Adaptive Governor Control for Pressurized Water Reactors Using Reinforcement Learning with Fuzzy Rewards: Enhancing Load-Following and Grid Stability

By: Ahmed Abdelrahman Ibrahim
Under supervision of: Prof. Lim Hak-kyu

---

1. Introduction and Problem Statement

Pressurized Water Reactors (PWRs) are a cornerstone of reliable, low-carbon electricity generation, traditionally designed for baseload operation (e.g., 90-100% of their 900-1200 MWe capacity). However, the increasing integration of variable renewable energy sources (vRES) like wind and solar introduces significant volatility into the power grid, with demand potentially fluctuating by 20-50% within hours. This paradigm shift necessitates that PWRs develop robust **load-following** capabilities, enabling them to dynamically adjust their power output (e.g., across a 50-100% capacity range) to maintain grid frequency stability, typically 60 Hz ±0.05 Hz in the U.S.. For example, a PWR might need to ramp its power output significantly to compensate for sudden drops in solar generation.

The **steam turbine governor** is critical in this context, as it regulates turbine speed (e.g., 1800 RPM for a 4-pole generator on a 60 Hz grid) and, consequently, electrical power output by modulating the position of steam control valves. While traditional Proportional-Integral-Derivative (PID) controllers are effective under steady-state or near-baseload conditions, their fixed gains, typically tuned for linear system responses, struggle with the inherent non-linearities and rapid transients of aggressive load-following operations. These challenges are exacerbated by factors such as varying steam pressure and changes in effective turbine inertia.

Suboptimal governor control during load-following can lead to several critical issues:

- **Turbine Overspeed:** Exceeding design speed limits (e.g., 120% of nominal, or 2160 RPM for an 1800 RPM nominal speed turbine as per config/parameters.py) can risk catastrophic mechanical failure.
- **Grid Frequency Deviations:** Excursions beyond acceptable limits (e.g., ±0.5 Hz from the nominal 60 Hz, as per config/parameters.py) can disrupt grid synchronization and potentially lead to wider grid instability.
- **Safety Risks:** Transients induced by poor control could, in extreme cases, challenge reactor safety systems by violating operational limits, such as those for fuel temperature (e.g., <2800°C) or primary system pressure, potentially breaching regulatory standards like those from the U.S. Nuclear Regulatory Commission (NRC) 10 CFR 50.55a and IAEA safety guides (e.g., NS-G-1.3).

This research focuses on the development and evaluation of an advanced governor control system for PWRs, leveraging **Reinforcement Learning (RL) with sophisticated Fuzzy Reward Functions**. The goal is to create a control strategy that can adaptively manage the steam valve position, optimizing for load-following performance, grid stability, and operational safety. RL offers the potential to derive complex control policies well-suited to the non-linear dynamics of PWR systems without requiring an exhaustive analytical plant model. The fuzzy logic component of the reward system is designed to intelligently balance multiple, often conflicting, objectives such as maintaining tight frequency control, ensuring fuel and turbine safety margins, and minimizing control effort. This research will conduct a comparative analysis of the proposed RL-based controller against conventional PID controllers and Fuzzy Logic Controllers (FLCs) within a high-fidelity simulation environment, focusing on a simulation-only scope.

---

2. Research Objectives

This research aims to achieve the following objectives:

1. **Develop an Integrated PWR Simulation Environment:** Construct a modular, high-fidelity simulation platform using Python and Gymnasium. This platform will accurately model the coupled dynamics of the PWR reactor core (point kinetics and lumped thermal-hydraulics), the steam turbine-generator system (including governor valve actuators), and its interface with the electrical grid (swing equation). The environment will facilitate repeatable testing and comprehensive evaluation of various control strategies.

2. **Design and Implement an RL-Based Governor Control System with Fuzzy Rewards:** Engineer an RL agent (e.g., based on Soft Actor-Critic, SAC) capable of learning an adaptive steam turbine governor control policy. The agent will be guided by a novel fuzzy reward function that evaluates performance across multiple criteria including operational safety (e.g., maintaining turbine speed below 2160 RPM, fuel temperatures within limits), grid stability (e.g., frequency deviation within ±0.05 Hz of the setpoint), and control efficiency (e.g., minimizing excessive valve movements). The system will incorporate safety-aware features, potentially through reward shaping and operational constraints.
3. **Perform a Comprehensive Comparative Analysis:** Evaluate the performance of the developed RL-based control system against well-tuned PID and FLC strategies. This analysis will be conducted across a diverse set of operational scenarios defined in analysis/scenario_definitions.py, including load ramps, step load changes, grid disturbances, and simulated system component degradations/failures. Performance will be quantified using a suite of metrics assessing dynamic response, safety adherence, and control effort, as calculated by analysis/metrics_engine.py.
4. **Investigate Potential Enhancements and Deployment Considerations (Simulation-Based):** Based on the simulation results, assess the potential of the adaptive RL controller to improve PWR load-following capabilities, enhance grid reliability, and maintain safety margins under challenging operational conditions. This investigation will remain within a simulation-only scope but aims to provide insights that could inform future research towards real-world deployment pathways.

Optional future work, beyond this scope, could involve Monte Carlo simulations for robustness analysis against parameter uncertainties and a more formal Probabilistic Risk Assessment (PRA).

---

3. Literature Review and Research Gap

Control systems for nuclear power plant components, particularly turbine governors, have evolved, yet challenges remain, especially in the context of flexible operation.

- **PID Control:** PID controllers are ubiquitous in industrial applications due to their simplicity and effectiveness in regulating systems around a setpoint. However, their performance degrades in systems with significant non-linearities or varying dynamics, such as a PWR turbine under large load transients. Fixed gains, often tuned for specific operating points (e.g., baseload), can lead to overshoot, slow response, or instability when the plant operates far from that point (e.g., Patel et al., 2012; Åström & Hägglund, 2006). The codebase implements a standard PID with anti-windup and derivative filtering in controllers/pid_controller.py.
- **Fuzzy Logic Control (FLC):** FLCs use rule-based systems based on expert knowledge to handle non-linearities more effectively than PID controllers (e.g., Karr & Gentry, 1993; Precup & Hellendoorn, 2011). They can map linguistic rules about system state to control actions. However, designing and tuning the rule base and membership functions can be complex, and FLCs are typically non-adaptive once designed. The codebase provides an FLC in controllers/flc_controller.py, using scikit-fuzzy.
- **Reinforcement Learning (RL):** RL agents learn optimal control policies through direct interaction with an environment (or a simulation thereof) by maximizing a cumulative reward signal (Sutton & Barto, 2018). This approach is well-suited for complex, non-linear systems where analytical models are difficult to derive or are computationally intensive. Recent advancements in deep RL, particularly actor-critic methods like Soft Actor-Critic (SAC) (Haarnoja et al., 2018), have shown promise in continuous control problems. The application of RL to nuclear systems, especially with safety-critical considerations and sophisticated reward engineering (e.g., using fuzzy logic for reward shaping), is an active area of research.

Research Gap: While PID and FLC have been applied to power plant control, they often lack the adaptability required for modern grid conditions with high vRES penetration. Existing RL applications in the nuclear domain are still emerging, with a need for robust validation across diverse operational scenarios and careful consideration of safety through reward design and learning constraints. Specifically, a comprehensive comparative study of an RL agent guided by a fuzzy multi-objective reward system against PID and FLC, evaluated within a detailed PWR simulation environment like the "PWR Controller Optimization & Monitoring Workbench (DTAF v2.2)", addresses a clear gap. This research aims to fill this by developing and rigorously testing such an RL-fuzzy approach, focusing on enhancing load-following and grid stability within a simulation-only framework.

---

4. Methodology

The research methodology adopts a systems engineering approach, encompassing four main phases: system modeling, control system design and implementation, simulation and testing under diverse scenarios, and comprehensive performance evaluation.

4.1 System Modeling (Integrated Simulation Environment)

A modular simulation environment, PWRGymEnvUnified, has been developed in Python, adhering to the Gymnasium API standard. This environment integrates three core subsystems:

1. **Reactor System (**models/reactor_model.py**):**
    - **Description:** Modeled using point kinetics with six delayed neutron groups to capture time-dependent neutron population dynamics.
    - **Key Dynamics:** Neutron population change based on reactivity, delayed neutron precursors, and thermal feedback from fuel and coolant temperatures affecting reactivity.
    - **Safety Constraints:** Fuel temperature is a critical safety parameter.
    - **Rationale:** The point kinetics model provides a good balance of accuracy for core-average dynamics and computational efficiency suitable for extensive RL training and scenario analysis (Kerlin & Upadhyaya, 2019). Six delayed neutron groups ensure realistic transient behavior.
    - **Implementation:** Solved using SciPy's ODE solvers, with parameters derived from standard PWR data.
2. **Steam Turbine System (**models/turbine_model.py**):**
    - **Description:** A lumped-parameter model representing the turbine-generator's mechanical power output, including first-order dynamics for the governor valve actuator.
    - **Key Dynamics:** Turbine mechanical power changes based on steam flow (modulated by actual valve position and effective steam power from the reactor) and turbine time constants. Valve actuator dynamics model the response of the valve to commanded positions.
    - **Safety Constraints:** Turbine speed is limited to prevent mechanical failure (e.g., 120% of nominal).
    - **Rationale:** This simplified model is computationally efficient for RL and captures the essential dynamics for governor control studies (Patel et al., 2012).
    - **Implementation:** ODEs solved with SciPy, with valve position as the primary control input.
3. **Grid Interface (**models/grid_model.py**):**
    - **Description:** Modeled using the swing equation, describing the grid frequency response to power imbalances between mechanical input from the turbine and the total electrical load (including plant load and grid interactions). The model simulates the generator connected to an infinite bus.
    - **Key Dynamics:** Grid frequency deviation based on the mismatch between mechanical power supplied by the turbine and electrical power consumed (P_mech - P_elec), moderated by generator inertia (H) and damping (D).
    - **Safety Constraints:** Grid frequency is constrained within operational bands (e.g., 60 ±0.5 Hz).
    - **Rationale:** The swing equation is standard for frequency control studies and provides a good representation of generator-grid interaction dynamics for this research's scope (Kundur, 1994).
    - **Implementation:** Solved as a second-order ODE. Electrical load can be a time-varying input defined by scenarios.

The overall modeling approach emphasizes **modularity**, **computational efficiency** for extensive simulations (RL training, multi-scenario analysis), and **realism** in capturing key PWR and grid dynamics relevant to governor control. Standardization via the Gymnasium API ensures reproducibility and facilitates controller development. Key parameters are sourced from config/parameters.py, drawing from established nuclear engineering references and standards where applicable.

4.2 Control System Design and Implementation

Three distinct turbine governor control strategies are implemented and compared:

1. **PID Controller (**controllers/pid_controller.py**):**
    - **Description:** A standard PID controller adjusts the steam valve position based on the error between the desired turbine speed (setpoint, e.g., 1800 RPM) and the measured speed.
    - **Implementation:** Implemented in Python with features like anti-windup and an optional derivative filter. Tuning can be performed using methods like Ziegler-Nichols (for baseline) or more advanced multi-scenario optimization (e.g., Differential Evolution via optimization_suite/pid_global_optimizer.py).
    - **Limitations:** Fixed gains may lead to suboptimal performance under widely varying conditions and non-linear dynamics inherent in load-following.
2. **Fuzzy Logic Controller (FLC) (**controllers/flc_controller.py**):**
    - **Description:** The FLC uses linguistic rules based on expert knowledge to map inputs (speed error and rate of change of speed error) to changes in valve position. It's designed to handle non-linearities more effectively than PID.
    - **Implementation:** Implemented using scikit-fuzzy with Mamdani inference. Membership functions are triangular, and a standard 5x5 rule base is employed. Scaling factors for inputs and output, as well as input/output universe ranges, are tunable, for instance, via Differential Evolution using optimization_suite/flc_optimizer.py.
    - **Limitations:** The rule base is static and does not adapt online. Performance heavily depends on the initial design of rules and membership functions, and tuning can be complex.
3. **Reinforcement Learning with Fuzzy Rewards (RL-FR) (**controllers/rl_interface.py**,** optimization_suite/rl_trainer.py**):**
    - **Description:** This system combines the adaptive learning capabilities of RL with the ability of fuzzy logic to manage uncertainty and balance multiple objectives in the reward signal. The RL agent learns an optimal valve control policy by interacting with the PWRGymEnvUnified simulation environment.
    - **Algorithm:** A Soft Actor-Critic (SAC) agent, a state-of-the-art off-policy actor-critic deep RL algorithm suitable for continuous control tasks, is used (Haarnoja et al., 2018). Implemented using the Stable Baselines3 library.
    - **Reward Structure (**controllers/rl_interface.py**):** The core of the RL agent's guidance is a sophisticated reward function (calculate_reward and calculate_fuzzy_reward in rl_interface.py). This function:
        - Utilizes **fuzzy logic** to evaluate performance across three key dimensions: **safety** (e.g., penalizing deviations towards fuel temperature limits, turbine speed limits, and frequency limits), **stability** (e.g., rewarding tight grid frequency and turbine speed control around setpoints), and **efficiency/effort** (e.g., penalizing excessive valve movements).
        - Combines these fuzzy scores with direct penalties for approaching or violating critical operational limits and for undesirable rates of change in key variables.
        - Incorporates critical warning penalties for operating too close to safety boundaries, even if not strictly violating them.

- **Advanced Training Features (**optimization_suite/rl_trainer.py**):** Includes curriculum learning (gradual increase in scenario complexity), domain randomization (varying environment parameters for robustness), and a basic safety shield mechanism.
- **Advantages:** Potential for high adaptability to changing plant conditions and complex dynamics. The fuzzy reward system aims to provide a more nuanced and interpretable guidance signal to the RL agent, balancing multiple objectives effectively. Built-in safety considerations in the reward function and training process are crucial for nuclear applications.

4.3 Simulation Scenarios and Testing (analysis/scenario_definitions.py)

The performance of the implemented controllers (PID, FLC, RL-FR) will be evaluated across a comprehensive suite of operational scenarios. These scenarios are designed to test adaptability, robustness, and safety under various conditions. The specific scenarios are defined in analysis/scenario_definitions.py and include:

1. **gradual_load_increase_10pct:** Gradual load increase from 90% to 100% of nominal load over 300 seconds.
2. **sudden_load_increase_5pct:** Sudden +5% step increase in load to induce frequency drop.
3. **speed_sensor_failure_30s:** Turbine speed sensor reads nominal (1800 RPM) for 30 seconds.
4. **emergency_shutdown_2s:** Valve closes linearly over 2 seconds to simulate an emergency shutdown command.
5. **steam_pressure_drop_effect_10s:** Simulated steam pressure drop via a 30% reduction in heat transfer efficiency (eta_transfer) over 10 seconds.
6. **oscillating_load_3pct_0_5Hz:** Load oscillates by ±3% of nominal at 0.5 Hz for 10 seconds.
7. reduced_efficiency_steady_state: Steady-state operation with a permanently reduced heat transfer efficiency.
8. **load_fluctuation_50MW_1min:** Load fluctuates by ±50 MW with a 60-second period.
9. **external_grid_disturbance_neg100MW:** Simulated external grid power imbalance (loss of 100MW external generation) for 50 seconds.

The analysis/scenario_executor.py module manages the execution of these scenarios with the selected controllers.

4.4 Performance Evaluation (analysis/metrics_engine.py, analysis/report_generator.py)

Controller performance is assessed using a range of quantitative metrics calculated by analysis/metrics_engine.py. Key metrics include:

- **Dynamic Response:** Settling time (for frequency and speed), maximum overshoot/undershoot (speed), maximum frequency deviation, Integral Absolute Error (IAE for frequency and speed).
- **Safety Adherence:** Total time spent outside safety limits (fuel temperature, turbine speed, grid frequency), maximum fuel temperature, maximum speed.
- **Control Effort:** Sum of absolute/squared valve movements, number of valve reversals.
- **Consistency:** Performance variation across different scenarios will be qualitatively and quantitatively assessed via reports.

Results, including time-series plots and metric comparisons, will be generated by analysis/visualization_engine.py and compiled into comprehensive reports by analysis/report_generator.py. The reports will facilitate a comparative "winner analysis" based on predefined criteria.

---

5. Tools and Technologies

The project leverages the following key tools and technologies, as indicated in requirements.txt and the codebase structure:

- **Programming Language:** Python 3.12
- **Core Scientific Libraries:** NumPy (<2.0), SciPy (ODE solving, optimization), Pandas (data manipulation).
- **Simulation Environment:** Gymnasium (for PWRGymEnvUnified).

- **Reinforcement Learning:** Stable Baselines3 (for SAC agent implementation), PyTorch (as the deep learning backend).
- **Fuzzy Logic:** scikit-fuzzy (for FLC implementation and fuzzy reward components).
- **Visualization:** Matplotlib, Seaborn.
- **Configuration & Reporting:** PyYAML (for parameter handling), Jinja2 (for report templates).
- **Experiment Tracking (RL):** Weights & Biases (wandb) for logging RL training progress and results.
- **User Interface:** Streamlit (for ui/app.py).

---

6. Expected Outcomes and Contributions

This research is expected to yield the following outcomes:

1. **An Integrated PWR Simulation Platform (DTAF v2.2):** A validated, modular, and reusable simulation environment for PWR dynamics, turbine control, and grid interaction, suitable for control design and analysis.
2. **Advanced RL-FR Controller:** A well-characterized RL-based steam turbine governor controller employing a fuzzy reward system, demonstrating adaptive and robust performance across diverse operational scenarios. It is anticipated that the RL-FR controller will outperform or show significant advantages over traditional PID and static FLC approaches, particularly in complex, non-linear load-following transients.
3. **Comparative Performance Insights:** Quantitative data and comprehensive analysis reports comparing the dynamic response, safety adherence, and control efficiency of RL-FR, PID, and FLC strategies under a wide range of conditions.
4. **Simulation-Based Deployment Pathways:** Insights into the potential benefits and challenges of deploying adaptive, learning-based control systems in nuclear power plant applications, specifically for enhancing operational flexibility and grid support.

The primary contributions of this research include:

- **Innovation:** Advancing the application of RL with fuzzy multi-objective rewards for sophisticated control of safety-critical nuclear power plant systems.
- **Safety and Reliability:** Emphasizing safety-conscious design in the RL reward structure and evaluating controllers against stringent safety and operational limits.
- **Impact:** Providing a simulation-validated pathway to enhance PWR load-following capabilities, thereby supporting greater integration of renewable energy sources and improving overall grid stability and resilience.

---

7. Impact Pathways for the Nuclear Community

The outcomes of this simulation-based research can inform several areas within the nuclear community:

- **Regulatory and Certification:** While simulation-only, this work can contribute to discussions on methodologies for validating and certifying AI-based control systems for nuclear applications, highlighting the importance of comprehensive scenario testing and robust performance metrics.
- **Operational Flexibility:** Demonstrating the potential of advanced adaptive controllers to improve the load-following capabilities of existing and future PWRs, making them more agile partners in grids with high vRES penetration.
- **Research and Development:** Providing a benchmark simulation model and controller implementations that can be used by other researchers for further development, comparison, and exploration of advanced control concepts in nuclear engineering.
- **Industry Application:** Offering a scalable and extensible framework (DTAF v2.2) that, with further development and validation (including potential Hardware-in-the-Loop testing), could form the basis for industry-grade tools for control system design and analysis.

---

8. Conclusion

This research proposes a comprehensive, simulation-based investigation into adaptive governor control for PWRs using Reinforcement Learning with Fuzzy Rewards. By developing a detailed simulation environment (DTAF v2.2) and rigorously comparing the RL-FR controller against PID and FLC strategies across numerous operational scenarios, this work aims to demonstrate a viable path towards enhancing PWR load-following performance, grid stability, and operational safety. While advanced analyses like full Monte Carlo simulations and PRA are outside the current core scope, the framework is designed to be extensible for such future explorations, ensuring flexibility and continued relevance.

---

9. References

- Åström, K. J., & Hägglund, T. (2006). *Advanced PID Control*. ISA - The Instrumentation, Systems, and Automation Society.
- Berner, J., et al. (2023). *Gymnasium: An API for Reinforcement Learning*. arXiv:2306.12793. (Reflects the use of Gymnasium, successor to OpenAI Gym)
- Camacho, E. F., & Bordons, C. (2007). *Model Predictive Control* (2nd ed.). Springer. *(Retained for context, though MPC is not in the final codebase)*
- Haarnoja, T., Zhou, A., Abbeel, P., & Levine, S. (2018). Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor.[1] *Proceedings of the 35th International Conference on Machine*[2] *Learning (ICML)*.
- IAEA. (2002). *Instrumentation and Control Systems Important to Safety in Nuclear Power Plants* (NS-G-1.3). International Atomic Energy Agency.
- IAEA. (2016). *Adaptive Non-Linear Model-Based Control for Nuclear Power Plants*. IAEA Nuclear Energy Series No. NP-T-3.16. Vienna.
- IEEE. (2018). *IEEE Standard Criteria for Safety Systems for Nuclear Power Generating Stations* (IEEE Std 603-2018).
- IEEE. (2020). *IEEE Recommended Practice for Excitation System Models for Power System Stability Studies* (IEEE Std 421.5-2016). *(Relevant for generator/grid modeling context)*
- IEA. (2022). *World Energy Outlook 2022*. International Energy Agency.
- Karr, C. L., & Gentry, E. J. (1993). Fuzzy Control of pH Using Genetic Algorithms. *IEEE Transactions on Fuzzy Systems, 1*(1), 46-53.
- Kerlin, T. W., & Upadhyaya, B. R. (2019). *Dynamics and Control of Nuclear Reactors*. Academic Press.
- Kundur, P. (1994). *Power System Stability and Control*. McGraw-Hill. *(Classic reference for swing equation and grid dynamics)*
- Li, Y., He, W., Reformat, M., & Gassert, H. (2021). Reinforcement Learning for Load Following of Small Modular Reactors with Parameter and Scenario Uncertainties. *Nuclear Engineering and Design, 385*, 111539.
- NRC. (2007). *Component Reliability Data Sheets Update* (NUREG/CR-6928). U.S. Nuclear Regulatory Commission.
- NRC. (2023). *10 CFR 50.55a - Codes and Standards*. U.S. Nuclear Regulatory Commission.
- Patel, R., Bhatikar, D. P., & Kothari, D. P. (2012). Adaptation of PID Controller Using AI Technique for Speed Control of Isolated Steam Turbine. *2012 IEEE International Conference on Power Electronics, Drives and Energy Systems (PEDES)*, 1-6.
- Precup, R. E., & Hellendoorn, H. (2011). A survey on industrial applications of fuzzy control. *Computers in Industry, 62*(3), 213-226.
- Rafferty, J., et al. (Stable Baselines3 team). (2021). *Stable Baselines3*. GitHub. https://github.com/DLR-RM/stable-baselines3. *(Reference for the RL library used)*

- Sutton, R. S., & Barto, A. G. (2018). *Reinforcement Learning: An Introduction* (2nd ed.). MIT Press.
- Tchodom, M., Mbihi, J., & Retif, J. M. (2019). Nonlinear Control of Steam Turbine Power Using Fuzzy Logic Approach. *International Journal of Power Electronics and Drive Systems (IJPEDS), 10*(3), 1251.
- U.S. Department of Energy. (2020). *Light Water Reactor Sustainability Program: Flexible Plant Operation and Generation Value Proposition*. Report INL/EXT-19-55727.
- Wang, Z., Wu, Y., & Dehghanian, P. (2022). Deep Reinforcement Learning for Power System Applications: An Overview. *IEEE Transactions on Smart Grid, 13*(4), 3186-3198.
- Zadeh, L. A. (1965). Fuzzy sets. *Information and Control, 8*(3), 338-353. *(Seminal paper on Fuzzy Logic)*
- Zhao, X., et al. (2021). Deep Reinforcement Learning for Power System Control: A Survey. *International Journal of Electrical Power & Energy Systems, 133*, 107234.