The Journal of Open Source Software

pyOMA2: A Python module for conducting operational modal analysis

Dag P. Pasca [1¶] **and Diego Federico Margoni** [2]

1 Norsk Treteknisk Institutt, Oslo, Norway **2** Politecnico di Torino, Italy ¶ Corresponding author

DOI: 10.21105/joss.07656

Software

- Review
- Repository
- Archive

Editor: Fabian-Robert Stöter

Reviewers:

- @Nitnelav
- @e-dub

Submitted: 12 September 2024
Published: 17 November 2025

Summary

Operational modal analysis (OMA) has garnered considerable attention from the engineering community in recent years and has established itself as the preferred method for estimating the modal properties of structures in structural health monitoring (SHM) applications, particularly in civil engineering. The key advantage of OMA over experimental modal analysis (EMA) is its ability to derive modal parameters solely from output measurements taken during the structure's normal operation. This makes OMA a more practical and efficient approach, as opposed to the traditional EMA, which requires both input and output data.

Statement of need

`pyOMA2` is the latest and improved version of the `pyOMA` module (Pasca et al., 2022), a Python library specifically designed for conducting operational modal analysis. While its predecessor relied on procedural workflows, `pyOMA2` fully utilises Python's object-oriented capabilities to offer a comprehensive suite of tools for performing OMA.

Notable improvements over the previous version include support for single- and multi-setup measurements, allowing users to handle multiple acquisitions that combine reference and roving sensors; enhanced user-friendliness through a broad range of tools for pre-processing and visualising data; interactive plotting that enables users to select desired modes directly from algorithm-generated graphs; a geometry-definition feature to visualise mode shapes on tested structures; and, since version 1.1.1, the possibility to estimate uncertainty bounds of modal properties for the SSI family of algorithms.

The following algorithms are included in the module:

- Frequency domain decomposition (FDD) (Brincker, Zhang, et al., 2001);
- Enhanced frequency domain decomposition (EFDD) (Brincker, Ventura, et al., 2001);
- Frequency spatial domain decomposition (FSDD) (Zhang et al., 2010);
- Reference-based covariance driven stochastic subspace identification (SSIcov) (Peeters & De Roeck, 1999; Reynders, 2012; Van Overschee & De Moor, 2012);
- Reference-based data driven stochastic subspace identification (SSIdat) (Peeters & De Roeck, 1999; Reynders, 2012; Van Overschee & De Moor, 2012);
- Poly-reference least square frequency domain (pLSCF) (Peeters et al., 2004);

The multi-setup analyzes can be performed according the so-called post separate estimation re-scaling (PoSER) approach as well as with the so-called pre-global estimation re-scaling (PreGER) approach (Amador & Brincker, 2021; Brincker & Ventura, 2015; Döhler & Mevel, 2013; Rainieri & Fabbrocino, 2014). The calculation of the uncertainty bounds for the SSI family of algorithms follows the efficient implementation by Döhler and colleagues (Döhler, 2011; Döhler et al., 2013; Döhler & Mevel, 2013). The interested reader may refer to the extensive scientific literature on the subject for further information.

A few commercial software programs implements the algorithms mentioned above. The most well-known presumably are ARTeMIS (Solutions, 2001), by Structural Vibration Solutions, and MACEC, a Matlab toolbox for modal testing and OMA (Reynders et al., 2014). When it comes to open source modules the only ones available to the authors best knowledge are the first version of `pyOMA` (Pasca et al., 2022) and `Koma` (Kvåle, 2024), which is also an open-source Python library available on GitHub. It provides tools for OMA, focusing on simplicity and ease of use. `Koma` is designed to be a lightweight alternative to more general libraries like `pyOMA`, making it suitable for smaller projects.

The module's reliability and applicability for research purposes have been demonstrated by the authors through various studies (Alaggio et al., 2021; Aloisio et al., 2020; Simoncelli et al., 2023). Additionally, the module has gained traction within the research community, as evidenced by its use in studies by Saharan et al. (2023), Croce et al. (2023), Talebi et al. (2023), Abuodeh et al. (2023), and others.

Module's structure

The module is structured into three primary levels:

1. At the first level are the `setup` classes. Users instantiate these classes by providing a data array and the sampling frequency for a single setup scenario, or a list of data arrays and their respective sampling frequencies, and reference indices, for a multi-setup scenario.
2. The second level comprises the `algorithms` classes. Users can instantiate the algorithms they wish to run and then add them to the setup class.
3. The third level contains the `support` classes, which serve as auxiliary components to the first two levels. This level includes various specialized classes:
   - `result` classes, where outcomes are stored.
   - `geometry` classes, for storing geometric data.
   - `run_param` classes, where parameters used for running the algorithms are kept.
   - Dedicated classes for animating mode shapes and interacting with plots generated by the algorithm classes.

In addition to the levels depicted in the figure, there is a further level not shown, comprised of the set of functions internally called by the class methods. Many of these functions represent an updated version of those available in our previous release, `pyOMA`.

The Journal of Open Source Software

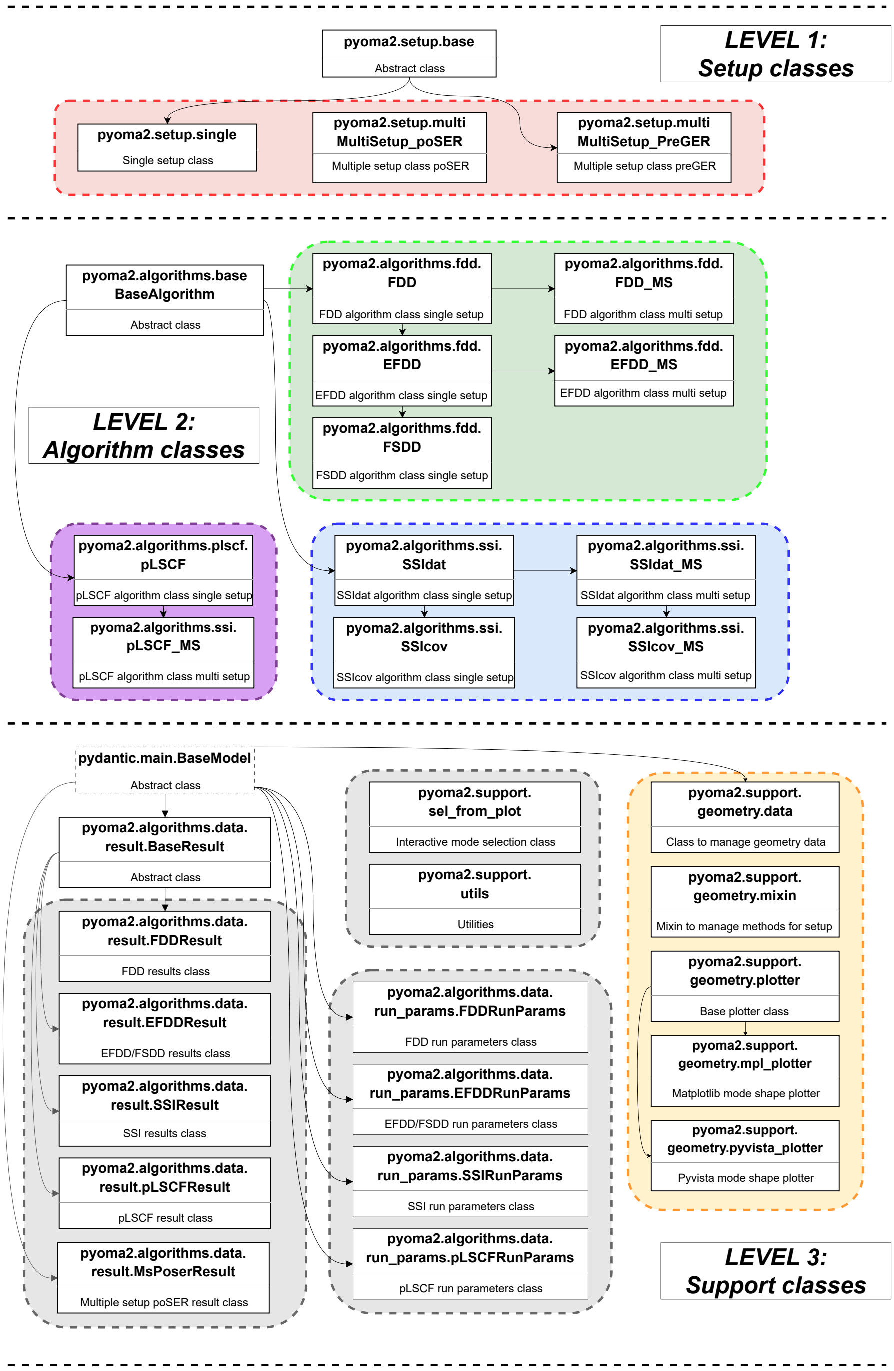

Figure 1: Schematic organisation of the module showing inheritance between classes

Documentation

A comprehensive documentation for pyOMA2, including examples, is available at https://pyoma.readthedocs.io/en/main/.

Acknowledgements

We acknowledge contributions from Angelo Aloisio and Marco Martino Rosso.

References

Abuodeh, O., Locke, W., Redmond, L., Sreenivasulu, R. V., & Schmid, M. (2023). Examining methods for modeling road surface roughness effects in vehicle–bridge interaction models via physical testing. *Society for Experimental Mechanics Annual Conference and Exposition*, 33–47. https://doi.org/10.1007/978-3-031-36663-5_5

Alaggio, R., Aloisio, A., Antonacci, E., & Cirella, R. (2021). Two-years static and dynamic monitoring of the santa maria di collemaggio basilica. *Construction and Building Materials*, *268*, 121069. https://doi.org/10.1016/j.conbuildmat.2020.121069

Aloisio, A., Pasca, D., Tomasi, R., & Fragiacomo, M. (2020). Dynamic identification and model updating of an eight-storey CLT building. *Engineering Structures*, *213*, 110593. https://doi.org/10.1016/j.engstruct.2020.110593

Amador, S. D., & Brincker, R. (2021). Robust multi-dataset identification with frequency domain decomposition. *Journal of Sound and Vibration*, *508*, 116207. https://doi.org/10.1016/j.jsv.2021.116207

Brincker, R., & Ventura, C. (2015). *Introduction to operational modal analysis*. John Wiley & Sons.

Brincker, R., Ventura, C. E., & Andersen, P. (2001). Damping estimation by frequency domain decomposition. *Proceedings of IMAC 19: A Conference on Structural Dynamics: Februar 5-8, 2001, Hyatt Orlando, Kissimmee, Florida, 2001*, 698–703.

Brincker, R., Zhang, L., & Andersen, P. (2001). Modal identification of output-only systems using frequency domain decomposition. *Smart Materials and Structures*, *10*(3), 441.

Croce, T., Girardi, M., Gurioli, G., Padovani, C., & Pellegrini, D. (2023). Towards a cloud-based platform for structural health monitoring: Implementation and numerical issues. *International Conference on Experimental Vibration Analysis for Civil Engineering Structures*, 610–619.

Döhler, M. (2011). *Subspace-based system identification and fault detection: Algorithms for large systems and application to structural vibration analysis* [PhD thesis]. Université Rennes 1.

Döhler, M., Lam, X.-B., & Mevel, L. (2013). Uncertainty quantification for modal parameters from stochastic subspace identification on multi-setup measurements. *Mechanical Systems and Signal Processing*, *36*(2), 562–581. https://doi.org/10.1016/j.ymssp.2012.11.011

Döhler, M., & Mevel, L. (2013). Efficient multi-order uncertainty computation for stochastic subspace identification. *Mechanical Systems and Signal Processing*, *38*(2), 346–366. https://doi.org/10.1016/j.ymssp.2013.01.012

Kvåle, K. A. (2024). *KOMA: Knut's operational modal analysis toolbox for python*. https://doi.org/10.5281/zenodo.14446286

Pasca, D. P., Aloisio, A., Rosso, M. M., & Sotiropoulos, S. (2022). PyOMA and PyOMA_GUI: A python module and software for operational modal analysis. *SoftwareX*, *20*, 101216. https://doi.org/10.1016/j.softx.2022.101216

Peeters, B., & De Roeck, G. (1999). Reference-based stochastic subspace identification for output-only modal analysis. *Mechanical Systems and Signal Processing*, *13*(6), 855–878. https://doi.org/10.1006/mssp.1999.1249

Peeters, B., Van der Auweraer, H., Guillaume, P., & Leuridan, J. (2004). The PolyMAX frequency-domain method: A new standard for modal parameter estimation? *Shock and Vibration*, *11*(3-4), 395–409. https://doi.org/10.1155/2004/523692

Rainieri, C., & Fabbrocino, G. (2014). Operational modal analysis of civil engineering structures. *Springer, New York*, *142*, 143. https://doi.org/10.1007/978-1-4939-0767-0

Reynders, E. (2012). System identification methods for (operational) modal analysis: Review and comparison. *Archives of Computational Methods in Engineering*, *19*, 51–124. https://doi.org/10.1007/s11831-012-9069-x

Reynders, E., Schevenels, M., & De Roeck, G. (2014). *MACEC 3.2: A matlab toolbox for experimental and operational modal analysis*. Department of Civil Engineering, KU Leuven. https://bwk.kuleuven.be/bwm/macec

Saharan, N., Kumar, P., & Pal, J. (2023). Convolutional neural network–based structural health monitoring framework for wind turbine blade. *Journal of Vibration and Control*, 10775463231213423. https://doi.org/10.1177/10775463231213423

Simoncelli, M., Aloisio, A., Zucca, M., Venturi, G., & Alaggio, R. (2023). Intensity and location of corrosion on the reliability of a steel bridge. *Journal of Constructional Steel Research*, *206*, 107937. https://doi.org/10.1016/j.jcsr.2023.107937

Solutions, S. V. (2001). *ARTeMIS extractor: Ambient response testing and modal identification software, user's manual*. Demark.

Talebi, A., Potenza, F., & Gattulli, V. (2023). Interoperability between BIM and FEM for vibration-based model updating of a pedestrian bridge. *Structures*, *53*, 1092–1107. https://doi.org/10.1016/j.istruc.2023.04.115

Van Overschee, P., & De Moor, B. (2012). *Subspace identification for linear systems: Theory—implementation—applications*. Springer Science & Business Media.

Zhang, L., Wang, T., & Tamura, Y. (2010). A frequency–spatial domain decomposition (FSDD) method for operational modal analysis. *Mechanical Systems and Signal Processing*, *24*(5), 1227–1239. https://doi.org/10.1016/j.ymssp.2009.10.024