The Journal of Open Source Software

FAME-Io: Configuration tools for complex agent-based simulations

Felix Nitsch [1¶], **Christoph Schimeczek** [1], **Ulrich Frey** [1], **and Benjamin Fuchs** [1]

1 German Aerospace Center (DLR), Institute of Networked Energy Systems, Curiestr. 4, 70563 Stuttgart, Germany ¶ Corresponding author

DOI: 10.xxxxxx/draft

Software

- Review
- Repository
- Archive

Editor: Frauke Wiese

Reviewers:

- @lkstrp
- @sstroemer

Submitted: 01 August 2025
Published: unpublished

Summary

We present FAME-Io, a Python package designed to help users and creators of agent-based simulation models (ABM) better manage the preparation and processing of their input and output datasets. The package was built with the needs of researchers in mind. FAME-Io was specifically developed to interface with the open framework FAME[1] and is published under the open Apache-2.0 licence. The software offers various logging capabilities, shell-integrated help and documentation, as well as extensive pre-run integrity checks and helpful warning messages. It also allows individual data components to be easily extracted and used in secondary workflows. The code itself is operating system independent and follows best practices in software development. Test coverage, at the time of writing, is 92% and the project uses continuous integration and offers frequent releases.

FAME-Io is designed as one of two main components of FAME, each addressing a particular aspect and user group within ABM development, see Figure 1:

- FAME-Io is a Python package[2] for ABM users. It supports configuring complex simulations and managing of files associated with FAME.
- FAME-Core (Schimeczek, Deissenroth-Uhrig, et al., 2023) is a Java library for ABM modellers. It supports developing and executing ABM simulations.

These two modules interact with additional supporting components, like FAME-Mpi[3], FAME-Protobuf[4], FAME-Gui[5] (in beta release) and FAME-Prepare[6] (under development). The demonstration project FAME-Demo[7] provides a ready-made example for modellers to investigate and experiment with.

[1] https://gitlab.com/fame-framework/
[2] https://pypi.org/project/fameio/
[3] https://gitlab.com/fame-framework/mpi
[4] https://gitlab.com/fame-framework/fame-protobuf
[5] https://gitlab.com/fame-framework/fame-gui
[6] https://gitlab.com/fame-framework/fame-prepare
[7] https://gitlab.com/fame-framework/fame-demo

The Journal of Open Source Software

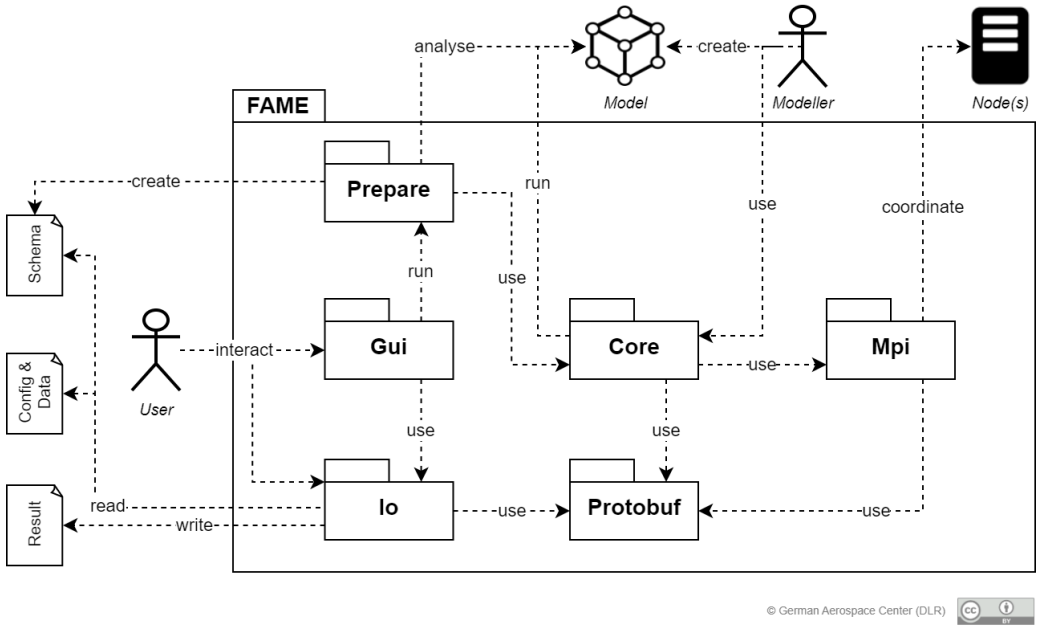

Figure 1: Components of FAME and their interactions.

Configuration of FAME model inputs

Defining agents and their interactions is the first central task when creating any ABM simulation. IN FAME, such agent interactions need not be hard-coded, but can be defined when configuring the model. FAME-Io enables researchers to quickly create and handle configurations of simulations, even complex ones with thousands of agents and numerous interactions. To this end, FAME-Io allows splitting configurations into multiple files, each dedicated to a specific aspect of a configuration. An overarching configuration comprises one or more structured YAML files. Large data structures, like time series, can be imported from external (CSV) files. FAME-Io supports easy definition of multiple similar agent interactions, i.e., "Many-to-One" , "One-to-Many" and "Many-to-Many" messages, significantly reducing repetitive parameterization tasks.

Once the simulation configuration files are created, they are validated by FAME-Io against a schema derived from the associated FAME model. This validation is comprehensive and checks the presence of each agent's mandatory parameters. It also tests the data type of each parameter and possible value restrictions. Furthermore, FAME-Io evaluates the plausibility of agent interactions, i.e., the correct type of data sent for each agent interaction. If errors are detected, FAME-Io issues meaningful messages that help researchers to fix configuration problems even before the actual simulation runs. In case no configuration errors are identified, FAME-Io creates a single binary protobuf file that serves as input for the associated ABM (executed with FAME-Core).

Conversion of FAME model outputs

Once a model is executed with FAME-Core, its entire output is stored in a single binary protobuf file. FAME-Io converts this output into human-readable files with a standardized output structure in CSV format. Researchers can customize this configuration process to suit their needs. FAME-Io allows limiting the conversion to subsets of agents, it offers different output formats, and the results of similar agents can be split or grouped. These options facilitate later individual post-processing of the results. FAME-Io can also handle very large output files, e.g., from simulations with thousands of agents. For computers with limited resources, it offers a mode with a low-memory profile.

The Journal of Open Source Software

Statement of need

The energy system transition towards a sustainable future is one of the most urgent concerns of our time. Hence, the field of energy systems analysis deals with possible paths to a more sustainable future. However, the increasing complexity of energy markets combined with different policies and heterogeneous actors poses ever new challenges to this field (Pfenninger et al., 2014; Pye et al., 2021). ABM is perhaps the most suitable approach to tackle these challenges (Klein et al., 2019), because

- the actors' perspective and their specific characteristics are taken into account (Frey et al., 2020; Kraan et al., 2018), and
- models remain computationally feasible even in large-scale simulations (Hansen et al., 2019).

FAME is a new software suite that helps researchers develop and use ABM simulations. FAME-models are highly configurable and allow the modelling workflow to be changed without affecting the model itself. This high flexibility of models, however, leads to the need for a sophisticated model configuration tool - FAME-Io addresses this need by providing powerful tools for complex ABM. It provides automatic translation between Java-coded agent specifications and highly organised configuration files. FAME-Io grants a convenient way to interact with FAME-based models, i.e. to input data and access their results. It also helps model users handling complex simulation configurations with hundreds of repeated interactions between agents.

State of the field

A scholarly search returns mostly review articles, e.g., by Ringkjøb et al. (2018). However, configuration support or input-output management is not an issue in any of them. Hence, we compare the two most popular ABM frameworks, NetLogo (Wilensky, 1999), which is widely used in teaching, and Repast (Collier & North, 2013), which was recently ported to Python. Unlike FAME-Io, both frameworks lack a dedicated input-output management and a configuration and validation tool. Help with parameterizing a NetLogo model is only provided by BehaviorSpace[8]. Given this lack of tools, we conclude that the need for a dedicated software is there and FAME-Io can fill this need.

Performance

FAME-Io offers high performance. It reads, validates, and converts configurations with approximately 25 agents, 150 contracts and one megabyte of time series data in less than five seconds on a desktop PC. However, for very large configurations, FAME-Io can run on multiple computing cores for further speed increase. The YAML and CSV file types used by FAME-Io are easily readable by both humans and machines. Thus, the configuration, execution, and result evaluation of FAME-based models can be easily integrated into automated workflows via FAME-Io. This paves the way for the use of FAME models as part of computationally intensive approaches, such as extensive parameter sweeps or genetic algorithms.

Applications

FAME was designed for energy systems modelling, but is not limited to this field. Thus, it can also be used in other domains for ABM simulations with similarly structured problems. FAME-Io is an integral part of FAME and currently the only tool available for configuring FAME simulations and accessing their results.

A large and powerful FAME-based model is AMIRIS (Schimeczek, Nienhaus, et al., 2023)[9], an agent-based market model for the investigation of renewable and integrated energy systems.

[8] https://ccl.northwestern.edu/netlogo/docs/behaviorspace.html
[9] https://dlr-ve.gitlab.io/esy/amiris/home/

AMIRIS is also open-source and is being used in current projects in the field of energy systems analysis. Specifically, FAME-Io has been used to integrate AMIRIS into multi-model workflows with automatic model configuration and result evaluation in the following projects: TradeRES[10], ERAFlex II[11], UNSEEN[12], and VERMEER[13]. Hence, it has already been applied for various scientific publications (Frey et al., 2020; Nitsch et al., 2021; Safarazi et al., 2020).

Funding

This project was funded by the German Aerospace Center (DLR).

Acknowledgments

The authors would like to thank Aurélien Regat-Barrel, who worked on the interface for FAME-Gui, for his advice and comments. Furthermore, the authors thank Manuel Bostanci for implementing parallelisation of FAME-Io configuration tasks.

References

Collier, N., & North, M. (2013). Parallel agent-based simulation with repast for high performance computing. *SIMULATION*, *89*(10), 1215–1235. https://doi.org/10.1177/0037549712462620

Frey, U. J., Klein, M., Nienhaus, K., & Schimeczek, C. (2020). Self-reinforcing electricity price dynamics under the variable market premium scheme. *Energies*, *13*(20), 5350. https://doi.org/10.3390/en13205350

Hansen, P., Liu, X., & Morrison, G. M. (2019). Agent-based modelling and socio-technical energy transitions: A systematic literature review. *Energy Research & Social Science*, *49*, 41–52. https://doi.org/10.1016/j.erss.2018.10.021

Klein, M., Frey, U. J., & Reeg, M. (2019). Models within models: Agent-based modelling and simulation in energy systems analysis. *Journal of Artificial Societies and Social Simulation*, *22*(4), 6. https://doi.org/10.18564/jasss.4129

Kraan, O., Kramer, G. J., & Nikolic, I. (2018). Investment in the future electricity system — an agent-based modelling approach. *Energy*, *151*, 569–580. https://doi.org/10.1016/j.energy.2018.03.092

Nitsch, F., Schimeczek, C., & Wehrle, S. (2021). *Back-testing the agent-based model AMIRIS for the Austrian day-ahead electricity market*. Zenodo. https://doi.org/10.5281/zenodo.5726738

Pfenninger, S., Hawkes, A., & Keirstead, J. (2014). Energy systems modeling for twenty-first century energy challenges. *Renewable and Sustainable Energy Reviews*, *33*, 74–86. https://doi.org/10.1016/j.rser.2014.02.003

Pye, S., Broad, O., Bataille, C., Brockway, P., Daly, H., Freeman, R., Gambhir, A., Geden, O., Rogan, F., Sanghvi, S., & others. (2021). Modelling net-zero emissions energy systems requires a change in approach. *Climate Policy*, *21*(2), 222–231. https://doi.org/10.1080/14693062.2020.1824891

Ringkjøb, H.-K., Haugan, P. M., & Solbrekke, I. M. (2018). A review of modelling tools for energy and electricity systems with large shares of variable renewables. *Renewable and*

[10] https://traderes.eu/
[11] https://www.enargus.de/detail/?id=2001065
[12] https://www.dlr.de/ve/en/desktopdefault.aspx/tabid-15957/25887_read-66481/
[13] https://www.dlr.de/ve/en/desktopdefault.aspx/tabid-15955/25885_read-66475/

Sustainable Energy Reviews, *96*, 440–459. https://doi.org/10.1016/j.rser.2018.08.002

Safarazi, S., Deissenroth-Uhrig, M., & Bertsch, V. (2020). Assessing the impacts of community energy storage systems on the German electricity market: An agent-based analysis. *2020 17th International Conference on the European Energy Market (EEM)*, 1–6. https://doi.org/10.1109/EEM49802.2020.9221924

Schimeczek, C., Deissenroth-Uhrig, M., Frey, U., Fuchs, B., El Ghazi, A. A., Wetzel, M., & Nienhaus, K. (2023). FAME-Core: An open framework for distributed agent-based modelling of energy systems. *Journal of Open Source Software*. https://doi.org/10.21105/joss.05087

Schimeczek, C., Nienhaus, K., Frey, U., Sperber, E., Sarfarazi, S., Nitsch, F., Kochems, J., & El Ghazi, A. A. (2023). AMIRIS: Agent-based market model for the investigation of renewable and integrated energy systems. *Journal of Open Source Software*. https://doi.org/10.21105/joss.05041

Wilensky, U. (1999). *NetLogo (and NetLogo user manual)*. http://ccl.northwestern.edu/netlogo/