Online Implementation of the Local Eigenvalue Modification Procedure for High-rate Model Assimilation

Alexander B. Vereen[1], Emmanuel A. Ogunniyi[1], Austin R.J. Downey[1,2],
Jacob Dodson[3], Adriane G. Moura[4], and Jason D. Bakos[5]

[1] Department of Mechanical Engineering
[2] Department of Civil and Environmental Engineering
University of South Carolina, Columbia, SC 29208

[3] Air Force Research Laboratory, Munitions Directorate
Eglin Air Force Base, FL 32542

[4] Applied Research Associates
Emerald Coast Division, Niceville, FL 32578

[5] Department of Computer Science and Engineering
University of South Carolina, Columbia, SC 29208

ABSTRACT

High-rate structural health monitoring of active structures operating in high-rate dynamic environments empowers the execution of preventative behavior in response to structural degradation or external stimuli. Examples of structures operating in high-rate dynamic environments include hypersonic vehicles, space crafts, and ballistic packages. The effective selection of reactive actions to be taken in real-time requires an up-to-date model of the structure's state. Importantly, the short timescale of relevance to these structures means that the model must be continuously updated with a time step of 1 millisecond or less. However, traditional frequency-based methods for updating the finite element model online require solving the generalized eigenvalue problem, which becomes more complex as the number of nodes or FEA model increases, thereby increasing computational time. In this work, the local eigenvalue modification procedure is put forward to accelerate the extraction of natural frequencies from finite element models updated online. The local eigenvalue modification procedure works by pre-computing the eigenvalue solution to a reference state of the system, then computes the single (i.e., local) change in the modal domain from the reference state to the current state online. The modal domain update in the local eigenvalue modification procedure bypasses the general eigenvalue problem, which is the most expensive computational step. For the online implementation of the state estimation, the Dynamic Reproduction of Projectiles in Ballistic Environments for Advanced Research testbed is used. The testbed allows for the controlled movement of a pinned condition attached to an otherwise free cantilever beam. In previous studies, the testbed has been used with the generalized eigenvalue solver in a frequency-based model assimilation approach to infer the most likely position of the pinned condition (i.e., state of the structure). This work reports the effectivity of the local eigenvalue modification procedure compared to the generalized eigenvalue solution of the system for inference accuracy while varying the nodes dedicated to the analysis. The optimal efficiency of the system's approach is explored for the testbed-based health assessments. Timing results and effects of sensor noise on the system are discussed in detail.

Keywords: High-rate, Modal analysis, FEA, eigenvalue, LEMP

INTRODUCTION

Active control can provide critical intervention to reduce or negate damage to a structure in situ; However, it requires the acquisition of the state of the structure as a prerequisite to the decision and implementation of an intervention [1]. In high-rate systems, estimating the structure's state through modal analysis allows sparse networks of sensors to be used to attain the natural frequencies of the structure [2]. The natural frequencies of a structure can be helpful despite the non-unique description of the structure given its shape and boundary conditions. Acquiring the natural frequencies of a structure and then proposing an explanatory model of the system can ascertain the system's state using sparse sensor networks [2, 3].

Candidate models are evaluated for their natural frequencies by solving the eigenvalue problem of the proposed models. State acquisition by way of modal analysis and candidate model testing is broken into three major tasks [2]. The first task entails system excitation followed by hardware acquisition and digitization. Second, frequency analysis is performed. Third, candidate models are proposed and evaluated for their similarity to the observed frequency response. The most time-consuming procedure is solving the eigenvalue problem for the frequencies of a candidate model. This step scales in $O(n^3)$ time by traditional mode-extraction methods [4–7].

For high-rate state estimation, attaining estimations of the system's state in less than one millisecond can enable responsive interventions in active aerospace structures [1]. Modal analysis can be used in all structural geometries; however, the complexity required for a minimally accurate solution can increase the required computations. The Dynamic Reproduction of Projectiles in Ballistic Environments for Advanced Research (DROPBEAR) testbed is used to explore a functional implementation of a high-rate dynamic system. The testbed allows for the controlled movement of a pinned condition attached to an otherwise free cantilever beam. This system is useful as it can be modeled as a 1-D Finite Element formulation and can instate a repeatable change to its boundary condition.

A myriad of known algorithms can attain the eigenvalue solution, each responding in a worst case of $O(N^3)$ where N is the finite element mesh size, and a best case of $O(N^2)$ [6]. However, in the high-rate state estimation problem, there is only a requirement to retrieve the eigenvalues of a proposed model. The Local Eigenvalue Modification Procedure (LEMP) avoids incurring the operational complexity of a mode-extraction algorithm by obtaining the eigenvalues of a modified state through modification of known eigenvalues and eigenvectors of the system [3, 8]. The central proposal of this work is to avoid solving the eigenvalue problem online by approximating the eigenvalues from a reference state and a single added change to the state. In a timing investigation of LEMP's online update step vs. a general eigen solution method for the eigenvalue problem, the implementation of LEMP provided an approximately $100\times$ reduction in time. Furthermore, it showed operational complexity to be directly proportional to the mesh size of the finite element model. In an implementation of the algorithm in hardware, the speed of LEMP was shown to reliably provide good estimates of the state while achieving the 1 millisecond time target, which was out of reach for the mode-extraction algorithms. The merit of this work is in exploring and quantifying the result of a direct algorithmic comparison between LEMP and traditional mode-extraction methods and showing the full state estimation implementation loop in hardware and the resulting time per estimate.

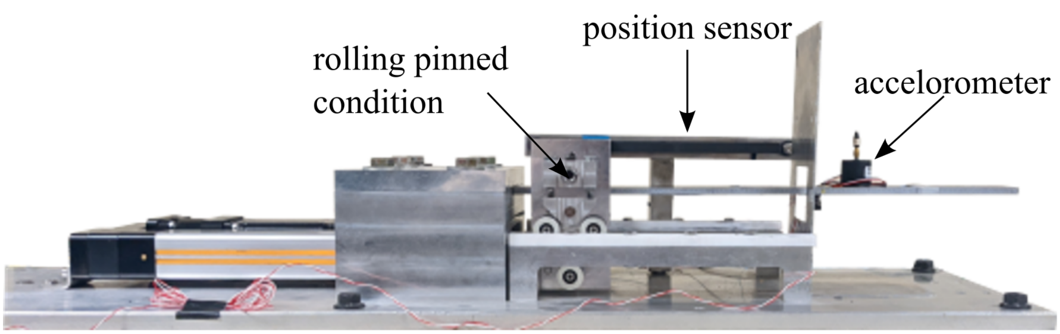

Figure 1: Dynamic Reproduction of Projectiles in Ballistic Environments for Advanced Research (DROPBEAR) testbed.

BACKGROUND

For the online implementation of the state estimation, the Dynamic Reproduction of Projectiles in Ballistic Environments for Advanced Research (DROPBEAR) testbed is used. The testbed allows for the controlled movement of a pinned condition attached to an otherwise free cantilever beam and is shown in Figure 1 and specifications for the system are provided in Table 1. The sensors utilized in the Hardware in the Loop (HIL) implementations were the Honeywell position sensor (model SPS-

L075-HALS) and a PCB Piezotronics accelerometer (model 393B05) at the end of the beam. Data from the trial recordings on the beam were played back for all HIL tests in this study to eliminate as much potential noise from the physical process allowing a fair comparison between algorithms. Computations were carried out using an eight-core Intel® Xeon® E5-2618L v3 processor mounted in a PXIe-8880 manufactured by National Instruments processing acceleration data digitized using a 14-bit ADC module (PXI-6133) mounted in a common PXI chassis (PXIe-1082).

Table 1: Specifications for the configuration of the DROPBEAR testbed used in this work.

beam properties		actuator properties	
beam length	501 mm	roller travel	84 mm
beam width	51 mm	roller max velocity	3.32 m/s
beam thickness	6.66 mm	roller start position	57.73 mm
beam Young's modulus	209.5 GPa		
beam density	7900 $\frac{kg}{m^3}$		

For this work, a previously developed 1-D FEA model describes the beam, and localized sub-sampling was implemented [3]. Subsampling allows a reduction in the number of scenarios tested at each estimation. Instead of testing every location along the beam each time step as a likely position, sampling a normal distribution around the last known position is used. The selected samples in the distribution are referred to as particles in this work. Subsampling within the scenario space is a necessary tradeoff for speed in high-rate state estimation. For a 1-D model of the beam with a mesh size of N, convergence to the most likely position pinned location to exactly one step would take solving for the frequencies of that beam in N models. Subsampling at least the current position, a minimally forward position, and a minimally backward position guarantee convergence in at most N steps testing three models per step. If, however, the pinned condition is local to its last known location, as is a safe assumption in tracking problems, then on average, the convergence will only take the difference of minimal steps between them. Allowing for a large reduction in calculations to achieve convergence quicker than the overhead of testing each of N points each step would allow.

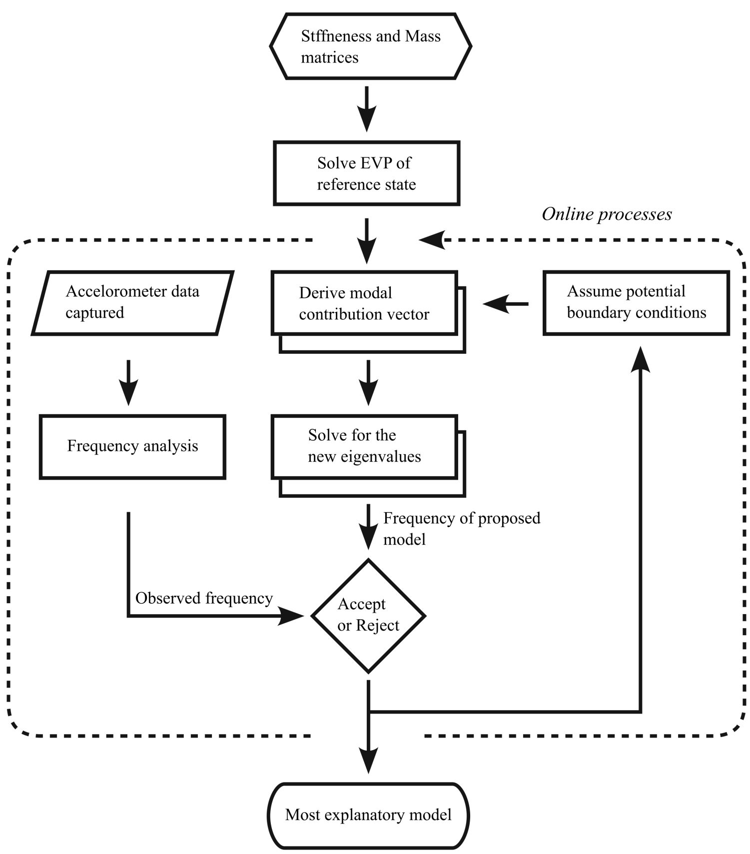

Figure 2: Data flow for the online LEMP algorithm used in this work.

Modal Analysis requires solving the eigenvalue problem; however, in the general case, a matrix inversion is required where the operational complexity of solving the inversion grows proportionally as a square of the size of the matrices. For example, an Euler-Bernoulli beam formulation is $(2N)^2$ where N is the number of nodes in the beam. LEMP avoids solving the eigenvalue problem and instead updates the solution as a change to the modal space representation [8]. The algorithmic implementation of LEMP for the online configuration used in this work is shown in Figure 2. As a result, solving for the frequencies of each model becomes a linear time problem in the state estimation. Therefore, a 1-Dimensional system only requires three solutions to the eigenvalue problem, and a P-Dimensional tracking requires only $2P$ orthogonal models as a minimum subsample. In terms of big O notation, LEMP allows a time complexity of $O(PN)$ P being the dimensionality and the N the size of the mesh of describing the structure while solving the eigenvalue problem incurs a time complexity of $O(PN^3)$. The authors have also previously demonstrated the viability of the LEMP algorithm [3] and a corresponding real-time solver [9].

ANALYSIS

Two timing studies were implemented. First, a timing study on an algorithm in each simulation step is carried out. LEMP was used to update a base cantilever beam model to a beam with a pinned condition, in comparison to the general eigen solution; both solving the same system. This was done for mesh sizes 20 through 200 at each multiple of 10; 1000 times per unique combination of parameters. The second study was on the full online implementation using the HIL setup. The parameters varied over this study were the mesh size, the number of particles solved for, and the FFT window size. Increasing the window size allows more frequency resolution at the cost of less time resolution and more computational time. Increasing the mesh size allows for more positional resolution resulting in increased computational time. Similarly, sampling more particles allows quicker convergence to a given position at the cost of computational time.

The first timing study shows that LEMP managed to be consistently solved in less time than the general eigen solution algorithm; as shown in Figure 3. Note that the y-axis is in $\log_{10}$. The diagrams from the timing study show the variations in timing that are an artifact of implementing the proposed method on a non-real-time operating system. This variation comes from the Windows operating system having to que operations alongside running of the LEMP and a LabVIEW native generalised eigenvalue solver. In Table 2 it is shown in detail that for this first algorithmic study that a $100\times$ reduction in execution time was found a the 40 node mesh size on average over 1000 trials. Notably, This gap in performance grows with mesh size.

Table 2: Selected timing values extracted from Figure 3.

	nodes				
	40	80	120	160	200
LEMP (μs)	38.90	36.51	36.16	39.26	45.39
GES (μs)	3,804	21,974	73,593	179,314	344,519

In the second timing study, the HIL implementation is investigated. Figure 4 shows the mean completion times for the LEMP algorithm over relevant parameters, namely the model size and the number of candidate models tested. In addition, it is observed

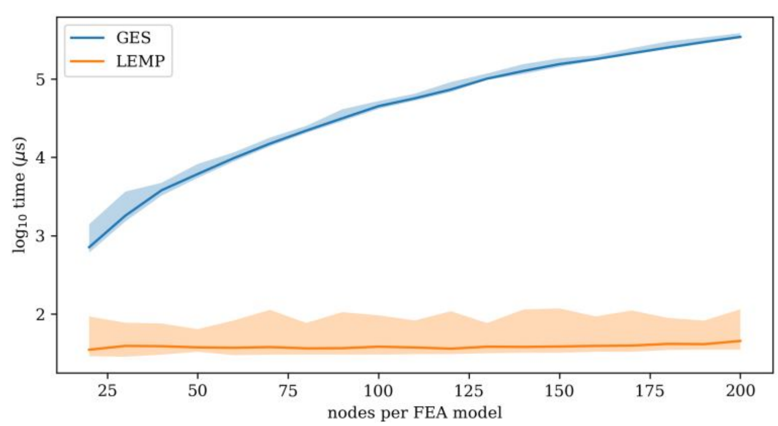

Figure 3: Time execution vs mesh size versus nodes for Local Eigenvalue Modification Procedure (LEMP) and General Eigenvalue Solver (GES).

that there are large time savings to be had using the LEMP model updating method. With the hardware in the loop, it was found that the implementation using LEMP required considerably more time than the pure algorithmic study pointing to there being room for optimization in the implementation to reach the same performance.

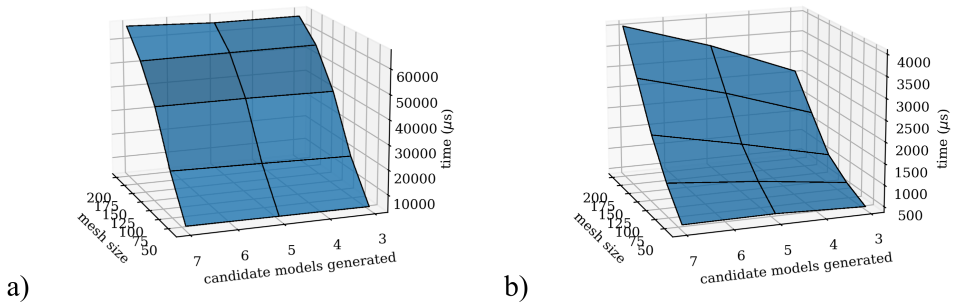

Figure 4: Time for FEA execution versus nodes for a) GES and b) LEMP each using a FFT window of 5000 points (0.02 s) .

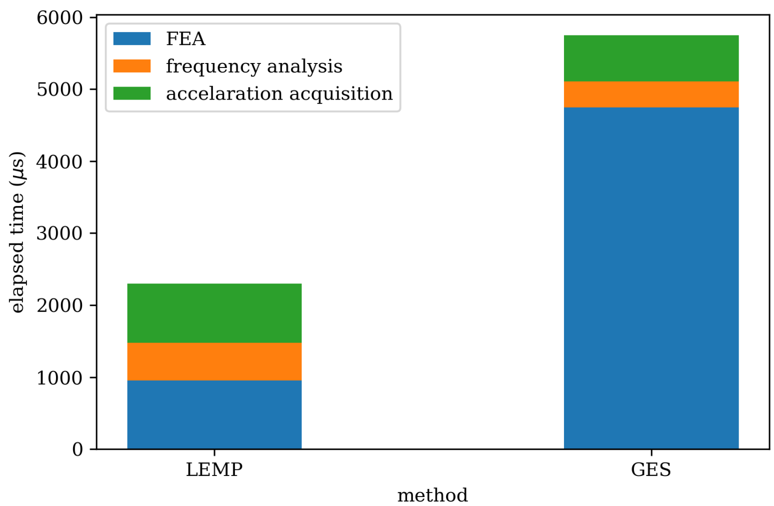

Figure 5: Time execution for Local Eigenvalue Modification Procedure (LEMP) vs a General Eigenvalue Solver (GES).

Figure 5 shows the time allocation of each step of the processes for a configuration with a mesh size of 40, testing 5 models, and a FFT window of length 5000 data points (0.02 s) The time saving provided by LEMP is seen in the time required to solve the FEA analysis. It is seen that with LEMP solving the FEA model for the considered problem requires a similar time to that for the data acquisition and frequency analysis steps.

Results for the second timing study focused are on HIL implementation are reported in Figure 6 and Figure 7. Figure 6 reports a typical time response obtained using theLEMP model updating method compared to the measured boundary condition change instated by the pin. For these results, an FEA model with 80 nodes and an FFT window size of 15000 (0.06 s) was utilized.

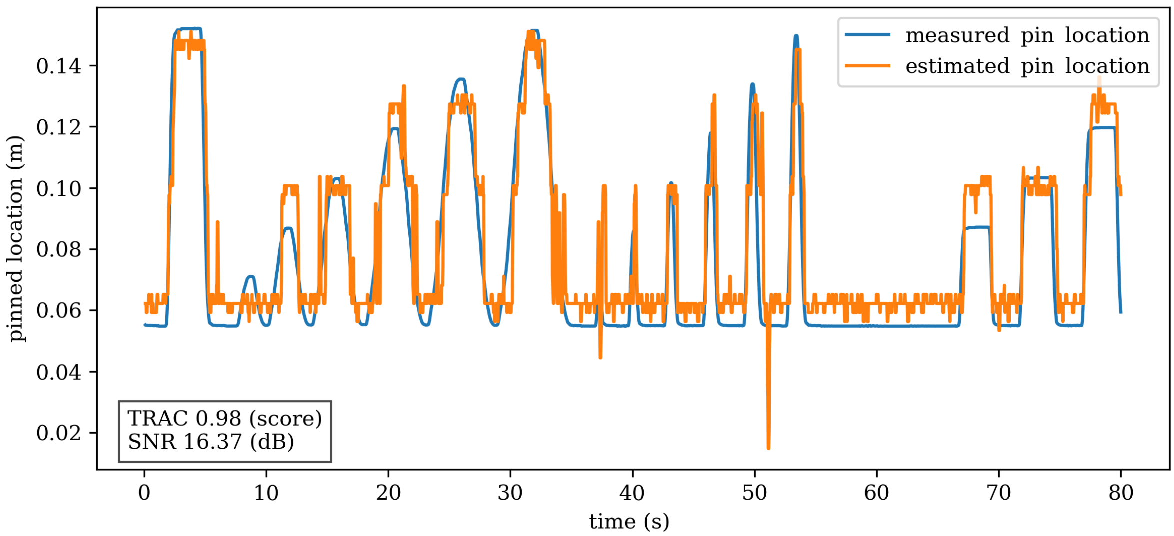

Figure 6: HIL position tracking implementation using LEMP with a mesh size of 120 nodes, testing 5 candidate models, and using a FFT window size of 5000 points (0.02 s).

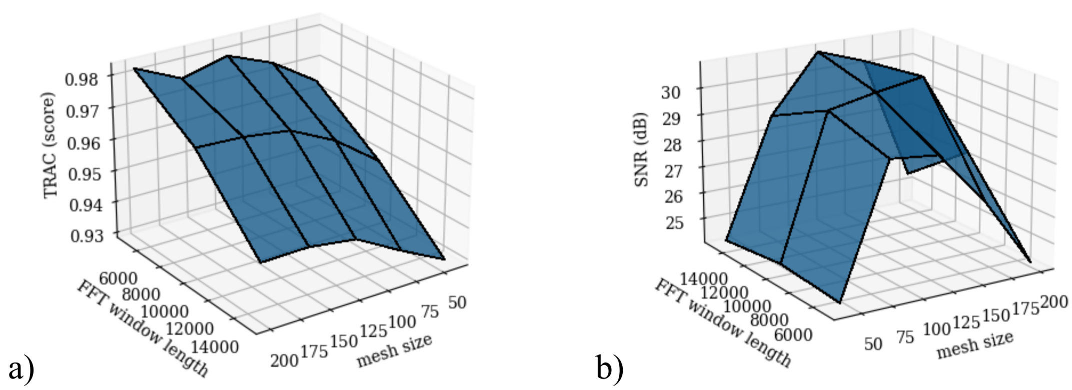

Figure 7: Performance metrics for HIL implementation shown in a) TRAC and b) the SNR of LEMP as a function of mesh size and FFT window length.

Figure 7 reports performance metrics for the HIL implementation of LEMP as a function of mesh size and FFT window length. The Time Response Assurance Criterion (TRAC) is used to gauge the similarity between time traces by comparing the numerical error and time delay of each estimation on a scale from 0 to 1 [10]. To expand, TRAC shows how well the signals replicate each other, disregarding the amplitude of each signal. As TRAC is sensitive to changes in phase, it shows the penalty for extending the FFT window size; indicating a phase lag incurred in the state estimation. Due to LEMP's high-speed, latency from the mesh size was shown to have a limited impact on the TRAC score in comparison. The Signal to Noise Ratio (SNR) performance metric shows a tradeoff between mesh size and the FFT window length. While, obtaining the optimal mesh size was found to be an important parameter to consider, the relative improvement from 24 to 31 dB shows that the LEMP solver is still useful even at less-than-optimal parameter selections.

CONCLUSION

In this work, the local eigenvalue modification procedure (LEMP) is demonstrated to accelerate the extraction of natural frequencies from finite element models updated online with real-time constraints on real-time hardware. LEMP's linear time cost makes it ideal for further extensions and studies into high-rate state estimations. For the online implementation of the state estimation, the Dynamic Reproduction of Projectiles in Ballistic Environments for Advanced Research (DROPBEAR) testbed is used. Timing reports for LEMP in comparison to the generalized eigenvalue method were reported showing speed-ups of $10\times$ in the HIL implementation. This is shown to not necessarily be a ceiling for the approach as in the algorithmic study the for meshes of the same size a $100\times$ speed up was shown. Performance metrics for the HIL implementation of LEMP were also discussed in detail. In future work, the algorithm will be extended to a working prototype of a 2D implementation of this system.

ACKNOWLEDGEMENTS

This material is based upon work supported by the United States Air Force through the Air Force Research Lab Summer Faculty Fellowship Program and by the Air Force Office of Scientific Research (AFOSR) through award no. FA9550- 21-1-0083. This work is also partly supported by the National Science Foundation Grant numbers 1850012, 1956071, and 1937535. The support of these agencies is gratefully acknowledged. Any opinions, findings, and conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation or the United States Air Force.

REFERENCES

[1] Jacob Dodson, Austin Downey, Simon Laflamme, Michael D Todd, Adriane G Moura, Yang Wang, Zhu Mao, Peter Avitabile, and Erik Blasch. "High-rate structural health monitoring and prognostics: An overview". *Data Science in Engineering, Volume 9*, pages 213–217, 2022.

[2] Austin Downey, Jonathan Hong, Jacob Dodson, Michael Carroll, and James Scheppegrell. "Millisecond model updating for structures experiencing unmodeled high-rate dynamic events". *Mechanical Systems and Signal Processing*, 138:106551, 2020.

[3] Seong Hyeon Hong, Claire Drnek, Austin Downey, Yi Wang, and Jacob Dodson. "Real-time model updating algorithm for structures experiencing high-rate dynamic events". In *ASME 2020 Conference on Smart Materials, Adaptive Structures and Intelligent Systems*. American Society of Mechanical Engineers, sep 2020.

[4] BF Feeny and U Farooq. "A nonsymmetric state-variable decomposition for modal analysis". *Journal of Sound and Vibration*, 310(4-5):792–800, 2008.

[5] I Dhillon. "A new o (n2) algorithm for the symmetric tridiagonal eigenvalue". *Eigenvector Problem*, 1997.

[6] Ed S Coakley and Vladimir Rokhlin. "A fast divide-and-conquer algorithm for computing the spectra of real symmetric tridiagonal matrices". *Applied and Computational Harmonic Analysis*, 34(3):379–414, 2013.

[7] Hiroto Imachi and Takeo Hoshi. "Hybrid numerical solvers for massively parallel eigenvalue computations and their benchmark with electronic structure calculations". *Journal of Information Processing*, 24(1):164–172, 2016.

[8] Peter Avitabile. "Twenty years of structural dynamic modification-a review". *Sound and Vibration*, 37(1):14–27, 2003.

[9] Emmanuel A Ogunniyi, Austin RJ Downey Jr, and Jason D Bakos. "Development of a real-time solver for the local eigenvalue modification procedure". In *Sensors and Smart Structures Technologies for Civil, Mechanical, and Aerospace Systems 2022*, volume 12046, pages 253–267. SPIE, 2022.

[10] Peter Avitabile and Pawan Pingle. "Prediction of full field dynamic strain from limited sets of measured data". *Shock and Vibration*, 19(5):765–785, 2012.