The Journal of Open Source Software

ExerPy: An open-source framework for automated exergy analysis

Sergio Tomasinelli [1¶], **Francesco Witte** [2], **Robert Müller** [1], and **Fontina Petrakopoulou** [1]

1 Chair of Energy Engineering and Climate Protection, Technische Universität Berlin, Germany **2** Institute of Networked Energy Systems, German Aerospace Center (DLR), Germany ¶ Corresponding author

DOI: 10.xxxxxx/draft

Software

- Review
- Repository
- Archive

Editor:

Submitted: 24 July 2025
Published: unpublished

Summary

ExerPy is an open-source Python framework that automates the exergy analysis of thermodynamic systems. It integrates with the process-simulation software Aspen Plus®, Ebsilon®Professional, and TESPy, through a unified JSON interface. ExerPy automatically identifies components and defines component- and system-level exergy balances. It also provides key performance metrics, including exergy destruction and exergetic efficiency. By accounting for both physical and chemical exergy and minimizing the need for manual calculations, ExerPy enables consistent second-law assessments across a wide range of systems, from simple thermodynamic cycles to complex plants. ExerPy facilitates the integration of analysis across different tools and supports subsequent data processing through standardized outputs.

Statement of need

Exergy analysis is an effective tool for assessing the quality of energy and capability to generate useful work. It facilitates the identification of thermodynamic irreversibilities within a system, thereby offering a more comprehensive understanding of energy conversion processes. The quantification of exergy destruction enables researchers and engineers to develop strategies that enhance efficiency, reduce costs, and promote sustainable conversion technologies (Meyer et al., 2009; Petrakopoulou et al., 2017; Tsatsaronis, 1993).

Despite its advantages, exergy analysis has not yet been widely integrated into most commercial software used for thermodynamic assessments, which primarly focus on energy and mass balance calculations. The calculation of both physical and chemical exergy of material streams, as well as an automated evaluation of the overall process, was seamlessly integrated into the open-source software TESPy (Hofmann et al., 2022; Witte et al., 2022; Witte & Tuschy, 2020). While this represented an important step toward facilitating the application of exergy analysis, current exergy analysis efforts still rely heavily on user input, are prone to incorrect interpretation of component balances, and lack interoperability with other open source or commercial tools. These shortcomings have driven the demand for specialized, user-friendly, automated open-source software that enables exergy-based analyses and interoperates with both commercial and open-source environments.

To address these needs, ExerPy provides a Python-based solution that automates exergy analysis of energy-conversion systems via a JSON data interface. The tool includes an API that automatically connects to different process-simulation environments, autonomously identifies components and assigns exergy balances, enabling detailed and accurate exergy analysis across the entire process. This level of automation streamlines the workflow, improving efficiency and accuracy in applying exergy analysis, and thereby supports the optimization of energy-conversion systems from an exergy perspective.

Software design

ExerPy has been developed as an exergy-analysis tool that is independent of any specific process-simulation environment. A central architectural decision was to separate data acquisition and normalization from the exergy-calculation core. Simulation-specific adapters extract stream and component information from Aspen Plus®, Ebsilon®Professional, or TESPy and convert it into a standardized JSON representation. The analysis module operates exclusively on this uniform component and connection based schema, which enables consistent component- and system-level balances, standardized outputs, and straightforward extension to additional analysis types (e.g., exergoeconomic analysis) without re-implementing simulation interfaces. Although TESPy had already incorporated an automated exergy-analysis tool (Hofmann et al., 2022; Witte et al., 2022; Witte & Tuschy, 2020), its integration within a simulator-centric codebase strongly couples analysis logic to a specific data model and limits interoperability. The extension of TESPy to effectively process simulation results from commercial tools would lead to an increase in complexity, dependencies, and maintenance requirements for users primarily interested in simulation. Therefore, a dedicated package was required to provide a single, maintainable, open-source framework that performs automated exergy analysis across both commercial and open-source tools, including user-supplied system data via JSON.

Features

ExerPy is divided into two main modules: the data-processing module, which manages the extraction and preparation of simulation data, and the exergy-analysis module, which conducts the detailed exergy calculations. The initial implementation supports Ebsilon®Professional, Aspen Plus®, and TESPy. The architecture is outlined in the following sections and is shown in Figure 1.

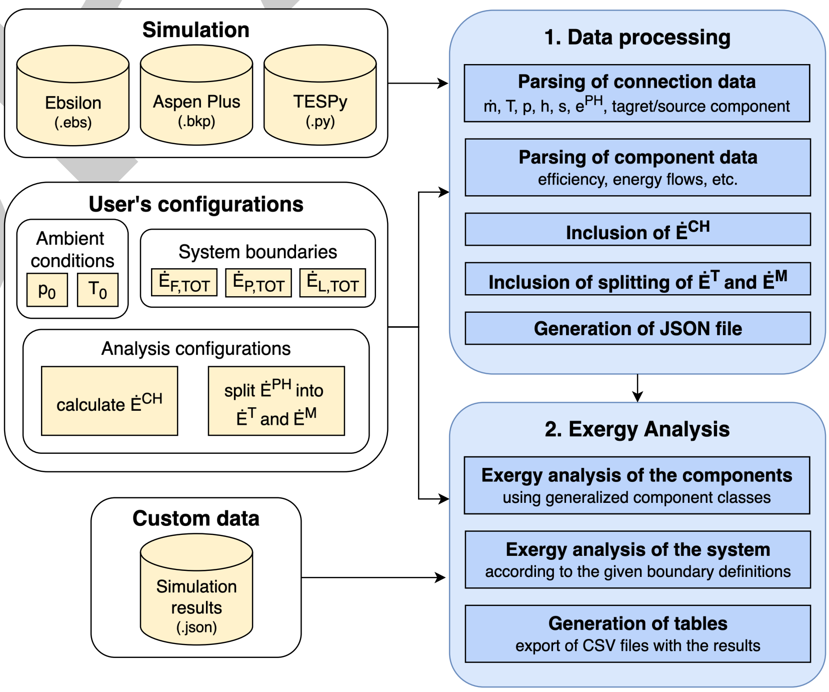

Figure 1: Structure of the ExerPy framework.

Data processing

A workflow begins with the parsing of simulation data from models created in Ebsilon®Professional, Aspen Plus®, or TESPy, using the respective functions: `from_ebsilon`, `from_aspen`, and `from_tespy`. It is important to note that the physical exergy, calculated from the entropy and enthalpy of the streams, is obtained directly from the simulation tools. Independent of the simulation software, users also have the option to supply their own JSON file using `from_json`, which must conform to the required format for exergy analysis.

During the parsing process, connection data such as mass flow rate (m), temperature (T), pressure (p), enthalpy (h), entropy (s), and physical exergy (e^{PH}) are parsed, along with the identification of target and source components. In addition, component data including efficiency, energy flows, and other relevant thermodynamic properties are also extracted. Ambient conditions can be taken directly from the simulation, or specified manually by the user.

ExerPy also allows the splitting of physical exergy into thermal (e^{T}) and mechanical (e^{M}) parts. This separation enables a more comprehensive analysis of thermodynamic processes, especially for components operating below ambient temperature (Morosuk & Tsatsaronis, 2019). These values are calculated using the native property functions of the simulation tools. In the initial release of ExerPy, this separation is not yet supported in Aspen Plus® due to limited access to thermodynamic functions, but it is planned for a future update.

After data parsing is complete, chemical exergy is calculated from stream composition following TESPy's approach (Hofmann et al., 2022) and the reference environment developed by Ahrendts (Ahrendts, 1977). For pure substances, values are obtained from tabulated data in the selected thermodynamic model. For mixtures (e.g., air, flue gas), ExerPy computes specific chemical exergy from constituent molar fractions using standard assumptions (ideal behavior for gas mixtures, with adjustments for condensables such as water). Finally, all parsed and calculated data are consolidated into a standardized JSON file, independent of the simulation tool used, including all the necessary information for a comprehensive exergy analysis.

Exergy Analysis

The framework performs exergy analysis at both the component and system levels. Each component of the system—such as turbines, compressors, and heat exchangers—is represented by a Python class that automatically assigns the exergy of the fuel and the exergy of the product of the component. Using these definitions, ExerPy calculates relevant metrics for each component (i.e., the exergy destruction and exergetic efficiency). Thermal energy losses of components are included in their exergy destruction, and streams discharged to the environment are treated as exergy losses of the overall system. This approach provides coherent calculations of the inefficiencies of individual component and supports targeted optimization.

At the system level, the total exergy balance is established by evaluating the exergy of streams crossing the system boundaries. To perform this analysis, and in the current release of the tool, it is necessary for the user to specify the product, the fuel, and the exergy loss of the overall process. The system-level exergy analysis yields the overall exergetic efficiency and the total exergy destruction of the overall system. Finally, the framework allows the results to be exported as CSV files for further examination and integration into additional workflows.

Validation

Validation has been conducted based on three different case studies documented in the online documentation of the framework (Tomasinelli et al., 2025). The results of the exergy analysis of a combined cycle power plant simulated with Aspen Plus® and with TESPy show a maximum difference of 1% compared to the simulation results from Ebsilon®Professional, validating the accuracy and confirming the flexibility of the tool. Additional applications and validation of

ExerPy, e.g., the CGAM process (Valero et al., 1994) and a heat pump, are also available in the documentation and on the GitHub repository.

Research impact statement

The demand for automated, publicly accessible exergy-analysis workflows is evidenced by the prior adoption of TESPy's exergy-analysis functionality in peer-reviewed educational (Hofmann et al., 2023) and applied studies (Barandier et al., 2023; Fritz et al., 2024; Hofmann et al., 2024). In addition to this, the near-term significance of automation has been emphasized in perspective work on the future of exergy-based methods, explicitly discussing the relevance of streamlined and automated implementations (Tsatsaronis, 2024).

ExerPy translates this demonstrated demand into a community-ready software by offering a dedicated, versioned, and openly developed package within the oemof ecosystem. Community-readiness signals include an OSI-approved license, public source repository, automated test infrastructure, and user documentation with working examples and contribution guidance, supporting reproducible analyses and facilitating external adoption.

AI usage disclosure

Generative AI tools (multiple ChatGPT models, Claude Code, and DeepL Write) were used to generate and modify portions of the codebase and to support language editing of this manuscript. All AI-assisted changes were reviewed and validated by the authors.

Acknowledgements

Parts of this work were funded by the German Federal Ministry for Economic Affairs and Climate Action through the research project SecöndLife, grant number 03EI1076A.

References

Ahrendts, J. (1977). *Die Exergie chemisch reaktionsfähiger Systeme.*

Barandier, P., Miranda, A., & Cardoso, A. J. M. (2023). Exergy analysis of aerothermal and geothermal heat pumps in a critical environment application. *6th European International Conference on Industrial Engineering and Operations Management*, 490–501.

Fritz, M., Freißmann, J., & Tuschy, I. (2024). *Open-source web dashboard zur simulation, analyse und bewertung von wärmepumpen.* https://doi.org/10.34712/142.68

Hofmann, M., Freißmann, J., Fritz, M., Alexe, J. H., Witte, F., & Tsatsaronis, G. (2024). Exergy-based methods for heat pumps. *Proceedings of the 7th Internation Seminar on ORC Power Systems*, 553–562.

Hofmann, M., Witte, F., Fritz, M., Freißmann, J., Tuschy, I., & Tsatsaronis, G. (2023). Free and open-source teaching: Understanding exergy using thermal engineering systems in python (TESPy). *36th International Conference on Efficiency, Cost, Optimization, Simulation and Environmental Impact of Energy Systems, ECOS 2023*, 195–209.

Hofmann, M., Witte, F., Shawky, K., Tuschy, I., & Tsatsaronis, G. (2022). Thermal engineering systems in python (TESPy): The implementation and validation of the chemical exergy [Conference paper]. In *Proceedings of ECOS 2022 - 35th International Conference on Efficiency, Cost, Optimization, Simulation and Environmental Impact of Energy Systems* (pp. 257–269). https://doi.org/10.11581/dtu.00000267

Meyer, L., Tsatsaronis, G., Buchgeister, J., & Schebek, L. (2009). Exergoenvironmental analysis for evaluation of the environmental impact of energy conversion systems. *Energy*, *34*(1), 75–89. https://doi.org/10.1016/j.energy.2008.07.018

Morosuk, T., & Tsatsaronis, G. (2019). Splitting physical exergy: Theory and application. *Energy*, *167*, 698–707. https://doi.org/10.1016/j.energy.2018.10.090

Petrakopoulou, F., Sánchez-Delgado, S., Marugán-Cruz, C., & Santana, D. (2017). Improving the efficiency of gas turbine systems with volumetric solar receivers. *Energy Conversion and Management*, *149*, 579–592. https://doi.org/10.1016/j.enconman.2017.07.058

Tomasinelli, S., Witte, F., Müller, R., & Petrakopoulou, F. (2025). *ExerPy: Exergy Analysis in Python*. https://exerpy.readthedocs.io

Tsatsaronis, G. (1993). Thermoeconomic analysis and optimization of energy systems. *Progress in Energy and Combustion Science*, *19*(3), 227–257. https://doi.org/10.1016/0360-1285(93)90016-8

Tsatsaronis, G. (2024). The future of exergy-based methods. *Energy*, *302*, 131881.

Valero, A., Lozano, M. A., Serra, L., Tsatsaronis, G., Pisa, J., Frangopoulos, C., & Spakovsky, M. R. von. (1994). CGAM problem: Definition and conventional solution. *Energy*, *19*(3), 279–286. https://doi.org/10.1016/0360-5442(94)90112-0

Witte, F., Hofmann, M., Meier, J., Tuschy, I., & Tsatsaronis, G. (2022). Generic and open-source exergy analysis—extending the simulation framework TESPy. *Energies*, *15*(11), 4087. https://doi.org/10.3390/en15114087

Witte, F., & Tuschy, I. (2020). TESPy: Thermal Engineering Systems in Python. *Journal of Open Source Software*, *5*, 2178. https://doi.org/10.21105/joss.02178