The Journal of Open Source Software

auto-AUTO: A Python Layer for Automatically Running the AUTO-07p Continuation Software

Jonathan Demaeyer [1] **and Oisín Hamilton** [1,2]

1 Meteorological and Climatological Information Service, Royal Meteorological Institute of Belgium, Brussels, Belgium **2** UCLouvain — Earth and Life Institute, Belgium

DOI: 10.21105/joss.08079

Software

- Review
- Repository
- Archive

Editor: Daniel S. Katz

Reviewers:

- @epspebble
- @bartoldeman
- @lockwo

Submitted: 21 February 2025
Published: 24 September 2025

Summary

auto-AUTO (or AUTO^2) is a Python package that acts as an intermediate layer between the user and AUTO-07p continuation software (Doedel et al., 2007) (AUTO from here on). auto-AUTO automates the running of AUTO by monitoring the continuation, while also keeping track of bifurcation points. The package can automatically continue along these branching points, and continue branching along further bifurcation points to attempt to construct full bifurcation trees. To achieve this in a reliable way, AUTO computations are managed by auto-AUTO by setting appropriate stopping conditions, such as meeting other branches, looping branches, and specified bifurcations. These stopping conditions supplement the usual AUTO bounds and ensure that branches of detected solutions are unique and well defined.

In addition, auto-AUTO provides a comprehensive and documented Python API to investigate properties of the computed continuations and solutions. auto-AUTO can be run in Jupyter notebooks (Kluyver et al., 2016) and the results can be plotted with Matplotlib (Hunter, 2007) using predefined plotting methods (see Figure 1). This facilitates the integration of the results from AUTO with other Python workflows.

Statement of Need

AUTO is a highly optimised and tested continuation code base, and for this reason is one possible choice for use in bifurcation analysis studies when the number of variable is higher than 10. However, the use of AUTO in continuation analysis requires a steep learning curve as the existing Python integration is limited. In addition, there is limited support for automated tracking and continuation of branching points in AUTO. Therefore, when faced with problems of great complexity (typically when the dimensionality of the problem is greater than 10), the user must engage in a long and tedious analysis, restarting the computations many times, with the risk of loosing tracks of computation and missing key elements in the process. This package aims to solve this problem, by automating the running of AUTO, and by systematizing the investigation of branching points.

auto-AUTO is currently used by the authors to investigate the bifurcations and stability of a coupled land-atmosphere model (Hamilton et al., 2025) using the qgs model (Demaeyer et al., 2020). This is part of a wider research project on the concept of climate tipping points (Armstrong McKay et al., 2022; Ashwin et al., 2012; Lenton et al., 2008; Wunderling et al., 2021), and which typically involves bifurcation analysis in high-dimensional dynamical systems[1]. For this reason, robust but easy to use continuation software is of great importance in this analysis.

[1]Although conceptual models, with a relatively low number of degrees of freedom, are still used extensively in increasing our knowledge about tipping points (Wunderling et al., 2021).

The Journal of Open Source Software

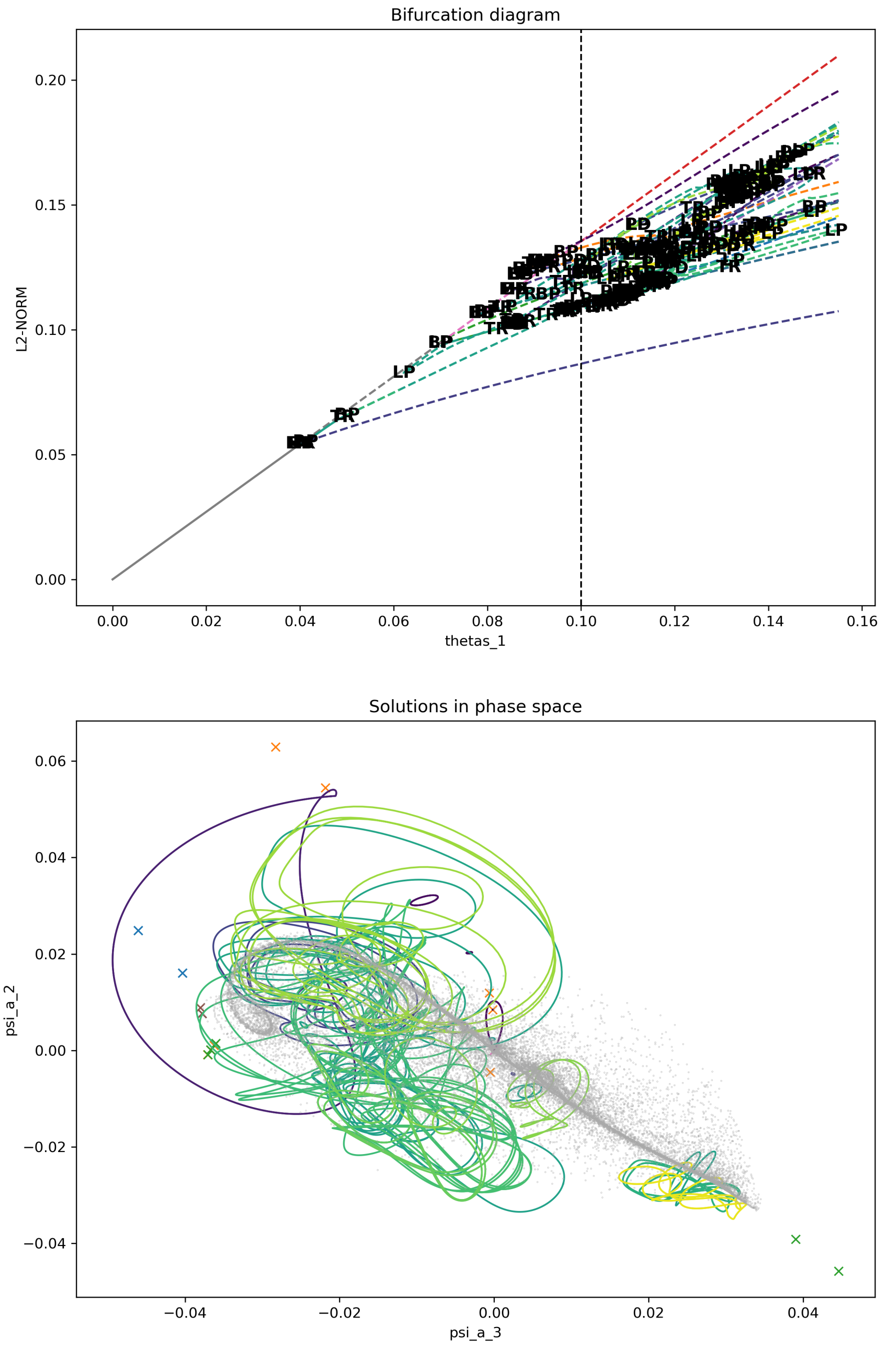

Figure 1: Example of plots using auto-AUTO functionalities to study a high-dimensional model (from the `RP1982` example notebook studying the Reinhold and Pierrehumbert 1982 model (Reinhold & Pierrehumbert, 1982)).

Existing Alternatives

Over time, and alongside the development of AUTO, many tools were introduced to help the users deal with the complexity of the output data stream of AUTO. We can mention for example XPPAUT (Ermentrout, 2012), which provides a frontend to AUTO, making it easy to continue the solutions to boundary value problems as well as equilibria, fixed points, and limit cycles (Ermentrout, 2007).

Another more recent development is PyCoBi, which "provides a Python interface to Auto-07p, allowing for a more intuitive usage of Auto-07p commands within Python scripts." (Gast, 2025)

Other bifurcation analysis software must also be mentioned here:

- BifurcationKit.jl (Veltz, 2020)
- PyDSTool (Clewley et al., 2007)
- Other useful tools can be found at https://dsweb.siam.org/Software

However, to the best of the authors' knowledge, these tools do not provide automatic continuation features relevant to the problems raised in the Statement of Need. PyCoBi proposes limited automatic generation for codimension-2 bifurcation diagrams (regime diagrams), a feature that is also planned in auto-AUTO future developments.

Future Developments

Future development plans for auto-AUTO include:

- Automatic continuation of regime diagrams, i.e., codimension-1 bifurcation branch tracking and detection of codimension-2 bifurcations by AUTO (regime diagrams)
- Investigation of the formalization and abstraction of automatic continuation using graph theory, leveraging the functionalities already available in auto-AUTO, and allowing for a more insightful construction of the diagrams

However, while the latter proposed development is an obvious way to pursue the development of auto-AUTO, it is presently out of the scope of the authors' research.

Acknowledgments

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska–Curie grant agreement no. 956170. In addition, funding has been provided through the "Fédération Wallonie-Bruxelles" with the instrument "Fonds Spéciaux de Recherche". The authors thank the three referees for their constructive review and their suggestions that helped improve the code and the manuscript.

References

Armstrong McKay, D. I., Staal, A., Abrams, J. F., Winkelmann, R., Sakschewski, B., Loriani, S., Fetzer, I., Cornell, S. E., Rockström, J., & Lenton, T. M. (2022). Exceeding 1.5°C global warming could trigger multiple climate tipping points. *Science*, *377*(6611), eabn7950. https://doi.org/10.1126/science.abn7950

Ashwin, P., Wieczorek, S., Vitolo, R., & Cox, P. (2012). Tipping points in open systems: Bifurcation, noise-induced and rate-dependent examples in the climate system. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, *370*(1962), 1166–1184. https://doi.org/10.1098/rsta.2011.0306

Clewley, R. H., Sherwood, W., LaMar, M., & Guckenheimer, J. (2007). *PyDSTool, a software environment for dynamical systems modeling*. http://pydstool.sourceforge.net

Demaeyer, J., Cruz, L. D., & Vannitsem, S. (2020). Qgs: A flexible Python framework of reduced-order multiscale climate models. *Journal of Open Source Software*, *5*(56), 2597. https://doi.org/10.21105/joss.02597

Doedel, E. J., Champneys, A. R., Dercole, F., Fairgrieve, T. F., Kuznetsov, Y. A., Oldeman, B., Paffenroth, R., Sandstede, B., Wang, X., & Zhang, C. (2007). *AUTO-07P: Continuation and bifurcation software for ordinary differential equations*.

Ermentrout, B. (2007). XPPAUT. *Scholarpedia*, *2*(1), 1399. https://doi.org/10.4249/scholarpedia.1399

Ermentrout, B. (2012). XPPAUT. In N. Le Novère (Ed.), *Computational systems neurobiology* (pp. 519–531). Springer Netherlands. https://doi.org/10.1007/978-94-007-3858-4_17

Gast, R. (2025). *Pyrates-neuroscience/PyCoBi: Dropped support for Python < 3.9* (Version v0.8.10). Zenodo. https://doi.org/10.5281/zenodo.15232900

Hamilton, O., Demaeyer, J., Crucifix, M., & Vannitsem, S. (2025). Using unstable periodic orbits to understand blocking behavior in a low order land–atmosphere model. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, *35*(8), 083126. https://doi.org/10.1063/5.0268852

Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. *Computing in Science & Engineering*, *9*(3), 90–95. https://doi.org/10.1109/MCSE.2007.55

Kluyver, T., Ragan-Kelley, B., Pérez, F., Granger, B., Bussonnier, M., Frederic, J., Kelley, K., Hamrick, J., Grout, J., Corlay, S., Ivanov, P., Avila, D., Abdalla, S., & Willing, C. (2016). *Jupyter notebooks – a publishing format for reproducible computational workflows* (F. Loizides & B. Schmidt, Eds.; pp. 87–90). IOS Press.

Lenton, T. M., Held, H., Kriegler, E., Hall, J. W., Lucht, W., Rahmstorf, S., & Schellnhuber, H. J. (2008). Tipping elements in the Earth's climate system. *Proceedings of the National Academy of Sciences*, *105*(6), 1786–1793. https://doi.org/10.1073/pnas.0705414105

Reinhold, B. B., & Pierrehumbert, R. T. (1982). Dynamics of weather regimes: Quasi-stationary waves and blocking. *Monthly Weather Review*, *110*(9), 1105–1145. https://doi.org/10.1175/1520-0493(1982)110%3C1105:DOWRQS%3E2.0.CO;2

Veltz, R. (2020). *BifurcationKit.jl*. Inria Sophia-Antipolis. https://hal.archives-ouvertes.fr/hal-02902346

Wunderling, N., Donges, J. F., Kurths, J., & Winkelmann, R. (2021). Interacting tipping elements increase risk of climate domino effects under global warming. *Earth System Dynamics*, *12*(2), 601–619. https://doi.org/10.5194/esd-12-601-2021