Further Application of Modelica-Based Nuclear Power System Simulation: A Review of Home-Tasks in Different Scenarios Driven by Model and Data

Ao Zhang [1] Xiang Wang [1]

[1] College of Nuclear Science and Technology, Harbin Engineering University, China, {zhangao, xiang.wang}@ hrbeu.edu.cn

Abstract

System simulation is inherently an applied technology, driven by specific application demands. The Modelica language enables equation-based, system-level, multi-domain, and visual modeling, facilitating researchers conducting system simulation studies of nuclear power systems. This paper introduces the custom-built AESE library on the OpenModelica platform. Examples in AESE are provided to illustrate the development and simulation of system models for conventional pressurized water reactors (PWRs) and high-temperature gas-cooled reactors (HTGRs). The paper also describes tasks completed under different application scenarios, including hardware-in-the-loop simulation, multi-objective optimization, system identification, and rapid optimization, using model-driven and data-driven approaches with the AESE library. The further application of simulation models and data has significant practical and engineering value. This paper serves as a valuable reference for the intelligent application of energy system models in the context of advanced engineering challenges.

Keywords: Modelica, Nuclear Power System, Model-driven, Data-driven

1 Introduction

System-level simulation provides a framework for understanding and analyzing the behavior of complex systems, enabling enhanced design and optimization. Multi-domain system-level modeling has gained significant popularity in seamlessly integrating physical models from various domains using a unified modeling language. This approach helps to overcome interface issues between different domain-specific software and enables the modeling of comprehensive and complex real-world systems (Mattsson et al. 1998).

Modelica, as an example of a multi-domain system modeling language, employs mathematical equations to uniformly describe physical processes across different domains. It offers visual modeling capabilities to represent the topology of a system and enables the solution of steady-state and transient simulations by solving algebraic and differential equations. The Modelica language has been widely used in the aerospace, shipbuilding, automotive, energy, and other fields (Briese et al. 2020; Soriano et al. 2016; Andreasson et al. 2013).

Nuclear power systems (NPSs) are inherently complex and face increasing demands for shorter development cycles, higher safety standards, and smarter operation. While traditional offline Modelica-based simulations have successfully supported reactor design and thermal-hydraulic analysis (Casella and Leva 2005; Cammi et al. 2011), the full potential of system simulation extends beyond this. Driven by the need for more intelligent system management and accelerated development, system simulation is evolving to support multiple practical scenarios that leverage both models and data. This work specifically focuses on the expanded application of Modelica-based NPS simulation across diverse technical dimensions:

a) Hardware-in-the-loop simulation (HILS), which enables integrated control verification by coupling physical hardware with digital models. For complex NPSs, the pressure to shorten development cycles and meet stringent safety requirements necessitates comprehensive control system testing to avoid costly failures. These demands have driven recent advancements in HILS (Mihalič et al., 2022). Although Modelica is widely used in offline NPS simulations, its application in system-level HILS research for nuclear systems remains limited.

b) Multi-objective optimization (MOO), which facilitates system-level design exploration by resolving trade-offs among competing objectives. In large-scale systems like NPSs, single-objective optimization often fails to represent the comprehensive requirements of engineering applications (Zhang and Wang 2024). Therefore, greater attention is given to multi-objective optimization problems that incorporate complex constraints. These objectives are often interdependent and cannot be optimized simultaneously, requiring coordinated strategies to identify Pareto-optimal solutions under varying conditions.

c) System identification, which leverages operational data for rapid prediction and data-driven modeling, supporting fast evaluation and intelligent optimization. A significant amount of data generated during nuclear power operation remains underutilized. However, emerging data-driven multi-objective (DDMO) optimization frameworks aim to bridge this gap. By coupling system identification algorithms with MOO, these frameworks enable efficient exploration of design spaces, enhance productivity, and improve predictive capabilities for future operational states.

These application scenarios underscore the diverse and critical roles that system simulation can play in modern NPS development. As depicted schematically in Figure 1, they encompass physical integration, algorithmic coordination, and data utilization, collectively forming a comprehensive modeling–application framework. This paper builds upon this framework by first introducing the general modeling approach using the AESE library (Section 2), then presenting these three representative applications in detail (Section 3), and finally concluding with a summary and outlook (Section 4).

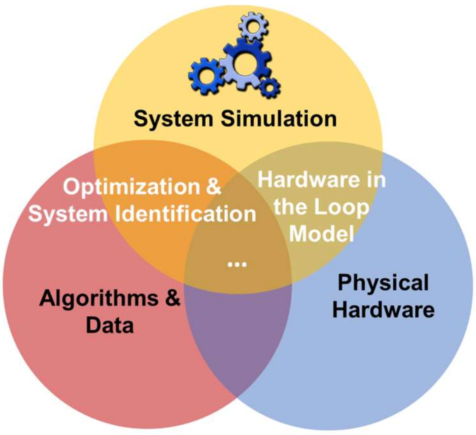


Figure 1. The further application directions of system simulation discussed in this paper

2 System Simulation

2.1 Framework

A custom-built Advanced Energy Systems Engineering (AESE) library (Zhang and Wang 2024) in OpenModelica (Fritzson et al. 2020) was developed in this study to describe various advanced energy systems. Figure 2 depicts the AESE architecture and the hierarchical structure of its He-Xe CBC system sub-library. Designed for modularity and openness, the library separates component models from working fluid models, allowing easy adaptation to future fluid models. Modelica facilitates this by supporting a high-level component library framework, defining standardized interfaces, and using inheritance or substitution for detailed component implementations. The following briefly outlines the modeling and simulation of two typical nuclear energy systems: the pressurized water reactor (PWR) and the high-temperature gas-cooled reactor (HTGR).

2.2 Pressurized Water Reactor (PWR)

The PWR-NPS consists of a nuclear reactor and two energy conversion loops (Figure 3). This study created a direct propulsion PWR-NPS model using Modelica, based on the "NS Mutsu" nuclear merchant ship (Peng 2009). The model, focused on a pressurized water reactor, describes fluid flow, heat transfer, phase changes, and power output. It was visually built (Figure 4) using adapted components from the ThermoPower (Casella and Leva 2024) and AESE libraries, including a reactor with a point kinetics model, a steam generator (SG) with drum and one-dimensional flow parts, condensers, turbines, pumps, and control units. Steady-state tests (Table 1) and dynamic analysis (Figures 5-7) verified the model's accuracy (Zhang et al. 2024). These results demonstrate that the Modelica-based model is capable of simulating and validating both steady-state and dynamic operations of conventional PWRs in the nuclear field.

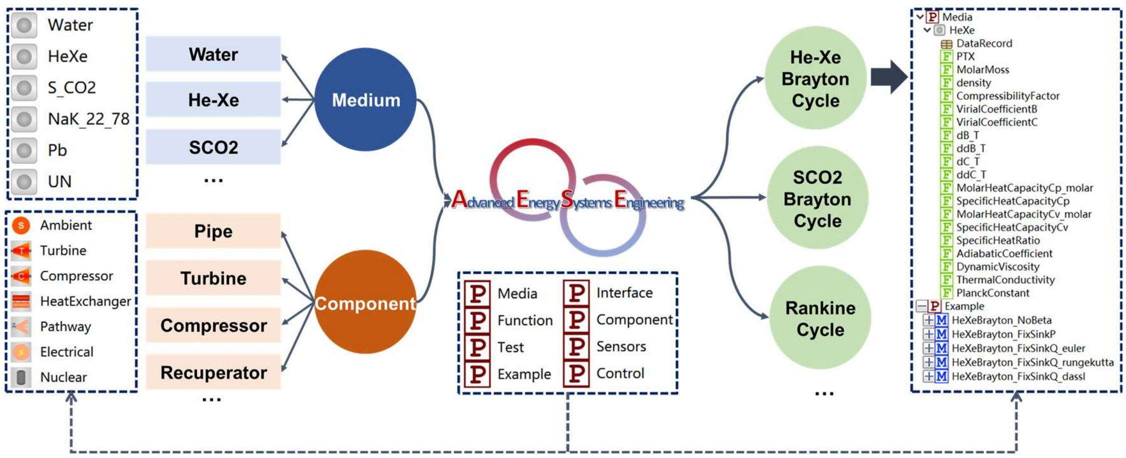


Figure 2. Schematic of the advanced energy systems engineering (AESE) library (Zhang and Wang 2024).

DOI 10.3384/ecp218373

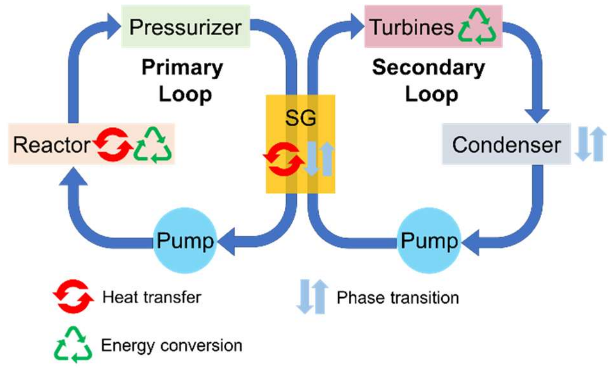


Figure 3. Structure of PWR-NPS (Zhang et al. 2024).

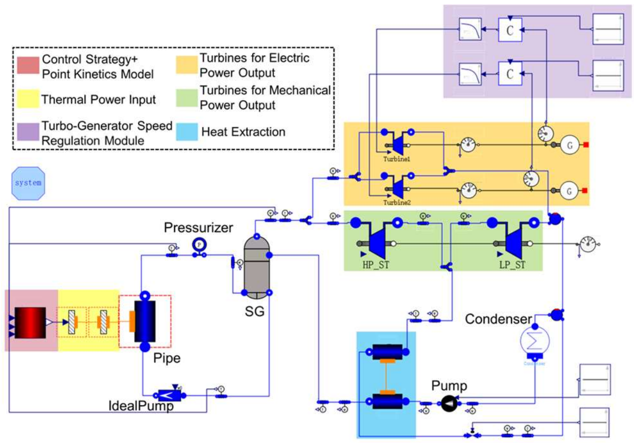


Figure 4. Simulation model of the PWR-NPS (Zhang et al. 2024).

Table 1. Steady-state simulation results.

Parameter	Reference	Result	Error
Reactor thermal power (MW)	36	35.81	0.52%
Temperature of SG primary side inlet (K)	558.15	555.18	0.53%
Temperature of SG primary side outlet (K)	544.15	544.82	0.12%
Pressure of SG secondary side (MPa)	3.9	3.72	4.61%
Feedwater temperature of SG secondary side (K)	433.15	431.28	0.43%
Steam temperature of SG secondary side (K)	524.15	519.19	0.95%
Steam output of SG (kg/s)	17	17.31	1.82%
Mechanical power output of steam turbine (MW)	7.35	7.38	0.41%

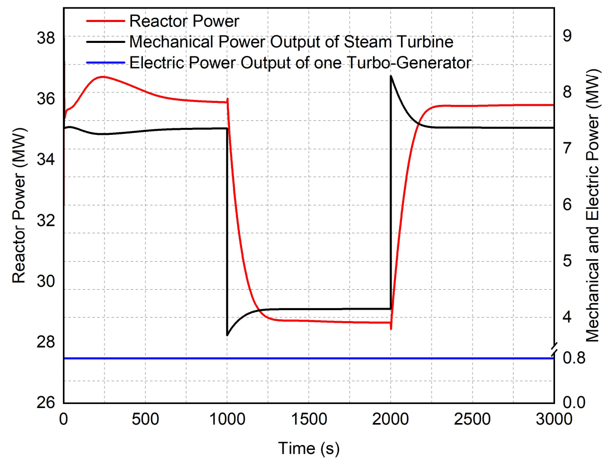


Figure 5. Transient simulation results of the nuclear power system (Zhang et al. 2024).

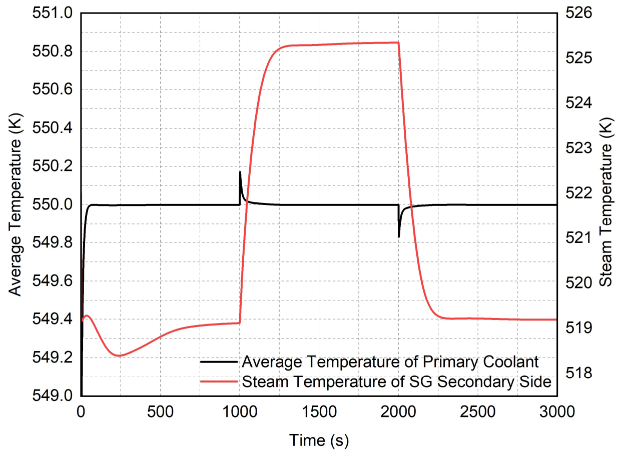


Figure 6. Transient heat transfer process on the secondary side of the SG (Zhang et al. 2024).

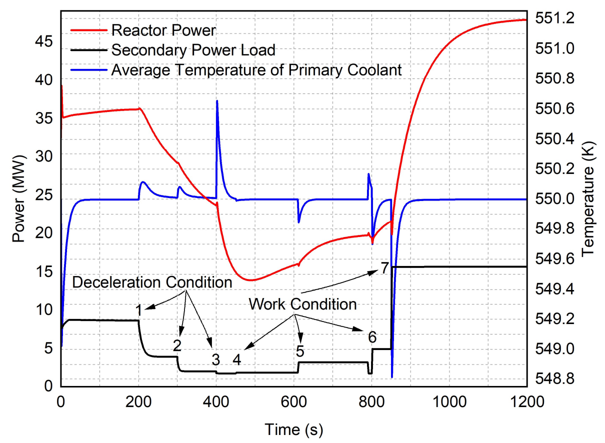
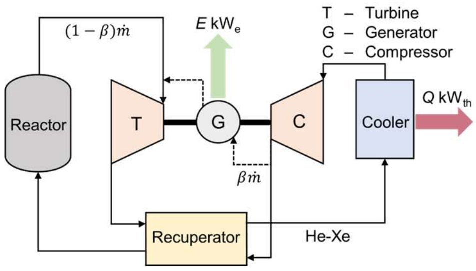


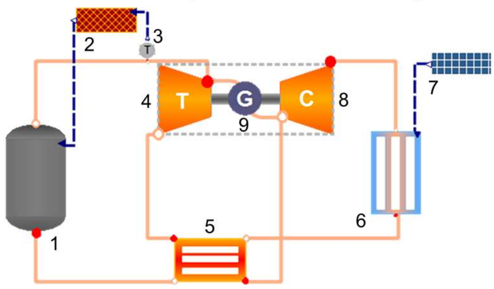
Figure 7. Transient simulation results of the NPS (Zhang et al. 2024).

2.3 High-Temperature Gas-Cooled Reactor (HTGR)

The HTGR enables efficient coupling with the Brayton cycle via its high outlet temperature, while Brayton cycle research focuses on system configuration and gas property effects. The AESE library also includes a He-Xe property library and supports closed-Brayton-cycle (CBC) systems (Figure 8a), offering new tools for simulating He-Xe CBC systems (Figure 8b). The presence of Xe in He causes deviations from ideal gas behavior, addressed by Tournier et al. (2006) using a binary noble gas calculation theory. Figure 9 displays the core physical properties (left) and their Modelica-based implementation in the AESE library (right). We validated He-Xe properties against literature (El-Genk et al. 2007), performed steady-state (Figures 10-11) and transient analyses for the He-Xe CBC-NPS, and assessed system performance across design parameters (Figure 12), guiding future optimization and design (Zhang and Wang 2024). These results indicate that the Modelica-based model supports pioneering studies of advanced reactor concepts in the nuclear field.



(a) Schematic diagram of He-Xe CBC system



(b) Visual model of the He-Xe CBC system in AESE

Figure 8. Modelica-based model of the He-Xe CBC system (Zhang and Wang 2024)
1- Reactor; 2-Temperature control module;
3-Temperature sensor;
4-Turbine; 5-Recuperator; 6-Cooler;
7-Cooling control module; 8-Compressor; 9-Generator.

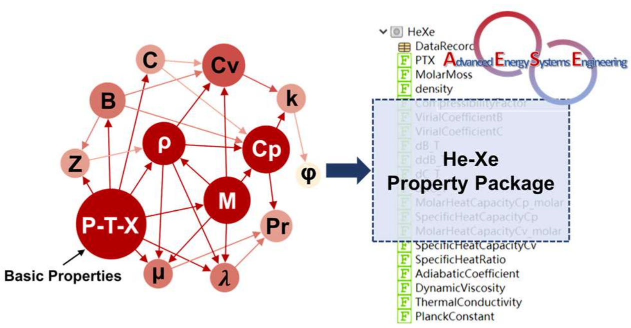


Figure 9. He-Xe property calculation process and property package in AESE (Zhang and Wang 2024).

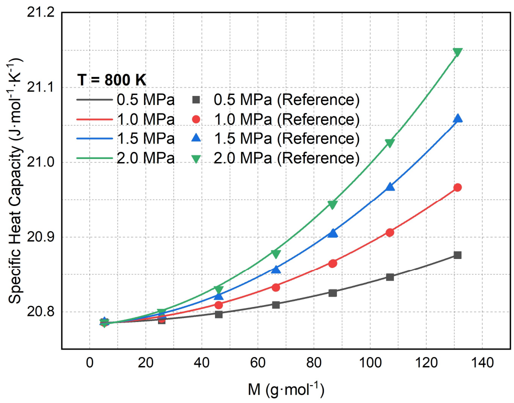


Figure 10. Verification of the specific heat capacity (Zhang and Wang 2024).

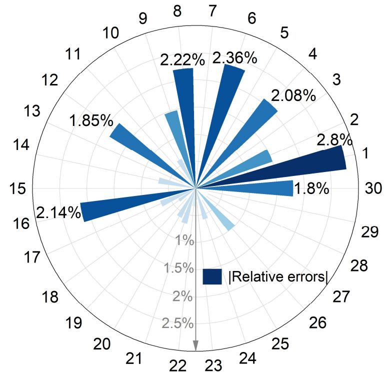


Figure 11. Relative error of key system parameter calculations compared to El-Genk et al. (2007).

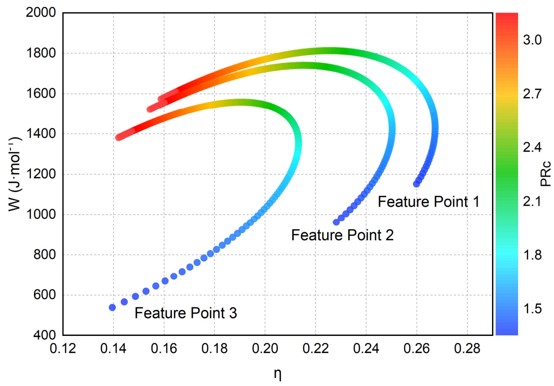


Figure 12. Efficiency and specific work of the system (Zhang and Wang 2024).

3 Further Applications in Different Scenarios

After validation, NPS models hold significant potential for valuable applications in equipment design and operation. For example, system models can drive real hardware to enable rapid control loop validation in HILS; support optimization algorithms for MOO design in large-scale systems; and utilize simulation data to drive intelligent algorithms for system identification and rapid optimization. Models can be further expanded and applied across various scenarios and dimensions.

DOI 10.3384/ecp218373

3.1 Hardware in the Loop Simulation (HILS)

HILS links a physical object with a computer-based simulation model for testing, accurately capturing the controller's dynamic, static, and nonlinear characteristics, making it a highly realistic simulation technology.

To study the variable operating condition responses of nuclear-powered engineering ships (NPES), we initially developed a desktop-level HILS model framework (Zhang et al. 2024). The HILS platform framework (Figure 13) integrates software, hardware, and a communication chain (Zhang et al. 2024), with the system model executed in OpenModelica and real-time signal transmission handled via an Arduino module. This configuration enables real-time coupling between the virtual NPES system and physical controllers for realistic validation and testing scenarios.

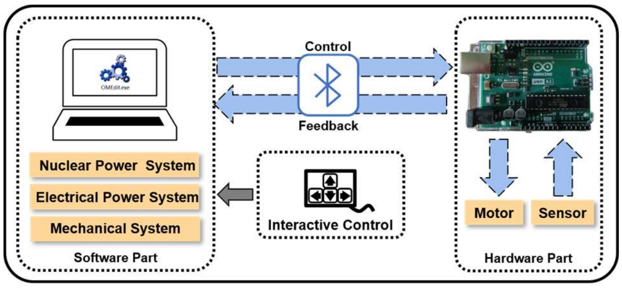


Figure 13. Framework of the HILS platform for an NPES (Zhang et al. 2024).

Both the offline model and the HILS software-level model are shown in Figure 14. The offline model uses the "Transient Signal Module," while the HILS model employs the "HILS Signal Module," enabling hardware communication via an Arduino module (COM3 port, signal pins) for control signals and sensor data. Real-time user interaction is supported through keyboard inputs, processed by signal control logic. The offline model is primarily used for preliminary system validation, whereas the HILS model allows real-time interaction with physical hardware for controller testing and signal feedback.

Figure 15 shows the HILS signal filtering process for the NPES, averaging the signal every second. The PID control was critical during the NPES's three-stage deceleration, as demonstrated by the velocity curve in Figure 16.

These results confirm the feasibility and effectiveness of the Modelica-based HILS platform for simulating and analyzing the dynamic responses of NPES under variable operating conditions. The key advantage of Modelica lies in its ability to represent multi-domain system models and integrate them with real hardware components, enabling the construction of more comprehensive and organically coupled HILS frameworks.

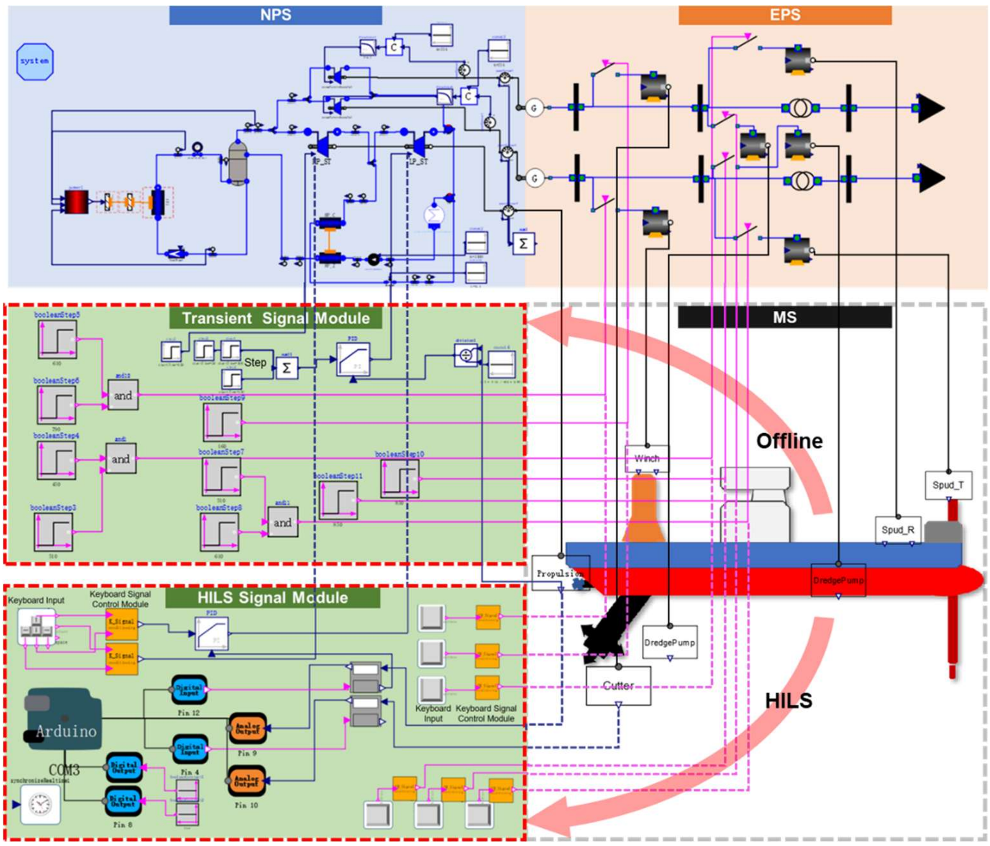


Figure 14. Software-level Models of the NPES in Offline and HILS Modes (Zhang et al. 2024).

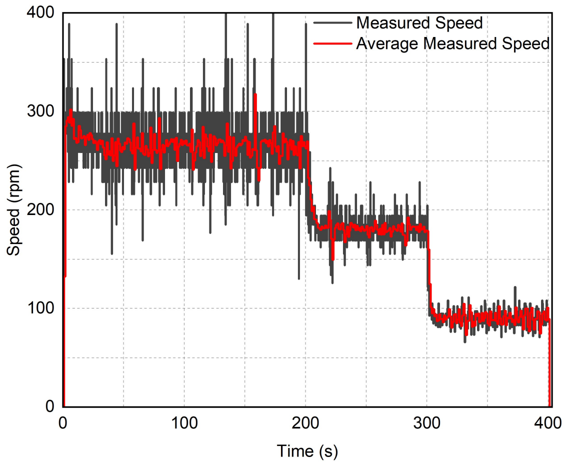


Figure 15. Simple filter process of the collected hardware response signal (Zhang et al. 2024).

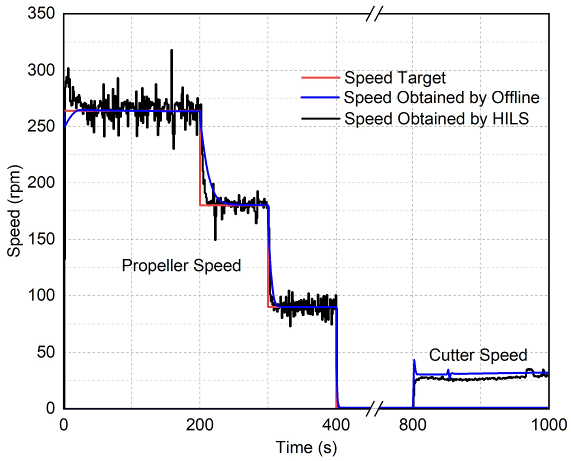


Figure 16. Data recording of hardware response and comparison of propeller speed control in HILS full condition experiment (Zhang et al. 2024).

3.2 Multi-Objective Optimization

In the He-Xe CBC system, multiple performance targets cannot be simultaneously maximized under the same operating condition, which necessitates a multi-objective optimization (MOO) approach. The NSGA-II algorithm (Deb et al. 2002) is applied within the feasible domain to achieve balanced optimization across conflicting objectives.

We implemented an NSGA-II optimization framework in Python (Jazzbin et al. 2020), coupled with the He-Xe CBC system model in OpenModelica via OMPython (Ganeson 2012) for MOO calculations (Zhang and Wang 2024). Figure 17 illustrates the MOO methodology architecture. OMPython links the NSGA-II algorithm to OpenModelica, relaying parameter settings for simulation and returning results to the optimization process. The MOO objectives include system efficiency, specific work, and the Brayton cycle system's equivalent volume with turbomachinery aerodynamic load distribution.

Figure 18 illustrates the Pareto optimal solution set obtained by optimizing two selected objectives, and Figure 19 shows the Pareto optimal solution set for three-objective optimization with different population sizes.

These results demonstrate the Modelica-based model's capability to integrate additional algorithms to support complex MOO studies in energy systems.

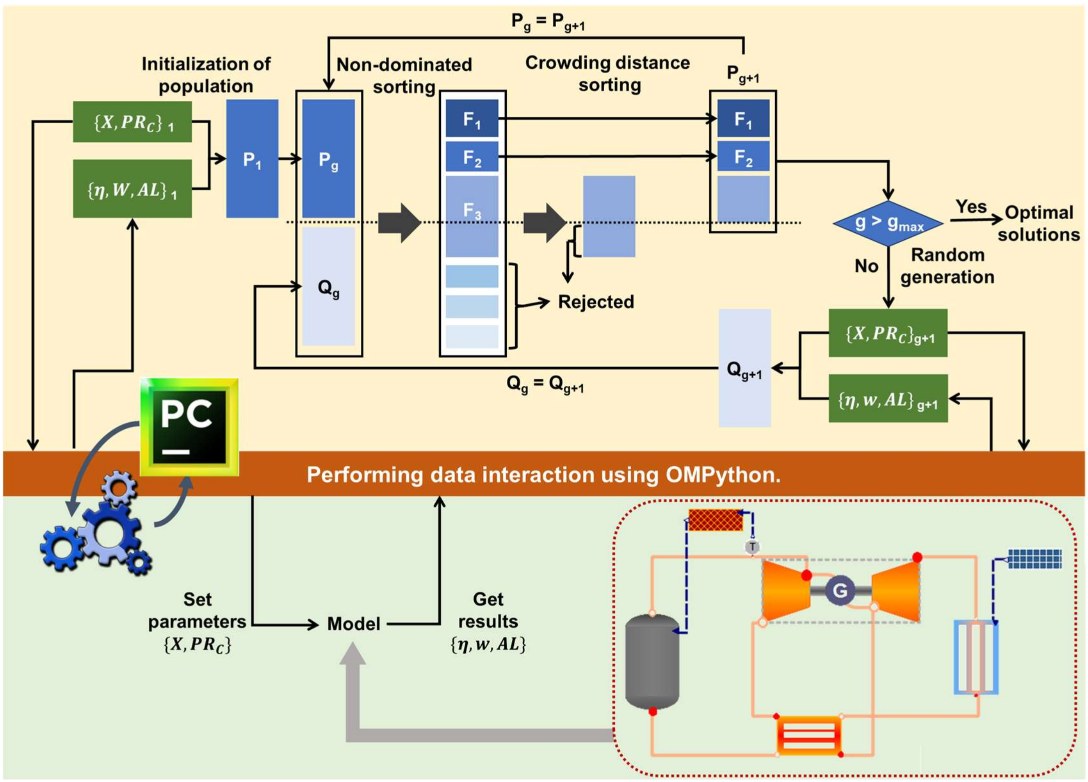
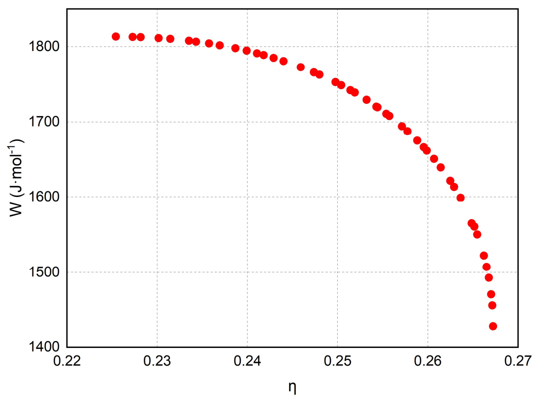
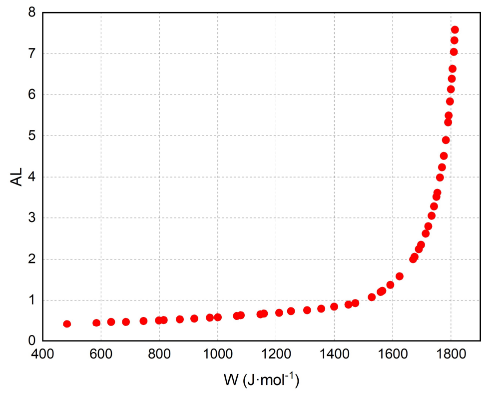


Figure 17. Workflow of NSGA-II MOO using OpenModelica and OMPython (Zhang and Wang 2024).



(a) Efficiency–specific work



(b) Specific work–aerodynamic load

Figure 18. Pareto optimal solution set for two-objective (Zhang and Wang 2024).

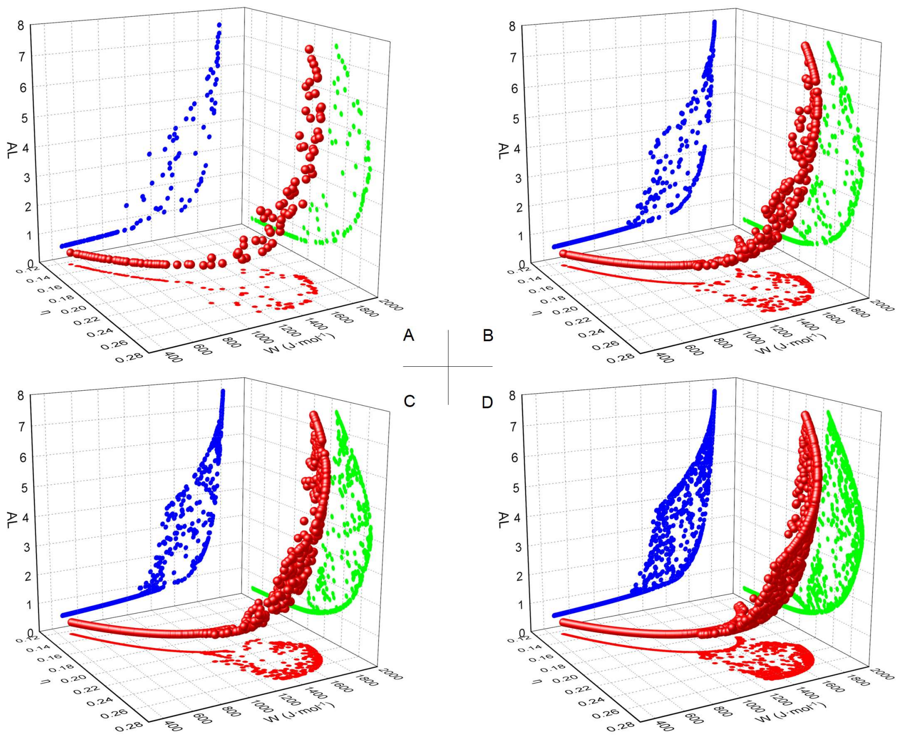


Figure 19. Pareto optimal solution set for three-objective optimization under different population sizes (Zhang and Wang 2024).

3.3 System Identification

System identification is essentially system approximation, aiming to find a model with acceptable accuracy from the data. The resulting model is an approximate representation of the actual system (Ljung 2010). The surrogate model obtained after system identification can accelerate the computational process, helping to reduce the time spent on repetitive calculations or iterations of system models. This is particularly beneficial in scenarios such as multi-objective optimization design and multi-dimensional model coupling calculations.

In the identification of steady-state systems, we constructed a result dataset based on simulations with a variety of different design parameters. The resulting dataset was used to train a neural network model, establishing a proxy for the system-wide input-output mapping. The BPNN is structured with an input layer, hidden layers, and an output layer (Figure 20). Nodes in the input layer represent the system's design parameters. The output layer includes the system's operational or performance parameters, such as temperatures and pressures at various device nodes, system efficiency, specific work, and aerodynamic loading. Each output parameter is mapped through a multi-node hidden layer, collectively forming a comprehensive model that represents the input-output relationships across the entire system.

Figure 21 compares the simulation results and network model calculations for 10 sets of randomly generated design parameters. The maximum relative error was approximately 1%, which indicates that the precision of the constructed network model is acceptable. Furthermore, the network model, which can surrogate the original model, can be efficiently reused to meet a wider range of computational needs or scenarios.

We are also working on the identification of dynamic system models in AESE, using dynamic mode decomposition (DMD) methods to predict the system's evolution over the long term or during a specific phase (Figure 22), as well as using controlled dynamic mode decomposition (DMDc) methods for rapid prediction of multi-condition transient scenarios (Figure 23). DMD extracts coherent spatiotemporal structures from system data, while DMDc incorporates control inputs to capture input–output dynamics (Zhang et al. 2025). The rapid prediction of a system's transient response aids in efficiently and intelligently handling dynamic data changes, and is of great significance for scenarios such as fast optimization of control systems, lifespan prediction, and fault diagnosis of NPSs.

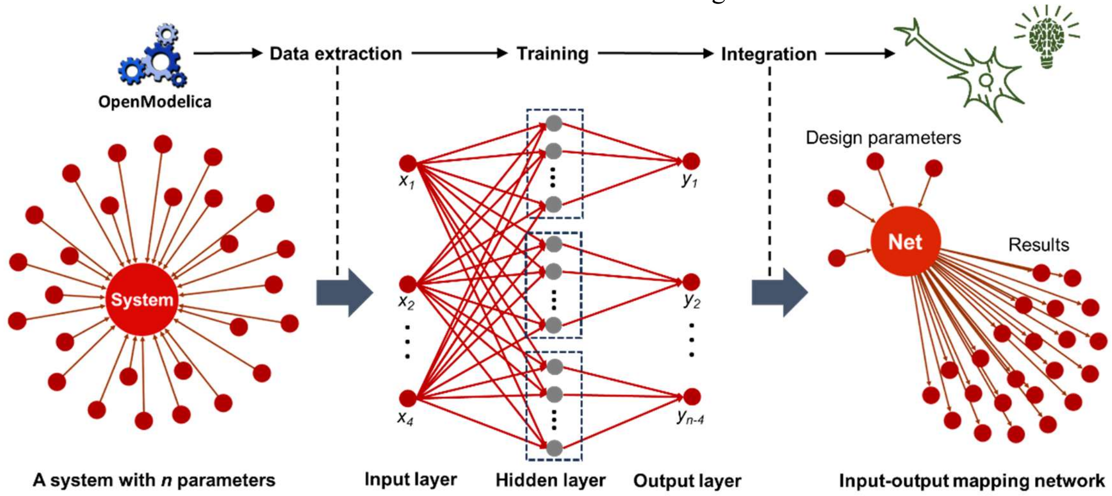


Figure 20. Neural network structure for mapping system design parameters to key performance outputs.

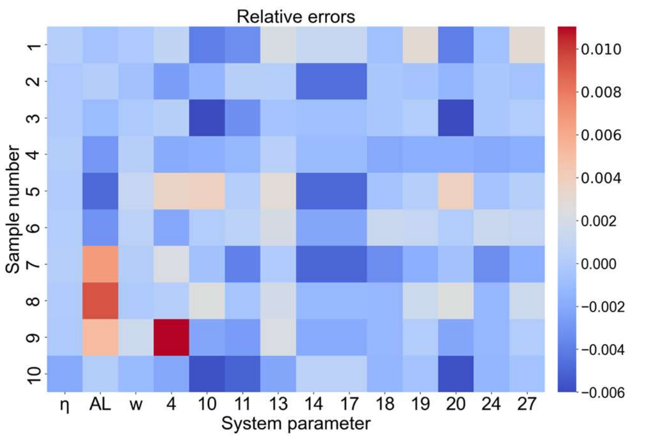


Figure 21. Verification results of the network model for key system parameter.

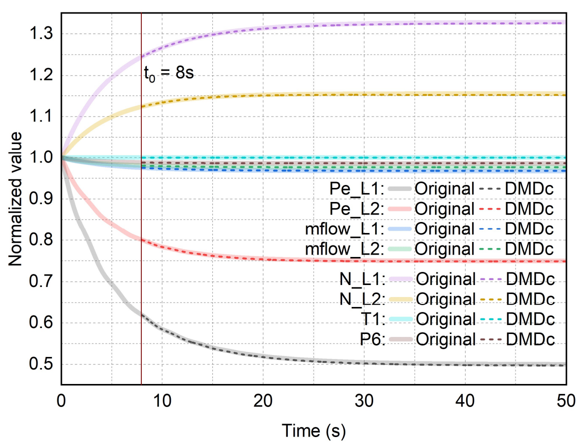


Figure 22. Dynamic prediction of key system parameters based on DMD (Zhang et al. 2025).

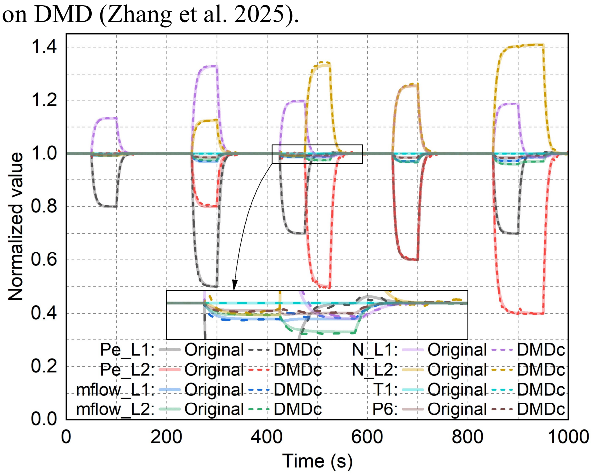


Figure 23. Dynamic prediction of key system parameters based on DMDc (Zhang et al. 2025).

3.4 Rapid Multi-Objective Optimization

As mentioned in the previous section, MOO can be more efficient due to the rapid computation enabled by the surrogate model formed after system identification.

During the optimization process, even with parallel computing, the process remains complex and time-consuming. This paper integrates the BPNN algorithm into machine learning to establish a data-driven multi-objective (DDMO) optimization framework (Figure 24). Using this framework, the system model simulation is not directly embedded in the MOO algorithm. Instead, a large

DOI 10.3384/ecp218373

dataset is first generated through simulations under various design parameters to collect data on different design levels of the system model. Using the BPNN algorithm, a network model is trained based on this dataset to accurately map the original system simulation model. Then, this network model is integrated into the MOO algorithm framework for the optimization design of the system.

Figure 25 presents the results of MOO for the He-Xe CBC system for different numbers of individuals (I) and generations (G) for both model-driven multi-objective (MDMO) and DDMO approaches. The results show consistency between the two modes of optimization. The DDMO optimization had significantly shorter computation times than the traditional MODO approach (Table 3). The DDMO shows faster optimization while maintaining accuracy. Even considering the impact of computer performance fluctuations, the data clearly shows that the DDMO framework accelerated the optimization process by over 10000 times.

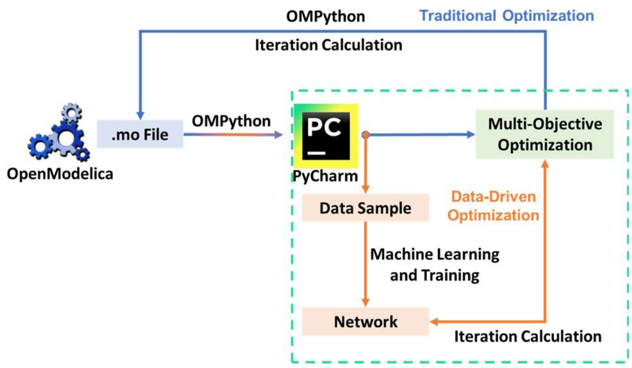


Figure 24. DDMO optimization framework.

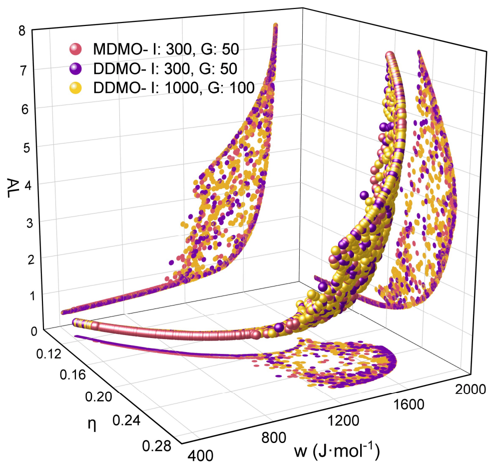


Figure 25. Results of the DDMO.

Table 2. Comparison of calculation times for two optimization methods.

Project	*MDMO*	*DDMO*	*DDMO*
Dual objectives	I: 50 G: 100	I: 50 G: 100	I: 1000 G: 100
η-w	75189.5 s	2.2 s	260.7 s
η-AL	58801.6 s	1.9 s	241.8 s
w-AL	56538.8 s	2.1 s	255.3 s
Triple objectives	I: 300 G: 50	I: 300 G: 50	I: 1000 G: 100
η-w-AL	164502.9 s	55.8 s	185.7 s

4 Conclusions

This paper provides a brief introduction to the AESE library developed for energy systems, particularly NPSs, based on the Modelica language on the OpenModelica platform. It summarizes the modeling and simulation validation of two classic types of models and specifically describes four scenarios for the further application of system simulation models.

Based on Modelica for multi-domain visualization development targeting NPSs, and benefiting from the openness of the OpenModelica platform, system models can be more conveniently integrated with hardware or algorithms to complete tasks such as hardware-in-the-loop simulation, multi-objective optimization, and system identification, addressing task requirements in design verification, operational control, fault diagnosis, and other scenarios in practical engineering with greater accuracy, safety, and speed. Model development is not limited to merely establishing models; further application of models also holds practical significance. Making good use of models and their data can lead to higher productivity.

In current applications, data-driven surrogate models often exhibit strong specificity, which is closely related to the datasets of their particular application targets. As a result, the advantages of surrogate models have not received sufficient attention. To enable more generalized rapid design and optimization of nuclear energy systems, our next step will focus on abstracting the essential nature of energy system models and integrating identification algorithms. This will allow intelligent participation and decision-making throughout the entire design and operation process within a moderately generalized NPS. In addition, to compensate for the limited accuracy of system simulation in modeling key components, the integration of Modelica-based system simulation with other three-dimensional computational tools also deserves further attention.

References

Andreasson, J., Andersson, D., Batteh, J., Gohl, J., Griffin, J., Krueger, I. (2013) “Integrated simulation of a e4WD vehicle using Modelica”, *IFAC Proceedings Volumes*, 46(21), pp. 440–445. DOI:10.3182/20130904-4-JP-2042.00097.

Briese, L.E., Acquatella B, P., Schnepper, K. (2020) “Multidisciplinary modeling and simulation framework for launch vehicle system dynamics and control”, *Acta Astronautica*, 170, pp. 652–664. DOI:10.1016/j.actaastro.2019.08.022.

Cammi, A., Casella, F., Ricotti, M.E., Schiavo, F. (2011) “An object-oriented approach to simulation of IRIS dynamic response”, *Progress in Nuclear Energy*, 53(1), pp. 48–58. DOI:10.1016/j.pnucene.2010.09.004.

Casella, F. and Leva, A. (2005) “Object-Oriented Modelling & Simulation of Power Plants with Modelica”, in Proceedings of the 44th IEEE Conference on Decision and Control. 44th IEEE Conference on Decision and Control, Seville, Spain: IEEE, pp. 7597–7602. DOI:10.1109/CDC.2005.1583388.

Casella, F. and Leva, A. (2024) “Modelling and Simulation of Thermal Power Generation Systems with Modelica: 20 Years of Experience with the ThermoPower Library”, *IFAC-PapersOnLine*, 58(13), pp. 227–234. Available at: https://doi.org/10.1016/j.ifacol.2024.07.487.

Deb, K., Pratap, A., Agarwal, S., Meyarivan, T. (2002) “A fast and elitist multiobjective genetic algorithm: NSGA-II”, *IEEE Transactions on Evolutionary Computation*, 6(2), pp. 182–197. DOI:10.1109/4235.996017.

El-Genk, M.S. and Tournier, J.-M. (2007) “Noble-Gas Binary Mixtures for Closed-Brayton-Cycle Space Reactor Power Systems”, *Journal of Propulsion and Power*, 23(4), pp. 863–873. DOI:10.2514/1.27664.

Fritzson P. (2012) *Introduction to Modeling and Simulation of Technical and Physical Systems with Modelica: Introduction to Modeling and Simulation of Technical and Physical Systems with Modelica*. ISBN:9781118094259.

Fritzson, P. *et al.* (2020) “The OpenModelica Integrated Environment for Modeling, Simulation, and Model-Based Development”, in *Modeling, Identification and Control: A Norwegian Research Bulletin*, pp. 241–295. Available at: https://doi.org/10.4173/mic.2020.4.1.

Ganeson, A. (2012) *Design and Implementation of a User Friendly OpenModelica - Python interface.* http://urn.kb.se/resolve?urn=urn:nbn:se:liu:diva-78992.

Jazzbin, et al. (2020) *geatpy: The genetic and evolutionary algorithm toolbox with high performance in python.* Available from: http://www.geatpy.com/.

Ljung, L. (2010) “Perspectives on system identification”, Annual Reviews in Control, 34(1), pp. 1–12. DOI:10.1016/j.arcontrol.2009.12.001.

Mattsson, S.E., Elmqvist, H. and Otter, M. (1998) “Physical system modeling with Modelica”, *Control Engineering Practice*, 6(4), pp. 501–510. DOI:10.1016/S0967-0661(98)00047-1.

Mihalič, F., Truntič, M. and Hren, A. (2022) “Hardware-in-the-Loop Simulations: A Historical Overview of Engineering Challenges”, *Electronics*, 11(15), p. 2462. DOI:10.3390/electronics11152462.

Peng, M.J. (2009) *Marine Nuclear Power Plant, 1st ed.*, Beijing: Atomic Energy Press. https://www.las.ac.cn/front/book/detail?id=2307eba88938e84d77e7a76931a15cc1.

Soriano, T., Hien, N.V., Tuan, K.M., Anh, T.V. (2016) “An object-unified approach to develop controllers for autonomous underwater vehicles”, *Mechatronics*, 35, pp. 54–70. DOI:10.1016/j.mechatronics.2015.12.011.

Tournier, J.-M., El-Genk, M. and Gallo, B. (2006) “Best Estimates of Binary Gas Mixtures Properties for Closed Brayton Cycle Space Applications”, in *4th International Energy Conversion Engineering Conference and Exhibit (IECEC)*. *4th International Energy Conversion Engineering Conference and Exhibit (IECEC)*, San Diego, California: American Institute of Aeronautics and Astronautics. DOI:10.2514/6.2006-4154.

Zhang, A., He, X., Cammi, A., Wang, X. (2024) “Modelica and Arduino-based hardware-in-the-loop simulation of a nuclear-powered engineering ship”, *Nuclear Engineering and Design*, 429, p. 113650. DOI:10.1016/j.nucengdes.2024.113650.

Zhang, A. and Wang, X. (2024) “Development of Modelica-based one-dimensional thermodynamic cycle library and its application in simulation and multi-objective optimization of a He–Xe closed-Brayton-cycle system”, *Progress in Nuclear Energy*, 172, p. 105205. DOI: 10.1016/j.pnucene.2024.105205.

Zhang, A., Cammi, A. and Wang, X. (2025) “System-level data-driven identification of complex closed Brayton cycles via dynamic mode decomposition”, *Applied Thermal Engineering*, p. 127438. DOI: https://doi.org/10.1016/j.applthermaleng.2025.127438.

DOI 10.3384/ecp218373