\mathbf{q}}_k} jn\omega\right] E_{n,k}\,.$$

Also, one must consider that the samples of the "position" and "velocity" are given by the inverse discrete Fourier transform, of the coefficients $(\mathbf{Q}_n)$ and $(jn\omega\mathbf{Q}_n)$, respectively.

$$\frac{\mathrm{d}\mathbf{R}_n}{\mathrm{d}\omega} = [jn\mathbf{C} - 2n^2\omega\mathbf{M}]\mathbf{Q}_n + \frac{\mathrm{d}\mathbf{F}_n^{\text{nl}}}{\mathrm{d}\omega} - \frac{\mathrm{d}\mathbf{F}_n^{\text{ext}}}{\mathrm{d}\omega}.$$

Using he chain rule, the derivative of the nonlinear force with respect to the frequency domain to the circular frequency is

$$\frac{\mathrm{d}\mathbf{F}_n^{\mathrm{nl}}}{\mathrm{d}\omega} = \mathrm{DFT}\left\{\frac{\partial \mathbf{f}_k^{\mathrm{nl}}}{\partial \omega} + \frac{\partial \mathbf{f}_k^{\mathrm{nl}}}{\partial \dot{\mathbf{q}}_k}\,\mathrm{iDFT}\{(\mathbf{Q}_n\, jn)\}_k\right\}_n .$$

The derivative of the external force can also be expressed with the DFT. The partial derivatives of the force samples with respect to ω are null in most applications because the frequency commonly only appears combined with time, as ωt. The sampling is independent of ω since $\omega t_k = 2\pi k/N$.

4.6. Automatic differentiation

Automatic differentiation is a collection of techniques to compute the derivative of a function detailed by a computer program, such that the final derivative gets outputted along with the regular output of the function. It manages a large set of functions without requiring specific case tailoring, allowing for user provided functions, and delivers accurate derivatives up to machine precision [15].

Forward accumulation automatic differentiation can be partially accomplished by extending the algebra of complex numbers via the combination with an additional component representing the differential of the given quantity. This extended algebra (known as dual numbers) describes the arithmetic of quantities x with an infinitesimal part $\mathrm{d}x$, such that $(\mathrm{d}x)^2 = 0$.

The generalization to d dimensions follows from the fact that the elements of $\mathbb{C}^d$ are of the form $\mathbf{x} = [x_1, \ldots, x_d]^{\mathrm{T}}$, and hence there exists d different differentials - one for each dimension: $\mathrm{d}x_1, \ldots, \mathrm{d}x_d$. The generalization defines $\forall p, p' : \mathrm{d}x_p\, \mathrm{d}x_{p'} = 0$. In this extended algebra, each component of a vector is of the form $\alpha_0 + \alpha_1\, \mathrm{d}x_1 + \cdots + \alpha_d\, \mathrm{d}x_d$, and thus a general vector is described by $d \times (d+1)$ complex scalars. Any vector element can be mapped through the trivial injection $\mathbf{x} \mapsto \mathbf{x} + \mathrm{d}\mathbf{x}$ where $\mathrm{d}\mathbf{x} = [\mathrm{d}x_1, \ldots, \mathrm{d}x_d]^{\mathrm{T}}$.

To generalize the idea of polynomial to the multivariate case, define the monomial $\mathbf{x}^{\boldsymbol{\alpha}}$ as a product of the variables raised to natural exponents, $x_1^{\alpha_1} \cdots x_d^{\alpha_d}$ with $\boldsymbol{\alpha} \in \mathbb{N}_0^d$. Then consider the multivariate polynomial $\mathbf{f} : \mathbb{C}^d \longrightarrow \mathbb{C}^{d'}, \quad \mathbf{x} \mapsto \mathbf{f}(\mathbf{x}) = \sum_{\boldsymbol{\alpha}} \mathbf{b}_{\boldsymbol{\alpha}} \mathbf{x}^{\boldsymbol{\alpha}}$, with $\mathbf{b}_{\boldsymbol{\alpha}} \in \mathbb{C}^{d'}$ for all $\boldsymbol{\alpha}$. Evaluating the polynomial using the extended algebra leads to

$$\mathbf{f}(\mathbf{x} + \mathrm{d}\mathbf{x}) = \mathbf{f}(\mathbf{x}) + \left.\frac{\partial \mathbf{f}}{\partial x_1}\right|_{\mathbf{x}} \mathrm{d}x_1 + \cdots + \left.\frac{\partial \mathbf{f}}{\partial x_d}\right|_{\mathbf{x}} \mathrm{d}x_d .$$

To handle relevant elementary functions one can implement additional rules, for instance, regarding the natural logarithm, $\ln(x + \mathrm{d}x) = \ln(x) + \mathrm{d}x/x$. Other examples would be exponentials, sinusoids and inverse trigonometric functions.

Unlike symbolic or numerical differentiation, automatic differentiation is efficient at computing partial derivatives of multivariate functions. It computes exact derivatives through elementary binary operations and functions, in a simultaneous and integrated manner [15]. Finally, it can deal with a large range of functions/programs, managing jump discontinuities by giving the unilateral derivative, which is useful for contact problems.

5. Results

This section discusses the success of the method developed in this work, MHBM, by comparing it to preexisting tools, both in the frequency and time domains. The solver used in this section adopts the Newton-Raphson corrector, the tangent predictor and the orthogonal parameterization. The solution is not given as an expression of the curve $\mathbf{R} = \mathbf{0}$, but instead as a finite set of approximate solutions that lay near it.

Note that, indicating a system of measurement is irrelevant when attributing numerical values to the geometry or material parameters, instead one only requires coherence between units.

5.1. Duffing oscillator

The Duffing oscillator [1] is a one-dimensional second order linear dynamic system characterized by the addition of a cubic spring (the nonlinear term) and excited by a sinusoidal external force. Namely the system follows $M\ddot{q} + C\dot{q} + Kq + \beta q^3 = P\cos(\omega t)$.

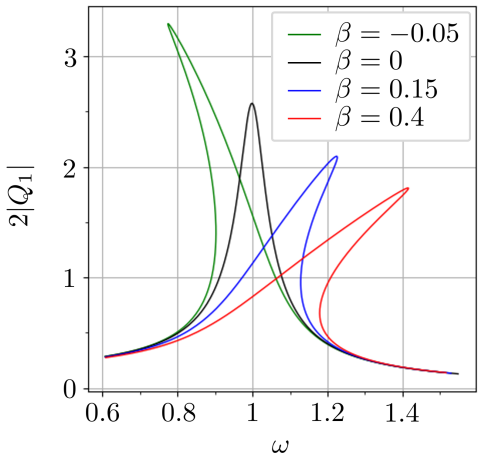

Figure 1: Frequency response of Duffing oscillator of the form, $\ddot{q} + 0.07\dot{q} + q + \beta q^3 = 0.18\cos(\omega t)$, under variation of β.

The results of the MHBM were verified by NLvib [1], another multi-harmonic balance tool. This was done regarding multiple configurations of the Duffing oscillator by varying the numerical values of its parameters. In figures 1 and 2, are some of the results.

One can observe that the nonlinear systems might show an overhang delimited by two turning points. This region is characterized by a chaotic behavior of the real system in the time domain due to the existence of an unstable equilibrium.

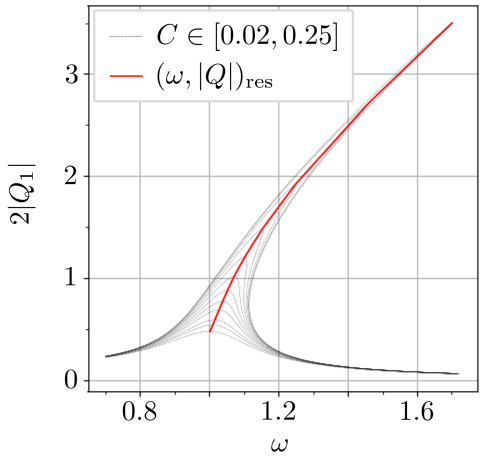

Figure 2: Construction of the backbone curve from of Duffing oscillators of the form, $\ddot{q} + C\dot{q} + q + 0.2q^3 = 0.12\cos(\omega t)$, under variation of C. The backbone curve was traced by joining the resonance peaks of multiple frequency response curves.

By zooming in, it was possible to observe step length adaptation: the steps shortened in the high curvature peak and lengthened as the solution curve straightened. Hence, the results verify the effectiveness and usefulness of predictor-corrector numerical path continuation for rendering the necessary curve features, such as the sharp turning points.

To make a time-domain analysis, one must consider a Duffing oscillator with numerical values attributed to all parameters. In the time domain, the explicit Runge-Kutta method of order four [8], RK45, was used as reference by initializing it with the MHBM's solution at $t = 0$, i.e the state $(q_0, \dot{q}_0)$. The relative error of the time series was below 10^{-4} for all solutions in the frequency range. The mean and maximum relative errors (in the time series) fall below 3% and 5%, respectively, for all $H \geq 3$ and decrease (at least) exponentially with increasing truncation order. A truncation order of $H = 9$ already guarantees a mean relative error in the time domain below 0.9%. Regarding each of the response's Fourier coefficients, one can verify that, independently of the harmonic, the relative error decays exponentially with increasing truncation order and with the same order and rate of convergence.

5.2. 2-DoF system with two cubic springs

Here, MHBM was used to generate the frequency response of the system in figure 3. The result is in figure 4, where one can observe four apparent "resonance" peaks. To verify the solution, the seven solutions marked with crosses were compared in the time domain to the RK45. The order of magnitude of the relative error reached a maximum of 10^{-3} but it was usually around 10^{-5}, and often reached lows of 10^{-6} and 10^{-7}. Also, the overall aspect of the frequency response curve is akin to the one obtained in [16] regarding a similar system.

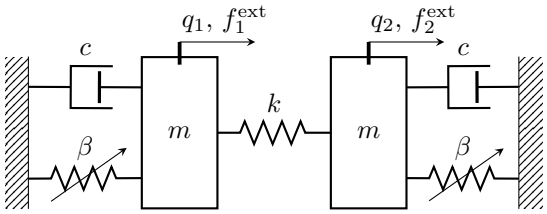

Figure 3: Two mass system with an interconnecting linear spring and grounded dampers and cubic springs.

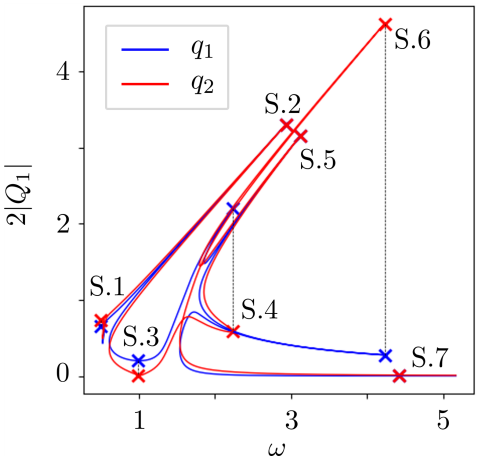

Figure 4: Frequency response of the system. Here $m = 1$, $c = 0.01$, $k = 1$, $\beta = 1$, $f_1^{\text{ext}} = 0$ and $f_2^{\text{ext}} = 0.2\cos(\omega t)$. The MHBM used a truncation order of $H = 9$.

5.3. Euler-Bernoulli beam with dry friction

In this subsection, the case study is the system in figure 5. Through Ritz method [7], and by using (third degree) polynomial shape functions to describe the bending deformation, one obtains the inertia and stiffness matrices.

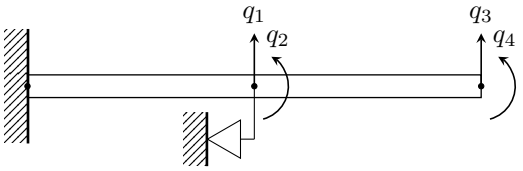

Figure 5: Two element discretization of a linearly elastic cantilever beam with a dry friction element attached at the middle node, acting in the transverse direction.

The dry friction element describes the nonlinear contact between two solid surfaces [4], and it can be treated, in its simplest form, using the Coulomb friction model, as seen in [9, 10]. The Coulomb friction force is given by $f_f \text{sign}(\dot{q}_t)$, where f_f is the friction magnitude, $\dot{q}_t$ is the tangential relative velocity at the

contact and sign is the signum function. This formulation of dry friction is discontinuous at $\dot{q}_\mathrm{t} = 0$, which heavily undermines the success of root-finding algorithms. A smooth regularization can solve this issue, for instance, by approximating the Coulomb friction by a hyperbolic tangent. This regularization solution is well acknowledged [1], is quickly implementable, and facilitates the convergence of Newton-type solvers by introducing domain-wise differentiability. Hence the dry friction becomes $f^{\mathrm{D.F.}} = f_f \tanh(\alpha \dot{q}_t)$, where the value the derivative of the friction force with respect to the tangential velocity at the static stick-condition is α. When $\alpha \to \infty$, this regularization tends to the original Coulomb friction.

Assembling the dry friction elements gives $\dot{q}_\mathrm{t} = \dot{q}_1$ and $\mathbf{f}^{\mathrm{nl}} = [\, f^{\mathrm{D.F.}},\, 0,\, 0,\, 0\,]^{\mathrm{T}}$. At this point numerical values will be given to the beam parameters: the length is 1, the height is 0.1, the thickness is 0.3, the Young's modulus is $185 \cdot 10^9$ and the density is 7830. Also $\alpha = 2 \cdot 10^6$ and $f_f = 1.5$.

The chosen external force is given by the expression $\mathbf{f}^{\mathrm{ext}} = [\, 0,\, 0,\, 0.2\cos(\omega t),\, 0\,]^{\mathrm{T}}$. The amplitude of 0.2 is smaller than the friction magnitude to promote frictional stick. To avoid singularities, it is useful to add damping. Therefore, C is such that, the damping force is roughly 1% of the stiffness force, at the first resonance of the equivalent linear system. The linear system does not have the dry friction element and instead has a pinned middle node, i.e $q_1 := 0$, simulating permanent frictional stick.

The results are in figure 6. The linear system presents a sharp resonance. Regarding the nonlinear system, its solution roughly equals the solution of the linear case, except in the range where an intensity plateau in the periodic steady-state response of q_3 occurs. This also coincides with the sudden jump in the q_1's amplitude. When $\dot{q}_1$ is comparable to $1/\alpha$, the oscillations are large enough to enter the slip regime and the friction starts acting with a large intensity in comparison to the external excitation, thus impeding further movement.

Globally, NLvib and the MHBM agree in the results. Additionally, the results were validated by a (time-domain) shooting method [1].

5.4. Cantilever beams with unilateral contact

This subsection's case study, the system in figure 7, is spatially discretized by the Ritz method as described in [7]. For the linear longitudinal dynamics, linear shape functions were chosen. For the linear bending dynamics, third degree polynomial shape function were used. The two elements are uncoupled in the linear subsystem. The nonlinear element can couple the two elements and it is characterized by two nonlinearities: unilateral contact with contact-separation in the normal (vertical) direction and stick-slip contact in the tangential (horizontal) direction [9].

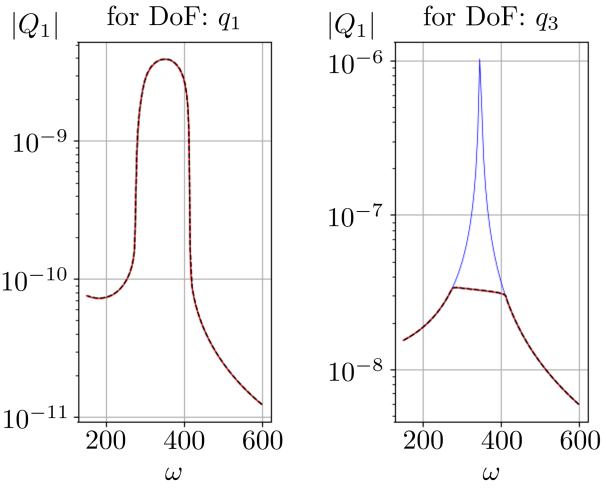

Figure 6: Comparison between NLvib (black line) and MHBM (red line) for the frequency response of nonlinear system represented in figure 5. The linear system with $q_1 := 0$ is in blue.

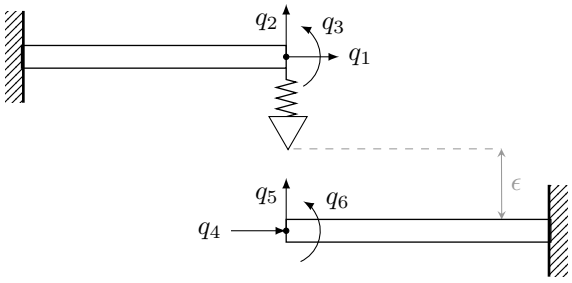

Figure 7: Two cantilever bar+beam elements connected by a frictional contact element. Both elements have the same geometrical and material properties.

The normal reaction force is modeled as a very stiff unilateral spring. The force is given by $f_2^{\mathrm{nl}} = kq_\perp$ where k is the contact stiffness, if the normal displacement, $q_\perp = \epsilon + q_2 - q_5$, is negative; else if $q_\perp \geq 0$, the force is null. Also, $f_1^{\mathrm{nl}} = f^{\mathrm{D.F.}}$ with tangential velocity $\dot{q}_\mathrm{t} = \dot{q}_1 - \dot{q}_4$. Furthermore, $f_f = -\mu f_2^{\mathrm{nl}}$, with μ the friction coefficient. Finally, the action-reaction pairs imply that $\mathbf{f}^{\mathrm{nl}} = [f_1^{\mathrm{nl}},\, f_2^{\mathrm{nl}},\, 0,\, -f_1^{\mathrm{nl}},\, -f_2^{\mathrm{nl}},\, 0]^{\mathrm{T}}$.

Now, numerical values are attributed. For each element, the mass is 1, the length is 1, the axial rigidity is 1/3 and the flexural rigidity 1/3. Also, for the nonlinear element, $k = 500$, $\epsilon = 0.01$, $\alpha = 150$ and $\mu = 0.1$.

To guarantee frictional contact engagement, a clamping force with intensity $P_5 = 0.4$ is applied. And the sinusoidal excitation is characterized by $P_1 = 0.1$. The axial excitation amplitude is larger than the friction intensity, since $P_1 > \mu P_5 > f_f$. The external excitation is $\mathbf{f}^{\mathrm{ext}} = [\, P_1 \cos(\omega t),\, -P_5,\, 0,\, 0,\, P_5,\, 0\,]^{\mathrm{T}}$. To avoid sharp resonances, Rayleigh damping is added: $\mathbf{C} = 0.05\mathbf{K}$.

Regarding the MHBM tool, the smallest truncation order reached while maintaining solver convergence was $H = 3$ and the solver attained results very akin

to the reference, $H = 70$. For $H = 15$, the maximum relative error is 0.2%. And, for $H = 25$, the maximum relative error is 0.02%. Thus, the solution seems to converge with increasing harmonic truncation order.

In the time series in figure 8, one can observe the stick-slip behavior. Stick occurs when the relative velocity first crosses zero and its rate of change suddenly alters such that the relative velocity remains null. The slip behavior is better observed in the time series of the nonlinear force, as it starts the large abrupt steps in f_1^{nl}. Also, the results showed that the bending strains and stresses are time-constant, through all the solutions in the frequency range, because the clamping force is constant and the longitudinal and bending dynamics are uncoupled in the linear system. Finally, the time domain results for multiple solution points were verified by the explicit Runge-Kutta method of order eight [8] (chosen for its high accuracy and stability) initialized with the MHBM's solution at $t = 0$.

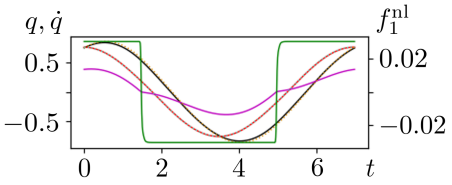

Figure 8: Time-series for the solution at $\omega = 0.9$. The black line and red lines are the MHBM solutions for q_1 and q_4, respectively. And the yellow and cyan lines are the corresponding DOP853 references. The green curve is the dry friction force and the magenta curve is the slip velocity.

6. Co-simulation with a commercial finite element analysis software

This section examines the development of a synergetic workflows between the MHBM and an established modeling and finite element analysis software, Abaqus CAE. This program formulates structural dynamics using the inertia term, internal force and external force, hence the previous equation of dynamics, Eq.(1), is substituted by the equivalent expression: $\mathbf{M}\ddot{\mathbf{q}} + \mathbf{f}^{\text{int}}(\mathbf{q}, \dot{\mathbf{q}}) - \mathbf{f}^{\text{ext}}(\omega, t) = \mathbf{0}$.

The internal force, $\mathbf{f}^{\text{int}}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{C}\dot{\mathbf{q}} + \mathbf{K}\mathbf{q} + \mathbf{f}^{\text{nl}}(\mathbf{q}, \dot{\mathbf{q}})$, can be evaluated in Abaqus by quantifying the initial acceleration of the free system, $\ddot{\mathbf{q}}^{\text{free}}$, in a parallel initial value problem [8] with predefined velocity and displacement. The initial value problem follows:

$$\mathbf{M}\ddot{\mathbf{q}}^{\text{free}} + \mathbf{f}^{\text{int}}(\mathbf{q}^{\text{free}}, \dot{\mathbf{q}}^{\text{free}}) = \mathbf{0}$$
$$\text{and} \quad (\mathbf{q}^{\text{free}}, \dot{\mathbf{q}}^{\text{free}})|_{t=0} = (\mathbf{q}, \dot{\mathbf{q}})|_{t=t_k}\,.$$

Then, the internal force sampled at time t_k is $\mathbf{f}_k^{\text{int}} = -\mathbf{M}\ddot{\mathbf{q}}^{\text{free}}|_{t=0}$. Consequently the time valuations of the alternanting frequency-time scheme, occur entirely in Abaqus. Also, Abaqus outputs the derivatives of the internal force with respect the "displacement" and the "velocity", correspondingly, the linearized stiffness and the linearized viscous damping. This serves to build the jacobian for the Newton-type solver.

Subsequently, the co-simulation procedure was compared with the Abaqus' steady-state dynamic analysis regarding the system in figure 9. This serves as a application with 30 volume elements and 80 nodes with three degrees of freedom each. The external excitation is null except for axial concentrated loads at four tip nodes. Figure 10 has the results. For the

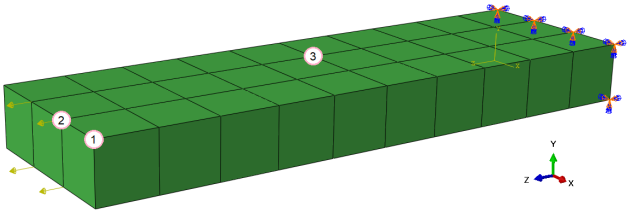

Figure 9: FE model in Abaqus and selected nodes.

axial degrees of freedom of the selected nodes, the frequency responses from Co-Sim and Abaqus' steady state analysis strongly coincide. The error is given by $\|\mathbf{q}^{\text{Co-Sim}}(\omega) - \mathbf{q}^{\text{Abaqus}}(\omega)\|$ and indicates a clear overlap between both methods. Thus, the effectiveness of the co-simulation in computing this system's frequency response is verified, along with the functionality of this framework.

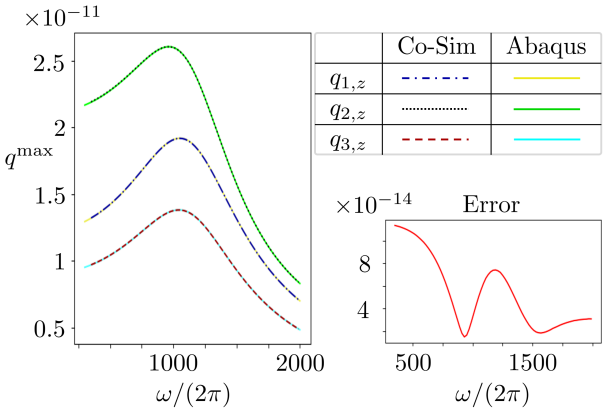

Figure 10: Frequency response of the axial displacements of three nodes and global relative error.

7. Conclusions

7.1. Remarks

A multi-harmonic balance method with a predictor-corrector numerical path continuation solver was successfully built in Python. It uses the alternating frequency time scheme to evaluate the Fourier coefficients of the nonlinear force, by employing the fast Fourier transform. Regarding numerical continuation, the secant and tangent predictors were implemented. Also, the corrector step was built using the Newton-Raphson method and included both the orthogonal and arc

length corrector step parameterizations. Finally, this tool was integrated in a co-simulation procedure along with a commercial software for finite element analysis responsible for the time domain evaluations, in the alternating frequency time scheme.

The implementation of an automatic differentiation procedure makes the program more versatile by supporting a wide range of user provided nonlinear forces. The program can handle a large variety of nonlinear systems without demanding specific-case tailoring. It also displayed sharp accuracy even with relatively low harmonic truncation orders. It also presented stability by computing unstable branches and handling strong nonlinearities. Moreover, this method was faster than time-integration techniques, as expected.

The tool is useful for structural engineering, regarding the determination of loading effects, performance and durability. Firstly, the tool aids vibration analysis by predicting the periodic steady-state oscillatory behavior of spatially discretized structures modeled after real-life engineering systems. Finally, the co-simulation procedure displayed promising results, which suggest new engineering workflow possibilities in the study of large-scale nonlinear systems.

7.2. Future developments

The program could be adapted to incorporate nonlinear forces dependent on higher time-derivatives of $\mathbf{q}$. Homotopy continuation could be implemented [1] to generate the first solution. Moreover, by performing numerical path continuation on different parameters, different bifurcation diagrams can be built. E.g. varying the intensity of the external excitation allows for nonlinear modal analysis [2]. Also, a stability analysis could be implemented. Additionally, the applicability of periodic framework could be expanded by including additional variables that describe certain non-periodic motions [1]. And, ultimately, the co-simulation workflow should also be used in nonlinear applications while enjoying the nonlinear force modeling capabilities of commercial finite element analysis software.

Acknowledgements

The author expresses great appreciation to Dr. Frederico Afonso, from IST, and Francesco Trainotti and Dr. Andreas Zwölfer, from TUM, for the excellent guidance; and deep gratitude to Dr. Fernando Lau and Dr. Daniel Rixen, for enabling this collaboration.

References

[1] M. Krack et al. *Harmonic Balance for Nonlinear Vibration Problems*. Mathematical Engineering. Springer International Publishing, 2019.

[2] G. Kerschen et al. "Nonlinear normal modes, Part I: A useful framework for the structural dynamicist". In: *Mechanical Systems and Signal Processing* 23 (1 2009), pp. 170–194. DOI: 10.1016/j.ymssp.2008.04.002.

[3] F. C. Moon. *Chaotic Vibrations: An Introduction for Applied Scientists and Engineers*. Wiley-VCH, 1987, p. 309.

[4] M. R. Brake, ed. *The Mechanics of Jointed Structures. Recent Research and Open Challenges for Developing Predictive Models for Structural Dynamics*. Springer, 2017, p. 1130.

[5] M. Urabe. "Galerkin's procedure for nonlinear periodic systems". In: *Archive for Rational Mechanics and Analysis* 20 (1965), pp. 120–152. DOI: 10.1007/BF00284614.

[6] T. Detroux et al. "The Harmonic Balance Method for Advanced Analysis and Design of Nonlinear Mechanical Systems". In: *Nonlinear Dynamics, Volume 2*. Ed. by G. Kerschen. Cham: Springer International Publishing, 2014, pp. 19–34. DOI: 10.1007/978-3-319-04522-1_3.

[7] D. J. Rixen. *Structural Dynamics Lecture Notes*. Technische Universität München, Lehrstuhl für Angewandte Mechnik, 2017.

[8] J. Stoer et al. *Introduction to Numerical Analysis*. 3rd ed. Springer New York, NY, 2002.

[9] A. Zwölfer. "Dynamic behaviour optimization of non-linear lap joints Novel flexible multibody system dynamics formulations to move towards digital twins View project". MA thesis. Imperial College London Department of Mechanical Engineering, 2017.

[10] T. M. Cameron et al. "An Alternating Frequency/Time Domain Method for Calculating the Steady-State Response of Nonlinear Dynamic Systems". In: *Journal of Applied Mechanics* 56.1 (1989), pp. 149–154. DOI: 10.1115/1.3176036.

[11] B. Cochelin et al. "A high order purely frequency-based harmonic balance formulation for continuation of periodic solutions". In: *Journal of Sound and Vibration* 324.1-2 (2009), pp. 243–262. DOI: 10.1016/j.jsv.2009.01.054.

[12] L. Woiwode et al. "Comparison of two algorithms for Harmonic Balance and path continuation". In: *Mechanical Systems and Signal Processing* 136 (2020). DOI: 10.1016/j.ymssp.2019.106503.

[13] D. V. Widder. *Advanced calculus*. Courier Corporation, 1989.

[14] E. L. Allgower et al. *Introduction to Numerical Continuation Methods*. Society for Industrial and Applied Mathematics, 2003.

[15] J. R. R. A. Martins et al. *Engineering Design Optimization*. Cambridge University Press, 2021.

[16] A. F. Vakakis. "Fundamental and subharmonic resonances in a system with a '1-1' internal resonance". In: *Nonlinear Dynamics* 3.2 (1992), pp. 123–143. DOI: 10.1007/BF00118989.