Journal of Sound and Vibration 597 (2025) 118791

Contents lists available at ScienceDirect

Journal of Sound and Vibration

journal homepage: www.elsevier.com/locate/jsvi

Response to "Discussion of S. Ponsioen, S. Jain and G. Haller: 'Model reduction to spectral submanifolds and forced-response calculation in high-dimensional mechanical systems', Journal of Sound and Vibration 488, 2020, pages 1-23"

Sten Ponsioen [a], Shobhit Jain [b], George Haller [c,*]

[a] MUFG Investor Services, 25 Ropemaker St, London EC2Y 9LY, United Kingdom
[b] Delft Institute of Applied Mathematics, TU Delft, Mekelweg 4, 2628 CD Delft, Netherlands
[c] Institute for Mechanical Systems, ETH Zürich, Leonhardstrasse 21, 8092 Zürich, Switzerland

ABSTRACT

We have reproduced our disputed simulation results from our original article as well as the simulations of the discussion article of Krack and Gross referenced above in the title. We have found several inaccuracies and omissions in the discussion article that invalidate its authors' main claims. We also briefly address inherent performance differences between the SSMTool and the NLvib algorithms to the benefit of the former.

1. Introduction

In the following sections, we give detailed responses to the points raised by Krack and Gross [1] (henceforth referenced as [KG]) regarding the use of NLVib by Ponsioen, Jain and Haller [2] (henceforth referenced as [PJH]).

2. Discretized Bernoulli beam with nonlinear spring

(1) [PJH] compared run times of SSMTool and NLVib on *full* finite-element models. The run time reported for NLVib by [KG] on this problem is *not* for a full finite-element model. Rather, it is for a significantly reduced model obtained from explicit condensation, whose treatment by harmonic balance requires several orders of magnitude less numerical effort, as [KG] discuss in their Section 1.1. This numerical reduction trick is only applicable to problems with a low number of single-coordinate nonlinearities, such as the simple Bernoulli beam with a single cubic problem we selected. As [KG] note, after this trick, the numerical effort for NLvib becomes *independent* of the number of degrees of freedom, i.e., it is the same for 1DOF and 10,000 DOF. Indeed, as admitted by [KG], without using this trick, they observed computation times similar to those of [PJH]. The nonlinear Timoshenko beam example (see Section 2 below) and general nonlinear finite-element models are not tractable via harmonic condensation.

(2) On a broader note, a detailed analysis of SSMTool's algorithm and harmonic balance (the algorithm used by NLvib) reveals major distinctions in computational complexity to the benefit of SSMTool for finite-element problems (see [3]). This benefit is compounded by the fact that SSM reduction does not just provide periodic response under periodic forcing, as harmonic balance does. Rather, SSM reduction also gives a reduced-order model from which the stability and the domains of attractions of the periodic responses can be deduced.

DOI of original article: https://doi.org/10.1016/j.jsv.2024.118808.
* Corresponding author.
E-mail address: georgehaller@ethz.ch (G. Haller).

https://doi.org/10.1016/j.jsv.2024.118791
Received 11 October 2024; Received in revised form 21 October 2024; Accepted 22 October 2024
Available online 30 October 2024

Table 1
Performance of NLvib 1.3 and 1.4 on the Timoshenko beam example for various choices of parameters. Computation performed using MATLAB 2023b installed on a PC with 11th Gen. Intel Core i5 at 2.6 GHz and 32 GB RAM.

Numerical parameter	**Appropriate values [KG]**	**Appropriate values [KG]**	**Updated values I for [PJH]**	**Updated values II for [PJH]**	**Original values [PJH]**
(Nlvib version)	**1.4**	**1.3**	1.3	1.3	1.3
Number of time steps used to sample periodic response (‘N’)	41	41	2^7	2^7	2^7
Number of Harmonics (‘H’)	10				
Initial solution guess (‘y0’)	Linearized response at the starting frequency				
Scaling parameter (‘qscl’)	**254.16**	**254.16**	**254.16**	**0.7**	**0.7**
Initial stepsize (‘ds’)	0.05	0.05	0.05	1	1
Optional settings					
Scaling of variables depending on the scaling parameter ‘qscl’ (‘DSCALE’)	[qscl*ones(size(y0));Om_s]				
Dynamic scaling of DSCALE ('dynamicDscale')	**1**	**1**	**1**	**1**	**0**
Results					
Solution points obtained	106	479	121	1,747	87,340
Computation time hour:min:sec	0:1: 53	0:14: 45	0:5:42	1:57:05	**>3 days**

Summary: Based on points (1)-(2), we stand by our conclusion that NLVib (including its latest version 1.4) does not run on typical nonlinear finite-element problems in a reasonable amount of time once the number of degrees of freedom reaches a few thousands (more on this in point (8) below).

3. Discretized, forced, geometrically nonlinear Timoshenko beam

We have repeated the NLvib calculations for this example using both the current version 1.4 used by [KG] in [1] and the latest version available in 2020 (version 1.3) used by [PJH] in [2]. The list of numerical parameters required by NLvib and the *optional* settings we have considered are given in the first column of Table 1.

We first carried out these calculations under the parameter values used by [KG] with NLvib 1.4 (second column of Table 1); then under the same parameters with NLvib 1.3 (third column of Table 1); then with the number of time steps changed to $N = 2^7$ in NLvib 1.3, as used by [PJH] (fourth column of Table 1); then with $N = 2^7$ and with the scaling parameter changed to $\mathrm{qscl} = 0.7$ in NLvib 1.3, as used by [PJH] (fifth column of Table 1); then finally with $N = 2^7$, $\mathrm{qscl} = 0.7$ but without using the optional binary parameter ‘dynamicDscale’, i.e., with $\mathrm{dynamicDscale} = 0$ as for [PJH] in [2] (last column of Table 1).

Based on the results in Table 1, we observe the following:

(3) The recommended (default) number of time steps per period (the parameter N) in NLvib 1.3 was between 2^6 and 2^8, which we followed with our choice of 2^7 in [PJH]. This default choice was only updated *later* in NLvib 1.4 to $4H + 1$, motivated by the paper [4] from 2020. This fact contradicts Remark 2 of [KG], which states: “It is important to emphasize here that the reduced computation times observed in this discussion are not due to any scientific or technological advancement achieved in the meantime (such as faster computers or more efficient linear algebra libraries).”

(4) [KG] note the higher accuracy of NLvib relative to SSMTool for $H = 10$ harmonics near the resonance peak. They describe the max. 8% error from SSMTool as significant. We note that this result is from the *lowest possible* (i.e., cubic) nonlinear approximation of the SSM-reduced normal form used in [PJH]. Several others papers (see, e.g., [3,5,6]) show that such an error can be eliminated by passing to a higher polynomial order of approximation (order 5 or 7), which was not the objective of [PJH]. In principle, it is possible for NLvib to locate periodic orbits outside the domain of convergence of polynomial approximations for SSMs. This, however, has not yet been demonstrated on any example that we are aware of.

(5) While the basic algorithm may be the same between NLvib 1.3 and 1.4, an order of magnitude performance difference is apparent due to changes in default settings such as relaxation of error tolerances. Simply using the ‘appropriate’ parameters of [KG] in the original NLvib v1.3 already gives an order of magnitude speedup for the same example: runtime goes down from 14.76 min to about 1.76 min, as seen in the last row of Table 1. This again contradicts the claim of [KG] in their Remark 2.

(6) The update of the step number N from 2^7 (used by [PJH]) to the recommended value of $(4H + 1)$, by itself, would in fact *slow down* the calculations performed in [PJH], as one sees from the last row of Table 1 (14.76 min vs 5.69 min). This is contrary to the assertion of [KG], who attribute the lack of tractability of the problem in [PJH] to the large N value used.

The default value of the optional scaling parameter ‘dynamicDscale’ was set to 0 in NLvib 1.3. [KG] set the binary value of this parameter $\mathrm{dynamicDscale} = 1$. The last two columns of our Table 1 show that this parameter indeed crucially affects NLvib’s performance: simply turning this parameter on results in a tractable computation time for NLvib on our Timoshenko beam example. [PJH] used NLvib 1.3 for nonlinear frequency response analysis (specified via the analysis option ‘frf’ in NLvib), which is also performed in 5 out of the 9 examples available for NLvib 1.3 at the time,[1] including also a beam example.

(7) If the dynamicDscale parameter was indeed clearly known to be crucially important at the time of [PJH], as implied by [KG], then the question arises: Why did the authors of NLvib 1.3 set $\mathrm{dynamicDscale} = 0$ *by default*? Even more importantly: Why was $\mathrm{dynamicDscale} = 0$ used in *all* of their own 5 examples involving nonlinear frequency response analyses similar to those in [PJH]?

Summary: [PJH] made a best effort to use NLVib 1.3, using the information and use cases made available to users of NLVib 1.3 *at the time*. In contrast and contrary to their claim, [KG] use knowledge and experience gained during the *last* 4 *years* that have passed since then.

4. Current state of the art

[KG] report computation times for current NLvib 1.4 that are shorter than those reported by [PJH] for the version of SSMTool available. They pose the open question whether computations by the current version of SSMTool continue to be slower than NLvib. Our response to this question is as follows:

(8) We have found forced response calculation via the current NLvib 1.4 to be an intractable task on a 1,320 DOF nonlinear finite element model of a shallow arch (Example 6.3 in [5]), even upon following the procedure and the parameters proposed by [KG]. Specifically, the machine referenced in the caption of Table 1 took approximately 4 days to assemble the Jacobian of the harmonic balance residual via NLvib 1.4, and hence, could not even finish the computation of a single solution point on the frequency response curve in the allotted timeframe (see [7] for the simulation files). In contrast, SSM-based reduced models produce forced response curves for any required number of forcing amplitudes within a few seconds (see [5] for details). This improved performance of SSMTool can be attributed to later developments that perform SSM computations in physical coordinates using the minimal set of eigenvectors [5]. More recent developments enable similar (nonintrusive) SSM-based computations within minutes for nonlinear finite-element models with more than a million DOF (see [6]).

(9) As stated by [KG], a further speedup of harmonic-balance-based numerical continuation may indeed be possible via more efficient implementations or parallelization strategies, such as those in [8]. Note, however, that the promise of the parallelization strategy has only been demonstrated in [8] on examples with *no more than 58* (!) DOF. This is well below the practically relevant number of DOFs (at least tens of thousands) for industry-grade problems.

Summary: As already discussed in our point (2) above, the inherent computational complexity advantage of SSM reduction over harmonic-balance-based continuation schemes (such as NLvib) becomes insurmountable on high-dimensional nonlinear finite-element models.

[1] These five examples were under the names ’02_twoDOFoscillator_cubicSpring’, ’03_twoDOFoscillator_unilateralSpring’, ’06_twoSprings_geometricNonlinearity’, ’07_multiDOFoscillator_multipleNonlinearities’, and ’08_beam_tanhDryFriction’.

CRediT authorship contribution statement

Sten Ponsioen: Writing – review & editing, Conceptualization. **Shobhit Jain:** Writing – review & editing, Formal analysis, Conceptualization. **George Haller:** Writing – review & editing, Writing – original draft, Conceptualization.

Declaration of competing interest

The authors declare no competing interests to declare.

Data availability

Data is already included or referenced in the text.

References

[1] M. Krack, J. Gross, Discussion of "S. Ponsioen, S, Jain and G. Haller: 'Model reduction to spectral submanifolds and forced-response calculation in high-dimensional mechanical systems', Journal of Sound and Vibration 488, 2020, pages 1-23", J. Sound Vib. (2024) in press.
[2] S. Ponsioen, S. Jain, G. Haller, Model reduction to spectral submanifolds and forced-response calculation in high-dimensional mechanical systems, J. Sound Vibr. 488 (2020) 115640, http://dx.doi.org/10.1016/j.jsv.2020.115640.
[3] T. Thurnher, G. Haller, S. Jain, Nonautonomous spectral submanifolds for model reduction of nonlinear mechanical systems under parametric resonance, Chaos 34 (2024) 073127, http://dx.doi.org/10.1063/5.0168431.
[4] L. Woiwode, N.N. Balaji, J. Kappauf, F. Tubita, L. Guillot, C. Vergez, B. Cochelin, A. Grolet, M. Krack, Comparison of two algorithms for harmonic balance and path continuation, Mech. Syst. Signal Process. 136 (2020) 106503, http://dx.doi.org/10.1016/j.ymssp.2019.106503.
[5] S. Jain, G. Haller, How to compute invariant manifolds and their reduced dynamics in high-dimensional finite-element models, Nonlinear Dynam. 107 (2022) 1417–1450, http://dx.doi.org/10.1007/s11071-021-06957-4.
[6] M. Li, T. Thurner, Z. Xu, S. Jain, Data-free non-intrusive model reduction for nonlinear finite element models via spectral submanifolds, 2024, submitted for publication https://arxiv.org/abs/2409.10126.
[7] S. Jain, Simulation files for the nonlinear finite element model of a shallow arch, 2024, https://github.com/jain-shobhit/shell_example_NLvib.
[8] J. Gross, V. Gupta, C. Berthold, M. Krack, A new paradigm for multi-fidelity continuation using parallel model refinement, Comput. Methods Appl. Mech. Engrg. 423 (2024) 116860, http://dx.doi.org/10.1016/j.cma.2024.116860.