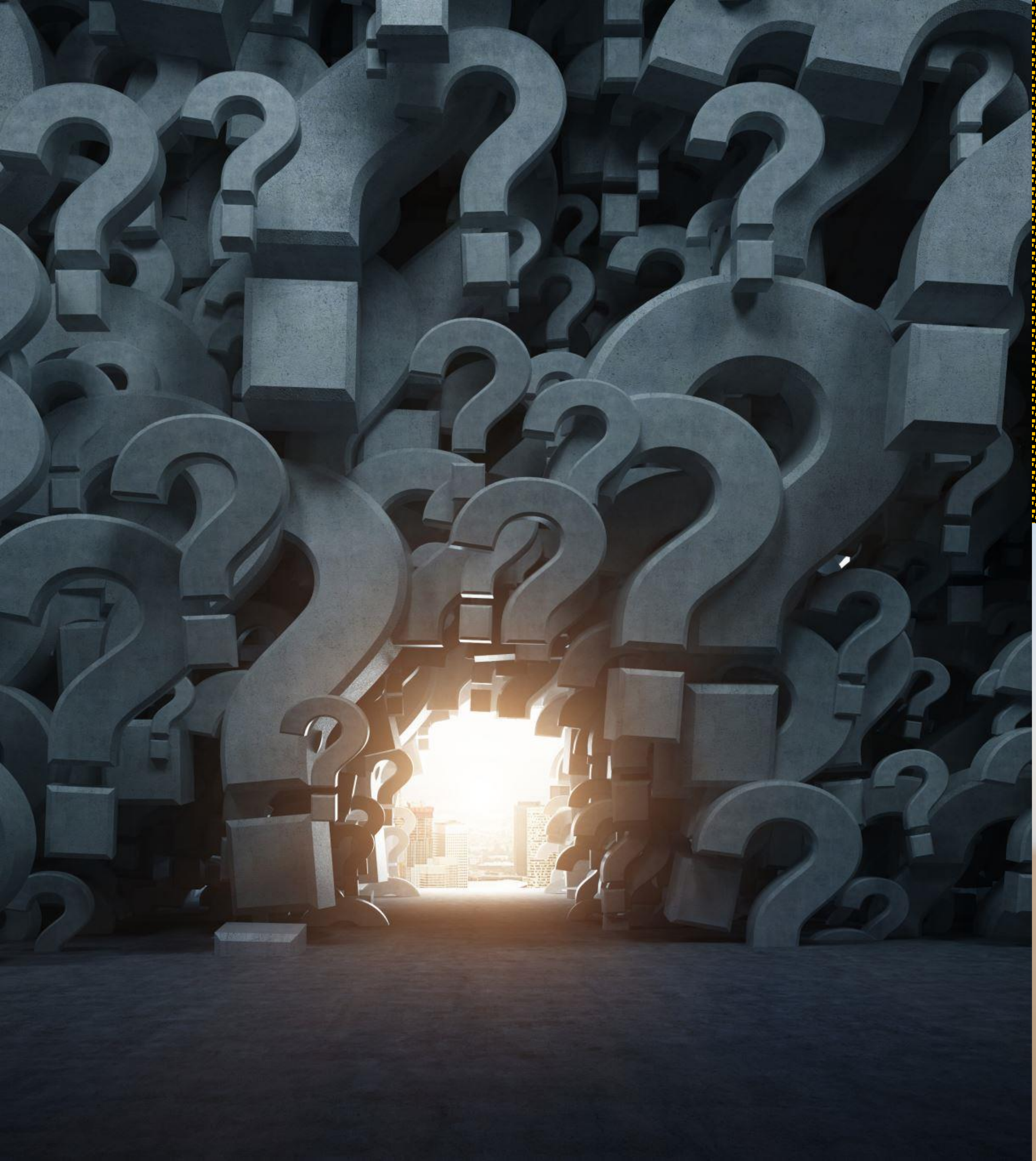
Deterministic Assurance Framework for Licensable AI Grid-Interactive Nuclear Control

Enhancing Load Following and Grid Stability

By: Ahmed Abdelrahman Ibrahim

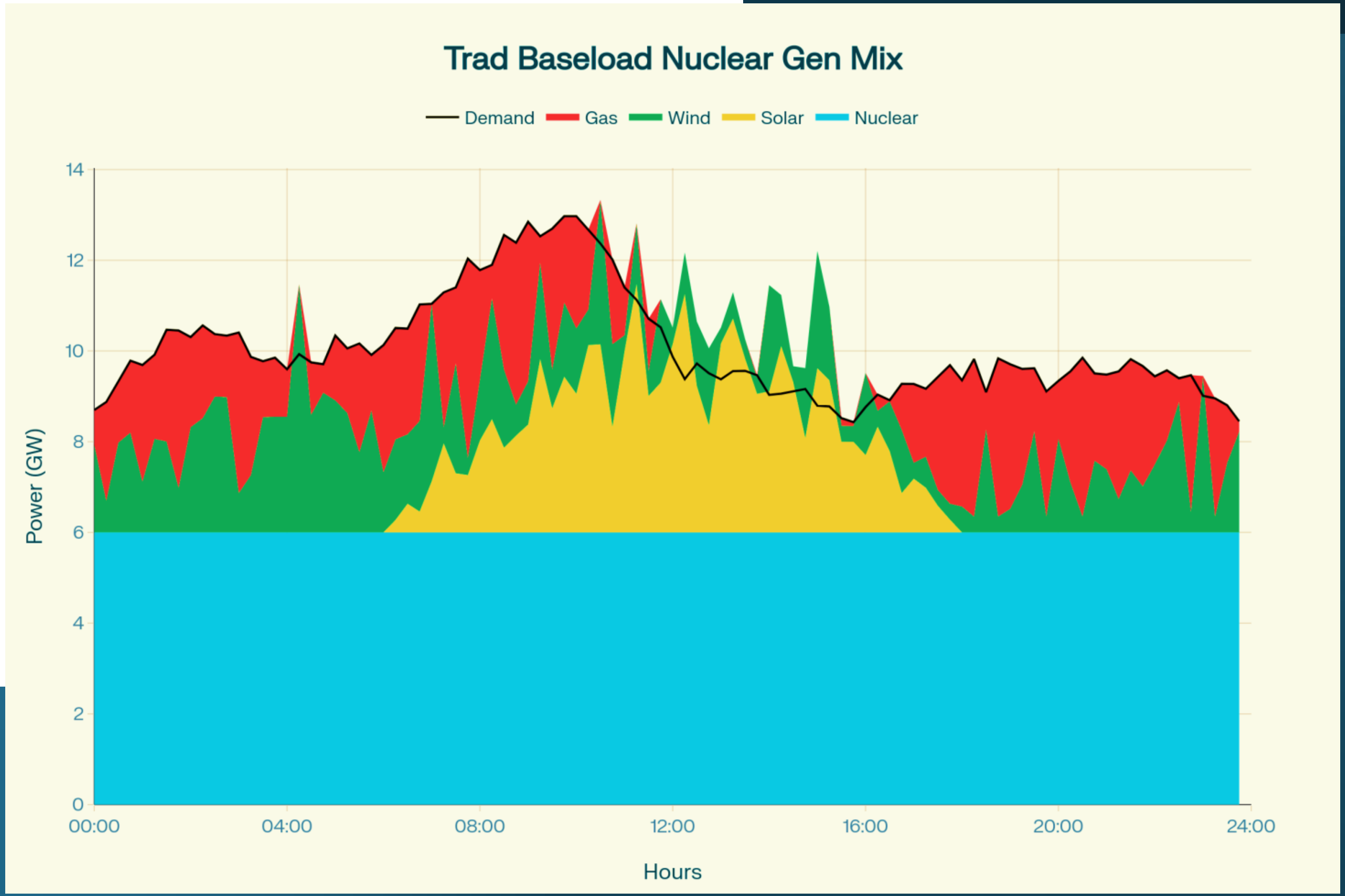
Under supervision of: Prof. Lim Hak-kyu



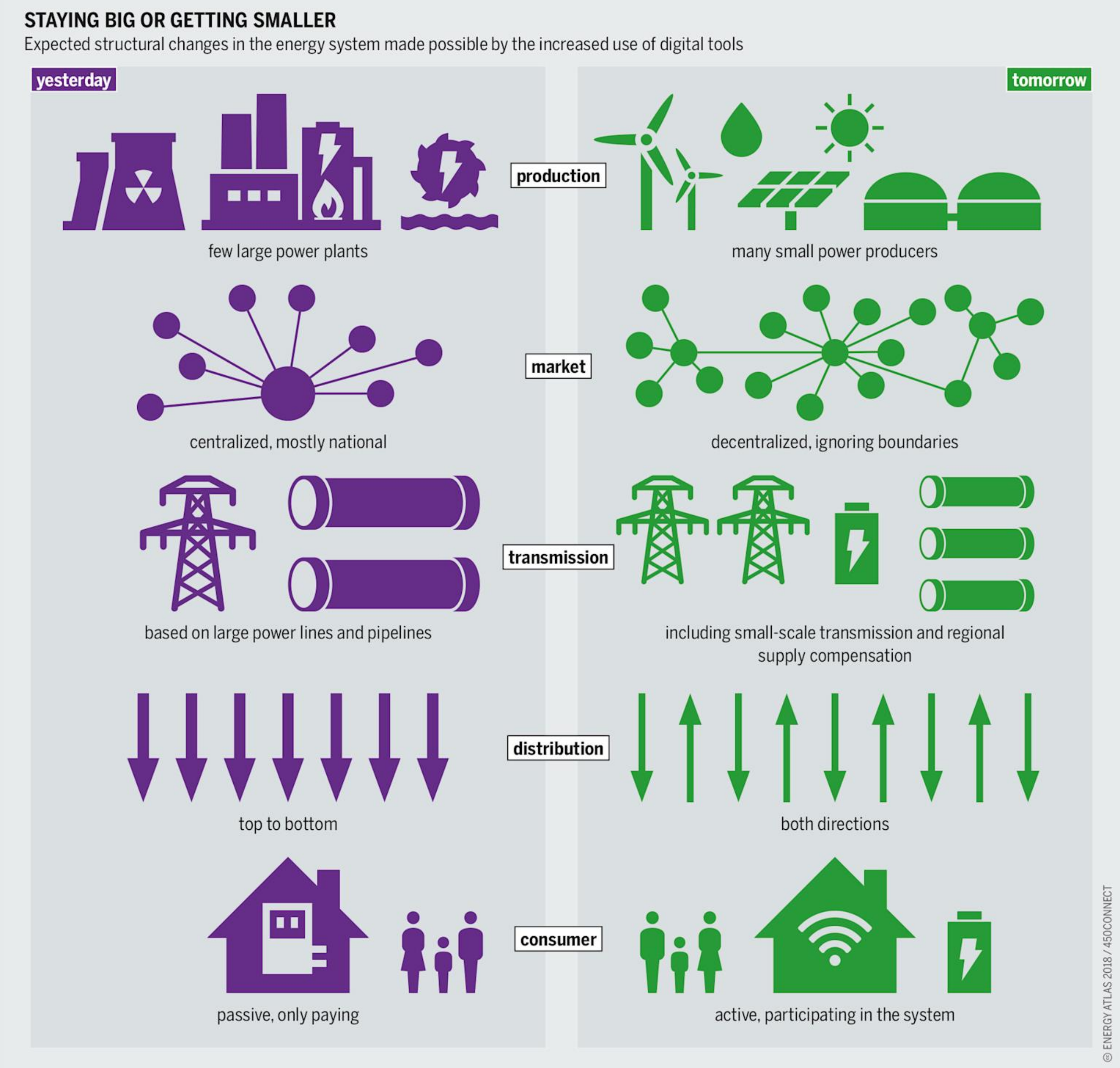
Problem & Research Gap

⚠️ The Challenge: Modern Grids & Nuclear Flexibility

Nuclear Power Plants (PWRs) are traditionally baseload providers, optimized for steady output.
However, the rise of **variable renewables** (solar, wind) causes significant grid demand fluctuations (20-50% swings).
PWRs now need **load-following capabilities** to adjust power output dynamically (e.g., 50-100% capacity) and maintain grid stability (frequency around 60 Hz).



Future Grid Architecture



The Adaptive Governor

AI-Driven Control for Nuclear Safety & Efficiency

1. The Challenge

The "PID Problem"

Traditional governors can't keep up with modern grid demands.

Nonlinear Dynamics: Struggle with rapid power changes.

Slow/Overshoot: Risks grid instability and component safety.

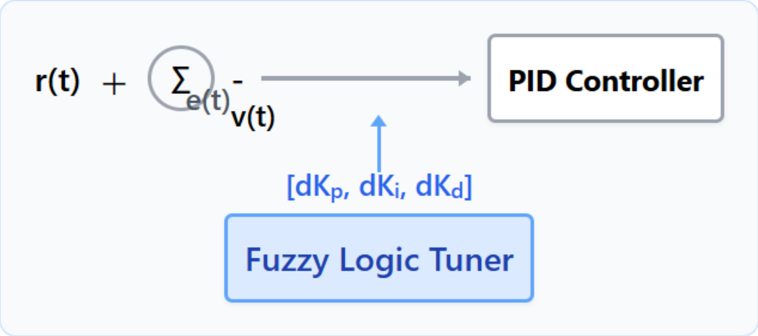
Fixed Tuning: Not adaptive to a wide range of load conditions.

The Research Gap

No existing work combines **Reinforcement Learning (RL)** with **Fuzzy Reward Functions** for this critical application.

2. The Innovation

A novel control system that learns and adapts by combining two AI agents.



1. Reinforcement Learning (RL)

An AI "pilot" that learns the **optimal strategy** to control the steam valve through trial-and-error in a safe simulation.

2. Fuzzy Reward Functions

A "smart co-pilot" that gives the RL agent flexible, linguistic goals (e.g., "IF safety is **high** AND efficiency is **good**...").

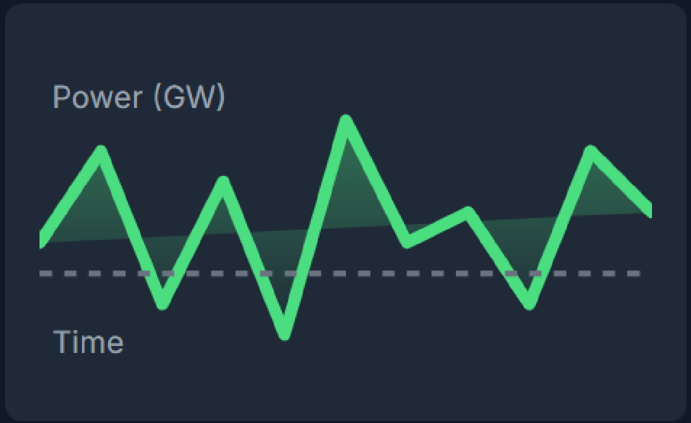
3. The Impact

Core Goal & Benefits

To develop an **adaptive, safe, and efficient** turbine governor for PWRs in modern energy grids.

- **Adaptive:** Handles complex, changing conditions without a perfect mathematical model.
- **Safe:** Manages complex trade-offs between safety, grid stability, and operational efficiency.
- **Efficient:** Optimizes performance for load following and grid synchronization.

The Core Challenge



THE DEMAND

FLEXIBILITY

Intermittent renewables demand flexible load-following from NPPs.

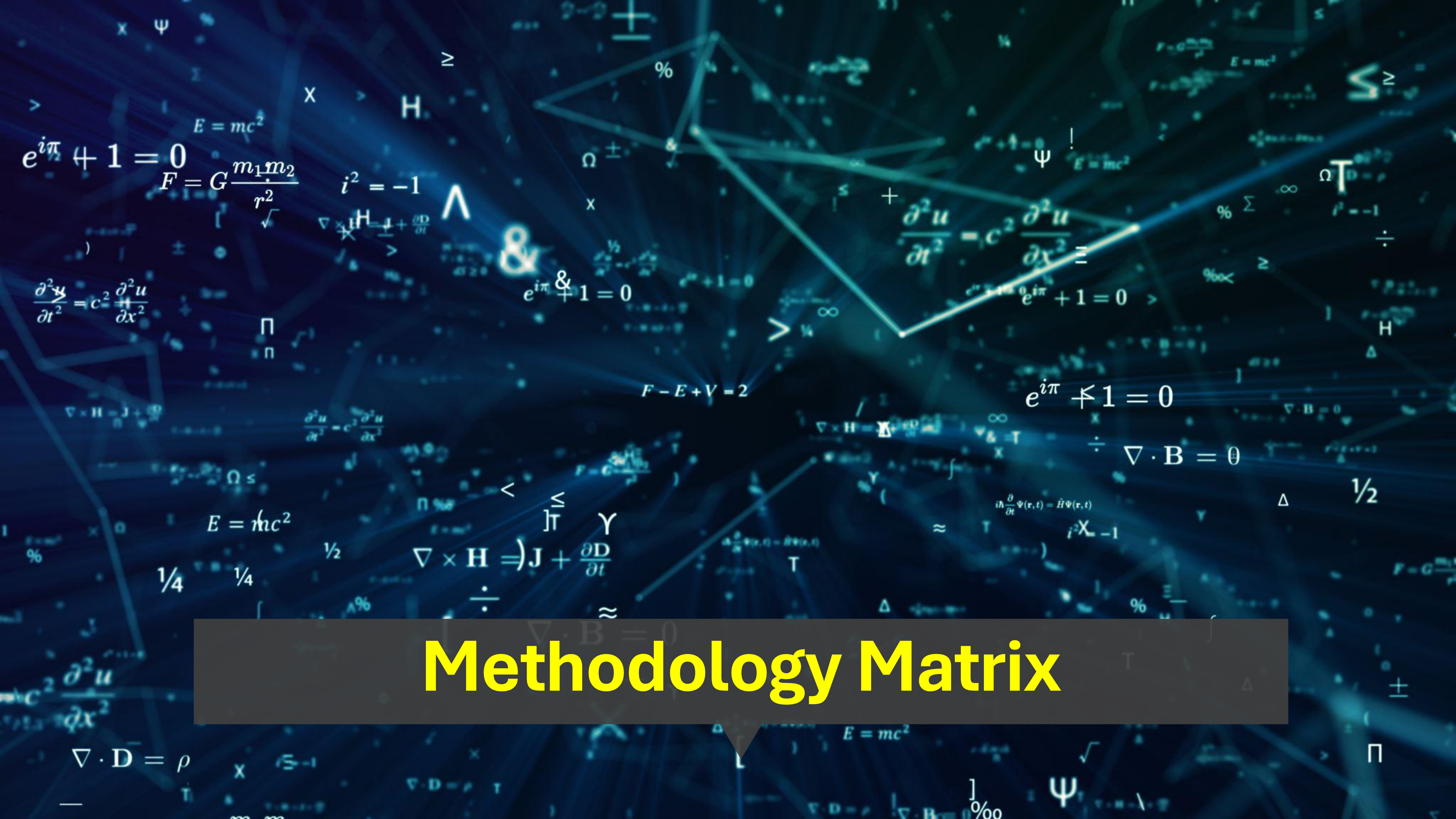
← VS →

THE MANDATE

DETERMINISM

The nuclear safety paradigm demands provable, transparent evidence.

The "Black Box" Problem: Advanced AI is flexible but opaque. Regulators do not license black boxes.



Methodology Matrix

Deterministic, Fair Comparison — Flow & Methodology

PWR software-in-the-loop · identical adversarial scenarios · licensing gates · auditable traceability

Fairness Protocol & Harness

- One high-fidelity PWR **digital model**; fixed sampling & limits
- **Predeclared scenarios**: Step · Ramp · Composite; optional noise/rate limits
- Parameters **frozen** before test; **no re-tuning** during evaluation
- **Single-run** deterministic evaluation; gates checked online
- Auxiliary fixed-seed probe is **non-licensing**

Same plant · Declared tests · Single-run · Gate-based

Evaluation Flow

Train SAC (curriculum) → Tune PID-DE / FLC-DE (equal DE budget)

Freeze policy & parameters → Run deterministic harness

TTU = 0	TSS ≤ bound
GLFI ≥ min	OS_ω ≤ limit

Licensing & Audit

- Scorecard: PASS/FAIL per gate + key metrics
- Trace logs: inputs → actions → outcomes
- Reproducible: configs, seeds, scripts

Traceability · Configs & logs

Differential Evolution — Strong Baseline Tuning

Objective: gate-aware cost aggregating licensing metrics with **hard penalties** for any gate violation and **soft penalties** for CE_sum, V_rev, OS_ω within the safe envelope.

Population = 40	Generations = 120
Mutation F = 0.8	Crossover CR = 0.9

1. Initialize population of PID/FLC parameter vectors
2. Mutate & recombine → candidate solutions
3. Evaluate closed-loop on identical scenarios (gate-aware cost)
4. Greedy select best; repeat until budget exhausted

SAC Reinforcement Learning — Training Process

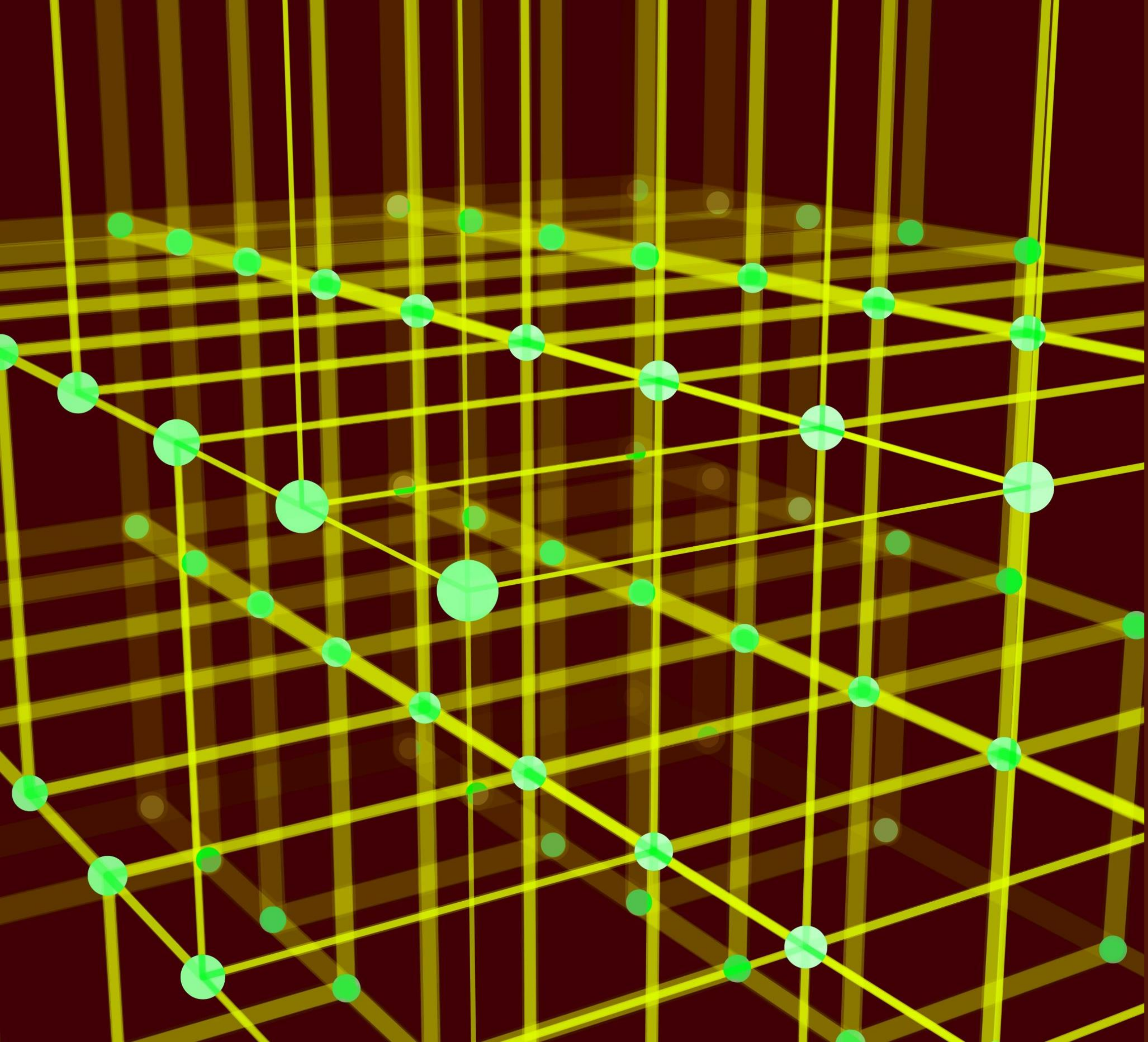
Curriculum: progressively harder scenarios; **entropy-regularized** objective; reward aligns with gates (tracking & effort).

1. Collect rollouts in the digital model (fixed physics)
2. Update actor/critic via off-policy SAC; target networks & replay
3. Validate on held-out stress tests; monitor GLFI/TTU/TSS
4. Early-stop; **freeze** policy for deterministic evaluation

Σ Entropy coef tuned · Versions locked

Deterministic protocol: licensing claims use single-run gate outcomes only; fixed-seed probe is non-licensing.

Reproducibility: scenarios, configs, seeds, logs, and scripts are released.



System Modelling

System Architecture

Control System

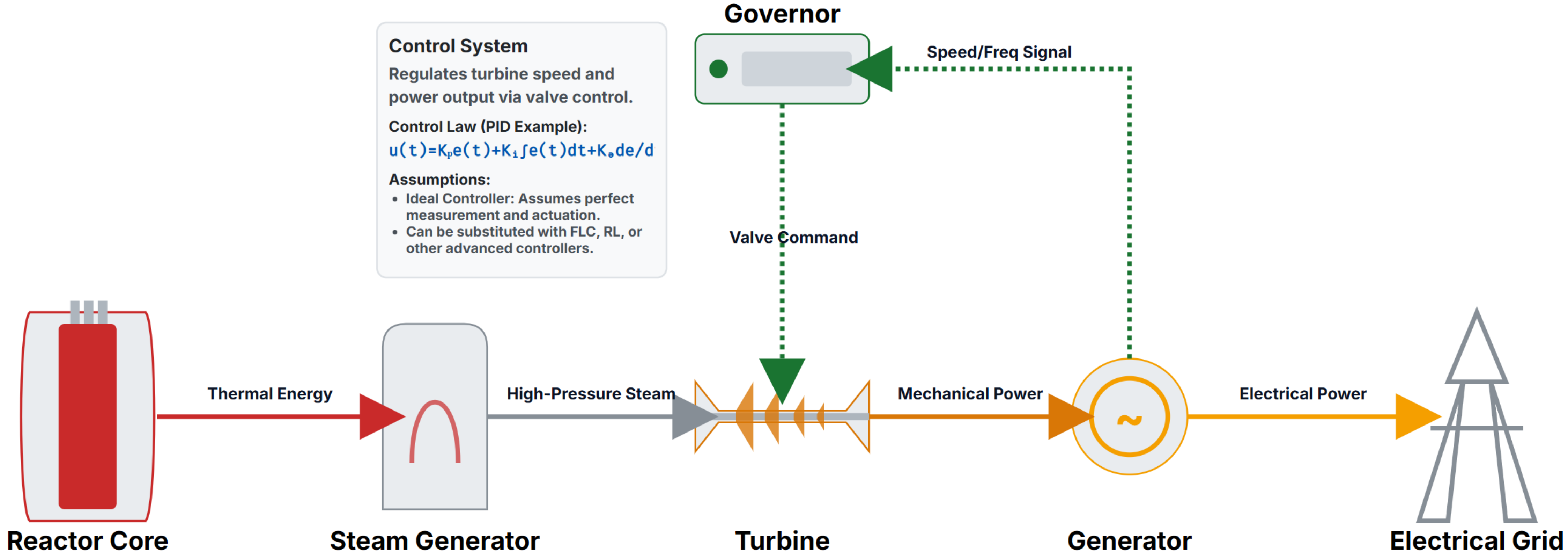
Regulates turbine speed and power output via valve control.

Control Law (PID Example):

$u(t)=K_p e(t)+K_i \int e(t)dt+K_d de/d$

Assumptions:

- Ideal Controller: Assumes perfect measurement and actuation.
- Can be substituted with FLC, RL, or other advanced controllers.



Point-Kinetics Model

Represents core-average power dynamics via fission.

Governing Equation:

$dP/dt = (\rho-\beta)/\Lambda * P + \Sigma\lambda_i C_i$

Assumptions:

- Point-Kinetics: Core is a single point, ignoring spatial effects.
- Lumped Thermal-Hydraulics: Fuel/coolant temperatures are averaged.

Lumped-Parameter Model

Abstracts heat exchange to produce high-pressure steam.

Governing Equation:

$dT_s/dt = (1/C)*(Q\_in - Q\_out)$

Assumptions:

- Lumped Model: Single average temp/pressure for the secondary loop.
- Ideal Heat Transfer: No losses in the heat exchange process.

First-Order Lag Model

Converts steam energy into mechanical rotational power.

Governing Equation:

$dP_m/dt = (P_s-P_m)/\tau_t$

Assumptions:

- Simplified Thermodynamics: Complex steam cycle abstracted to a time constant.
- Ideal Actuators: Ignores non-linear valve effects (stiction, backlash).

Swing Equation

Models electromechanical dynamics and grid interaction.

Governing Equation:

$d\omega/dt=(1/2H)*(P_m-P_e-D\Delta\omega)$

Assumptions:

- Classical Model: Swing equation is sufficient for frequency dynamics.
- Ignores sub-transient electrical phenomena.

Infinite Bus Model

Represents the external grid as an ideal power sink.

Model Definition:

$V\_grid = const; f\_grid = co$

Assumptions:

- Infinite Bus: Grid voltage and frequency are perfectly stable.
- Removes need for voltage/VAR control analysis.

Simulation Environment & Tools

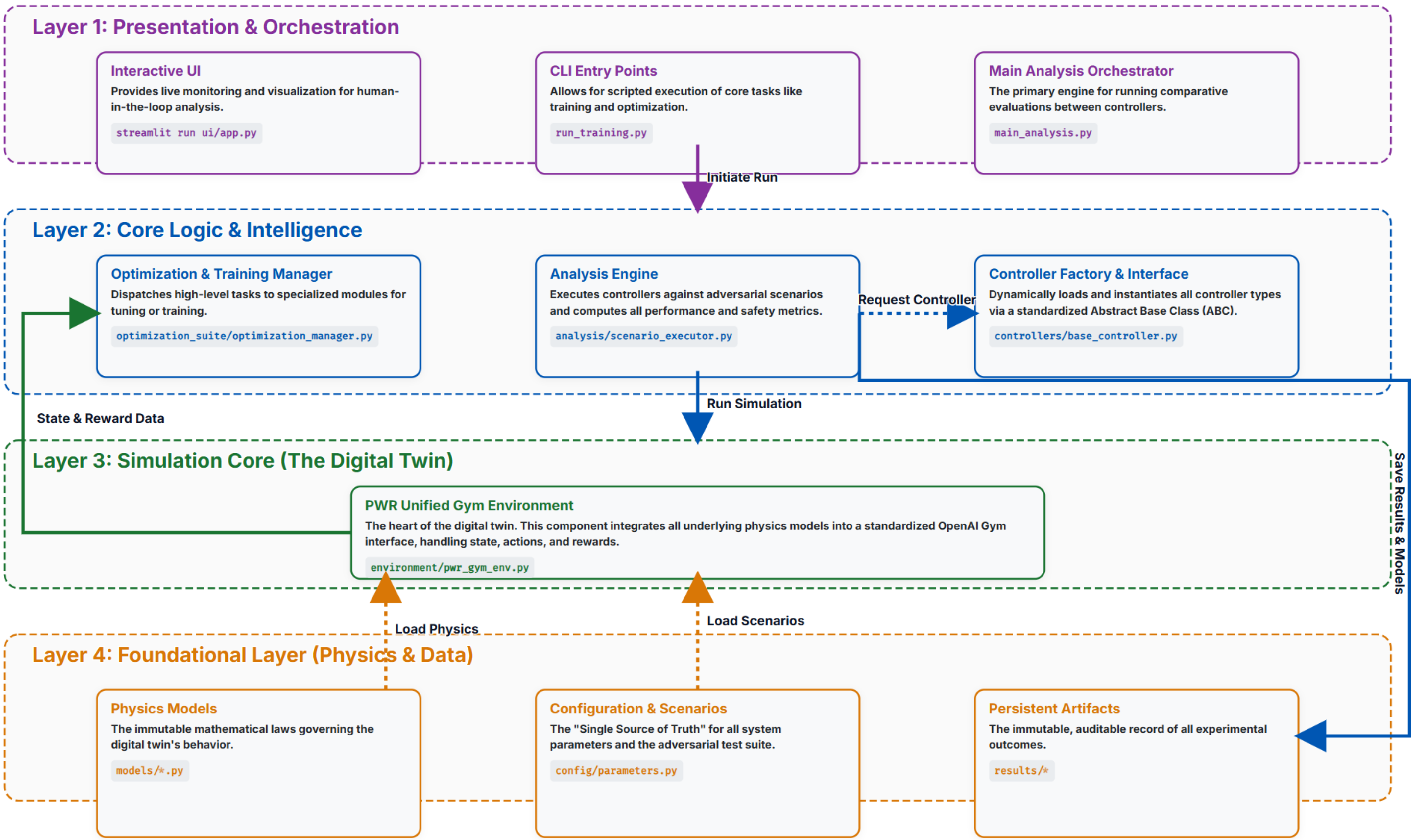
A modular simulation environment is crucial for testing and training the RL agent safely and efficiently.

Core Components:

- **Physics Engine:** Custom Python code using **NumPy** & **SciPy** for ODE solving to simulate reactor, turbine, and grid dynamics.
- **RL Framework:** **OpenAI Gym** provides a standardized environment interface for the RL agent.
- **RL Algorithm Implementation:** **Stable Baselines3** (built on **PyTorch**) used for the Constrained Soft Actor-Critic (SAC) agent training.
- **Fuzzy Logic Engine:** **scikit-fuzzy** library used to define and compute the fuzzy reward signals.
- **Visualization:** **Matplotlib** & Seaborn for plotting results and analysis.

- NumPy
- SciPy
- OpenAI Gym
- Stable Baselines3
- PyTorch
- scikit-fuzzy
- Matplotlib

System Flow


Layer 1: Presentation & Orchestration
Interactive UI
Provides live monitoring and visualization for human-in-the-loop analysis.
streamlit run ui/app.py
CLI Entry Points
Allows for scripted execution of core tasks like training and optimization.
run_training.py
Main Analysis Orchestrator
The primary engine for running comparative evaluations between controllers.
main_analysis.py
Initiate Run
Layer 2: Core Logic & Intelligence
Optimization & Training Manager
Dispatches high-level tasks to specialized modules for tuning or training.
optimization_suite/optimization_manager.py
Analysis Engine
Executes controllers against adversarial scenarios and computes all performance and safety metrics.
analysis/scenario_executor.py
Request Controller
Controller Factory & Interface
Dynamically loads and instantiates all controller types via a standardized Abstract Base Class (ABC).
controllers/base_controller.py
Run Simulation
State & Reward Data
Save Results & Models
Layer 3: Simulation Core (The Digital Twin)
PWR Unified Gym Environment
The heart of the digital twin. This component integrates all underlying physics models into a standardized OpenAI Gym interface, handling state, actions, and rewards.
environment/pwr_gym_env.py
Load Physics
Load Scenarios
Layer 4: Foundational Layer (Physics & Data)
Physics Models
The immutable mathematical laws governing the digital twin's behavior.
models/*.py
Configuration & Scenarios
The "Single Source of Truth" for all system parameters and the adversarial test suite.
config/parameters.py
Persistent Artifacts
The immutable, auditable record of all experimental outcomes.
results/*

System Architecture & Data Flow

The proposed system integrates the RL agent and Fuzzy Reward logic with the core plant components.

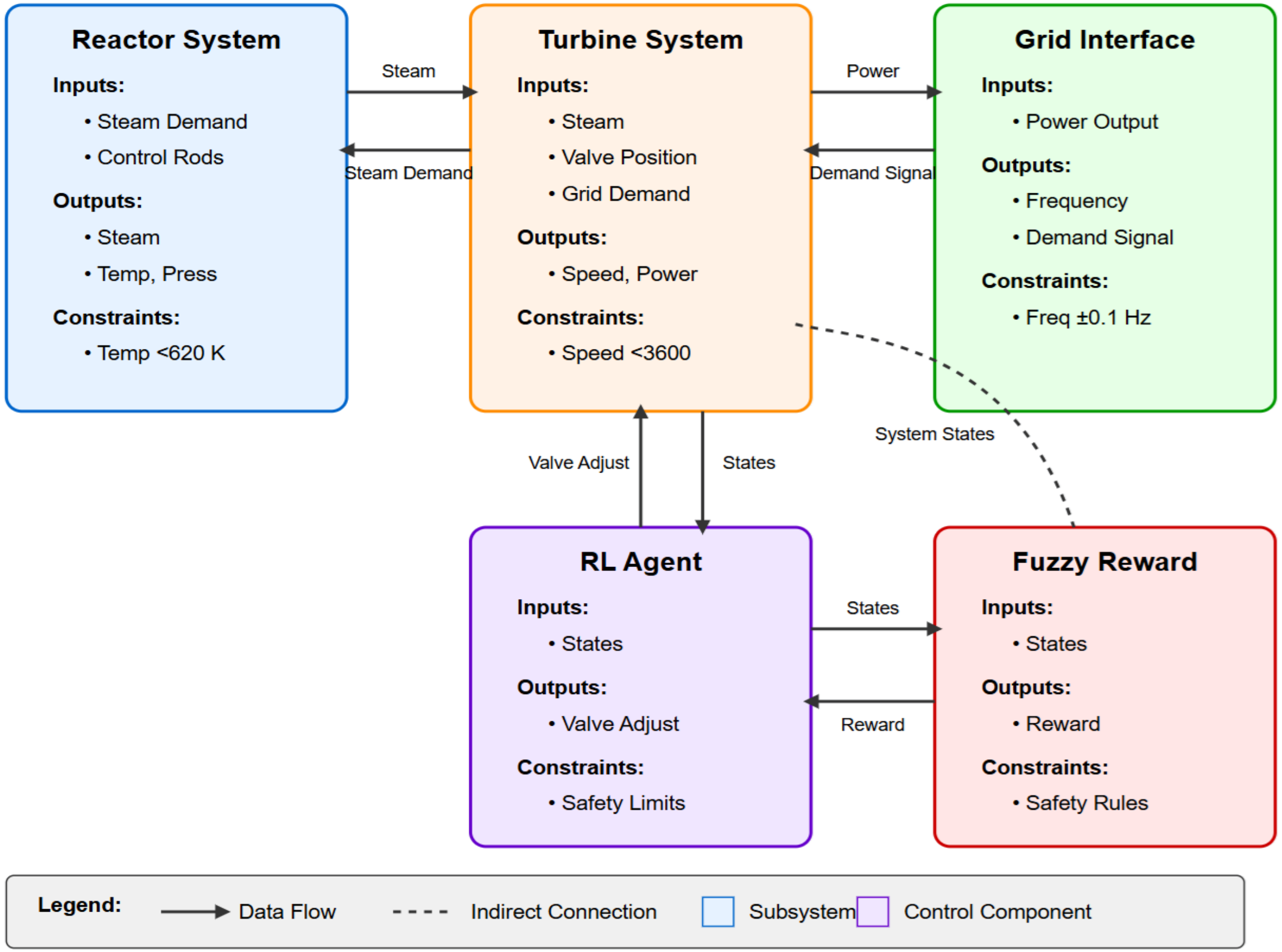
Reactor System: Models PWR core physics (point kinetics, thermal feedback). Outputs: Steam conditions, Power level. Inputs: Control rods, Demand. Constraints: Temp < 2800°C, Pressure < 15.5 MPa.

Turbine System: Models turbine/generator dynamics (lumped parameter). Outputs: Electrical Power, Speed. Inputs: Steam flow, Valve position. Constraints: Speed < 3600 RPM.

Grid Interface: Models grid frequency dynamics (swing equation). Outputs: Frequency, Voltage (simplified). Inputs: Power from turbine, Load demand. Constraints: Freq. deviation ±0.5 Hz.

RL Agent (SAC): Learns optimal valve control policy via trial-and-error in simulation. Inputs: System states (power, speed, freq, etc.). Outputs: Valve position adjustments. Goal: Maximize long-term fuzzy reward.

Fuzzy Reward System: Calculates reward based on fuzzy rules evaluating safety, stability, and efficiency. Inputs: System states. Outputs: Scalar reward signal for RL Agent. Balances competing objectives.



The Agent C/Cs

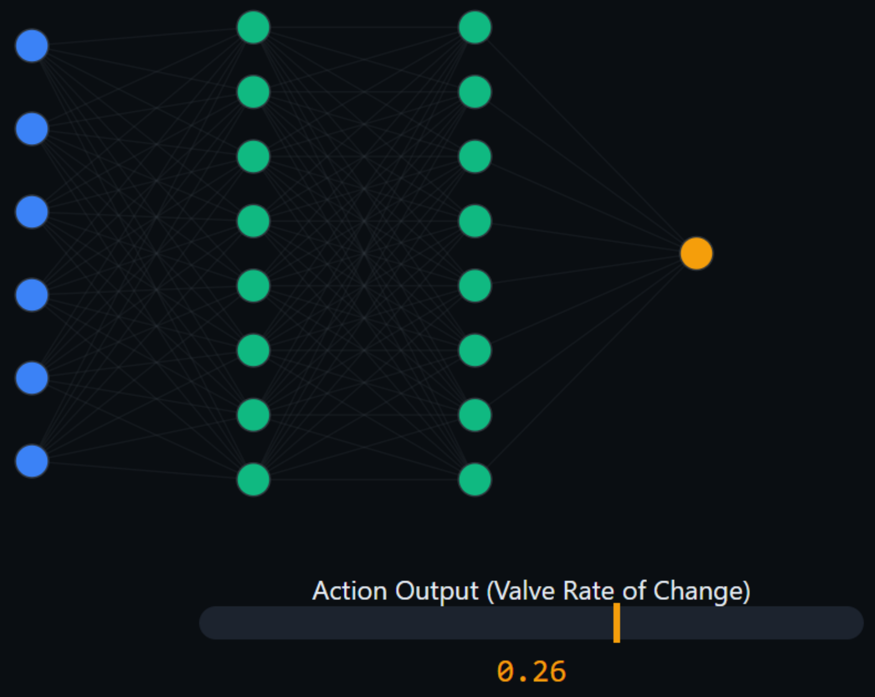
Observation Space: The Agent's Cockpit

The agent's informational advantage stems from its observation of a comprehensive, normalized 6-dimensional state vector. This dashboard visualizes the agent's "senses" in real-time. **Source:** `environment/pwr_gym_env.py`, `_get_obs()` method.

Action Space & Network Architecture

Action Space: The agent outputs a single continuous value $a_t \in [-1.0, 1.0]$, representing the **rate of change** of the governor valve position. This delta-based action space promotes smoother control signals.

Network Architecture: The agent's "brain" is a Multi-Layer Perceptron (MLP). The visualization below shows how the 6D state vector is processed through two hidden layers to produce the final action.

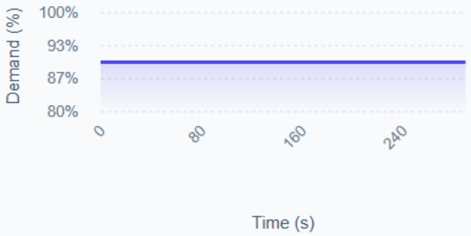


Framework Scenarios

Baseline Steady State

Objective: Test stability at a constant 90% power demand.

Reflects: Normal, stable baseload operation.



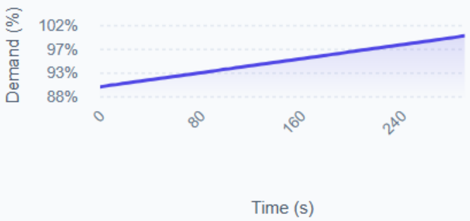
Key Metrics:

Stability | Control Effort

Gradual Load Increase

Objective: Evaluate load-following as demand ramps from 90% to 100% over 300s.

Reflects: Typical "load-following" to meet rising grid demand.



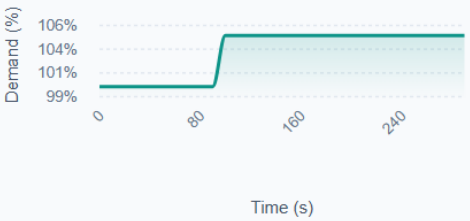
Key Metrics:

Tracking | Smoothness

Sudden Load Increase

Objective: Measure transient response to an instantaneous +5% step in load demand.

Reflects: A sudden grid disturbance, requiring immediate response.



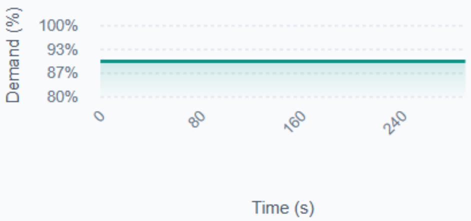
Key Metrics:

Response | Settling Time

Efficiency Probe

Objective: Enforce minimal control action during a long 1800s steady-state hold.

Reflects: Long periods of quiet operation to preserve equipment.



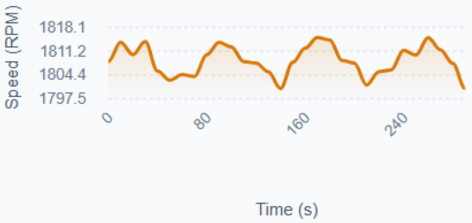
Key Metrics:

Preservation | Efficiency

Deceptive Sensor Noise

Objective: Test robustness to faulty sensor data with high noise and an 8 RPM bias.

Reflects: A sensor malfunction with imperfect, misleading data.



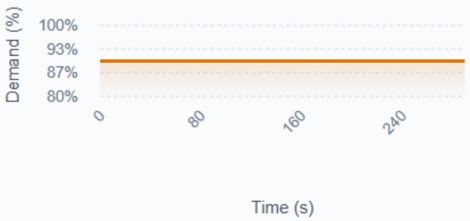
Key Metrics:

Robustness | Filtering

Parameter Randomization

Objective: Ensure generalization by varying physics params (`grid.H`, `grid.D`).

Reflects: Natural variation and uncertainty in real-world physical systems.



Key Metrics:

Generalization | Adaptability

Cascading Grid Fault

Objective: Test resilience against a multi-stage failure: 20% load rejection then recovery.

Reflects: A severe, multi-stage grid failure with compounding disturbances.

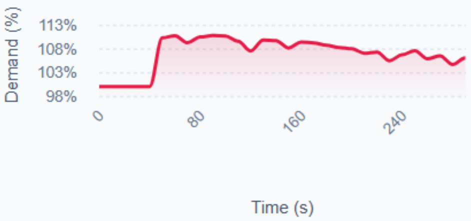
Key Metrics:

Resilience | Recovery

Combined Challenge

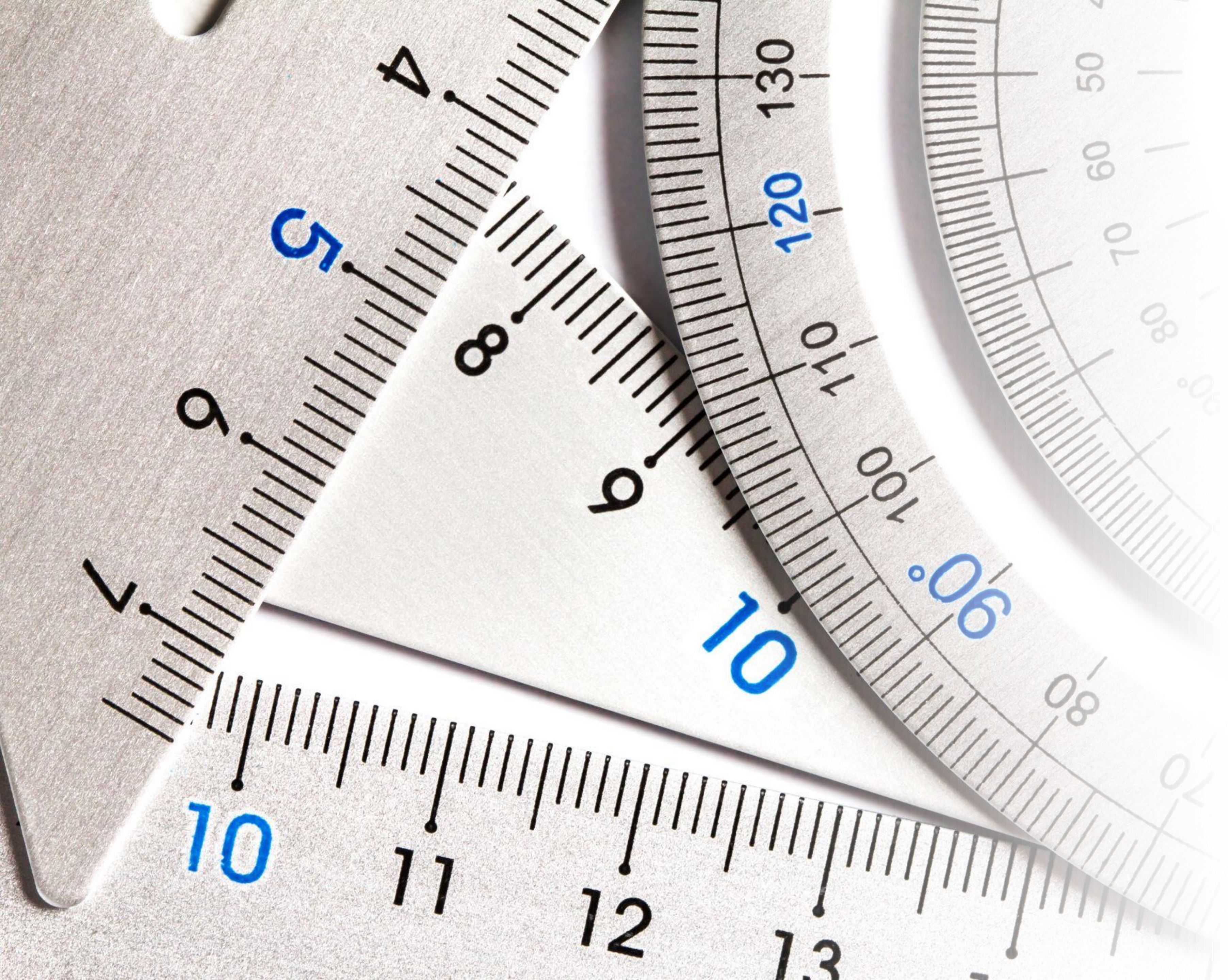
Objective: Final exam: +10% load step with sensor noise and degrading heat transfer.

Reflects: A "worst-case" day with multiple, simultaneous faults.



Key Metrics:

Overall Robustness | Adaptability



Applied Metrics

Tracking

Grid Load-Following Index (GLFI)

`grid_load_following_index`

Definition

$$\mathrm{GLFI} = \frac{1000}{1 + \mathrm{MSE}(P_{\mathrm{mech}} - P_{\mathrm{demand}})}$$

Units: dimensionless | Direction: Higher is better

Rationale: Tracks setpoint following via mean-squared power error (MSE/ISE family).

Transient

Transient Severity Score (TSS)

`transient_severity_score`

Definition

$$\mathrm{TSS} = w_1 \widehat{\Delta P}_{\max} + w_2 \hat{\zeta}^{-1} + w_3 \frac{\mathrm{TTU}}{T}$$

Units: dimensionless | Direction: Lower is better

Rationale: Composite of normalized peak deviation, damping proxy, and unsafe-time fraction.

Safety

Time Outside Frequency Limits

`time_outside_freq_limit_s`

Definition

$$T_{\mathrm{OFL}} = \int \mathbb{1}\{ f(t) \notin [f_{\min}, f_{\max}] \} \, \mathrm{d}t$$

Units: s | Direction: Lower is better

Rationale: Duration of frequency excursions beyond the regulatory band.

Safety

Total Time Unsafe (TTU)

`total_time_unsafe_s`

Definition

$$\mathrm{TTU} = T_{\mathrm{fuel}} + T_{\omega} + T_{\mathrm{speed}}$$

Units: s | Direction: Lower is better

Rationale: Aggregate of time over fuel-temperature, outside-frequency, and over-speed thresholds.

Tracking

Integral of Absolute Frequency Error (IAE)

`iae_freq_hz_s`

Definition

$$\mathrm{IAE}_f = \int | f(t) - f^\star | \, \mathrm{d}t$$

Units: Hz·s | Direction: Lower is better

Rationale: Classical performance index emphasizing total absolute error over time.

Tracking

Integral of Squared Frequency Error (ISE)

`ise_freq_hz_s`

Definition

$$\mathrm{ISE}_f = \int (f(t) - f^\star)^2 \, \mathrm{d}t$$

Units: Hz^{2}\!\cdot\!s | Direction: Lower is better

Rationale: Penalizes larger errors; standard control objective.

Transient

Maximum Frequency Deviation

`max_freq_deviation_hz`

Definition

$$\max_t | f(t) - f^\star |$$

Units: Hz | Direction: Lower is better

Rationale: Peak excursion from nominal frequency during transients.

Transient

Maximum Overshoot (Speed)

`max_overshoot_speed_pct`

Definition

$$100 \times \frac{\max_t (n(t) - n^\star)}{n^\star}$$

Units: \% | Direction: Lower is better

Rationale: Canonical transient metric; high overshoot indicates poor damping.

Effort

Control Effort (Squared Increments)

`control_effort_valve_sq_sum`

Definition

$$\sum_t (\Delta u_t)^2$$

Units: arb. | Direction: Lower is better

Rationale: Proxy for actuator wear/energy; minimizing effort extends valve life.

Effort

Valve Reversals (Chattering)

`valve_reversals`

Definition

$$\sum_t \mathbb{1}\{\mathrm{sign}(\Delta u_t) \neq \mathrm{sign}(\Delta u_{t-1})\}$$

Units: count/run | Direction: Lower is better

Rationale: Measures chattering/limit cycling; fewer reversals ⇒ smoother control.

Safety

Maximum Fuel Temperature

`max_fuel_temp_c`

Definition

$$\max_t T_{\mathrm{fuel}}(t)$$

Units: °C | Direction: Lower is better

Rationale: Thermal safety margin; exceeding limits risks material damage.

Learning

Policy Entropy (SAC diagnostic)

`control_policy_entropy`

Definition

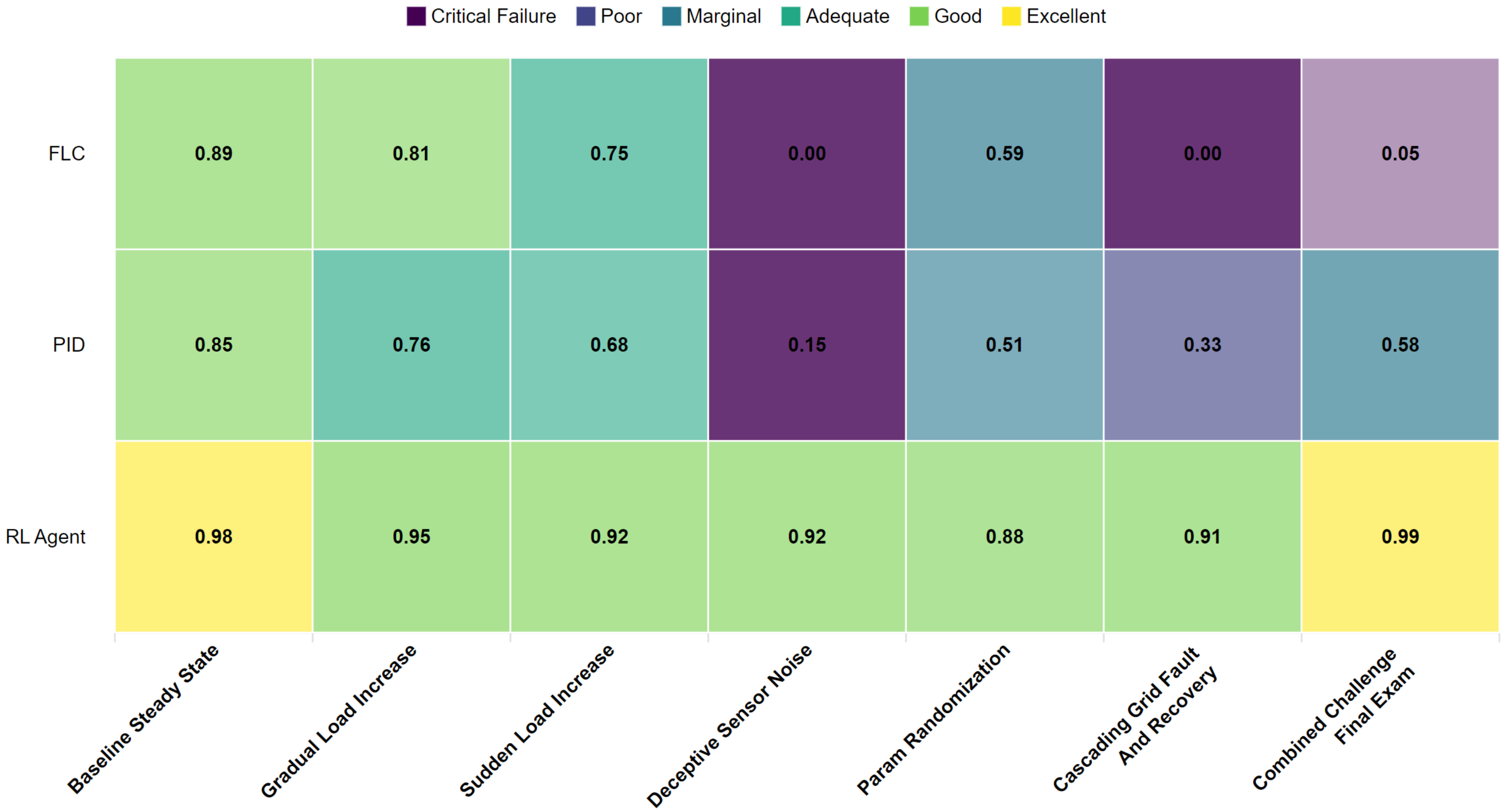
$$H(\pi) = -\mathbb{E}_{a \sim \pi(\cdot|s)}[\log \pi(a|s)]$$

Units: nats | Direction: Diagnostic

Rationale: Exploration/regularization diagnostic for SAC runs.


RESULTS

Performance Heatmap



The head-to-head evaluation was conducted across the full suite of eight adversarial scenarios. This Chart presents a comprehensive heatmap of the **primary outcome metric**, the ***Composite Robustness Score (CRS),*** which provides a definitive, high-level verdict. The results reveal a clear performance hierarchy.

- ✓ The SAC agent maintained a deep green (high performance) profile across all conditions, including the ***most severe*** tests.
- ✓ The Strong Benchmark PID performed adequately in nominal scenarios but showed significant degradation (yellow to orange) under adversarial conditions, such as Sensor Noise and Grid Fault.
- ✓ The FLC failed catastrophically in three of the eight scenarios, indicated by the dark red cells, rendering it unsuitable for this application.

Training Process Overlook

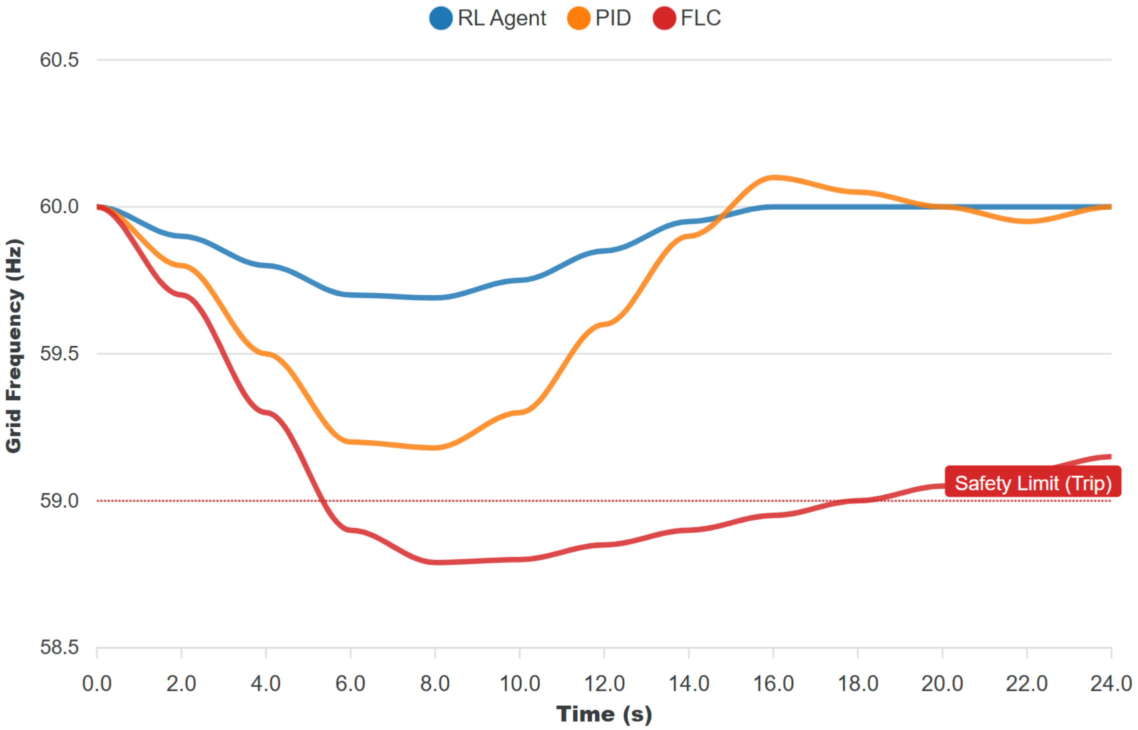
This plot visualizes the agent's skill acquisition. The **general upward trend in reward** shows learning.

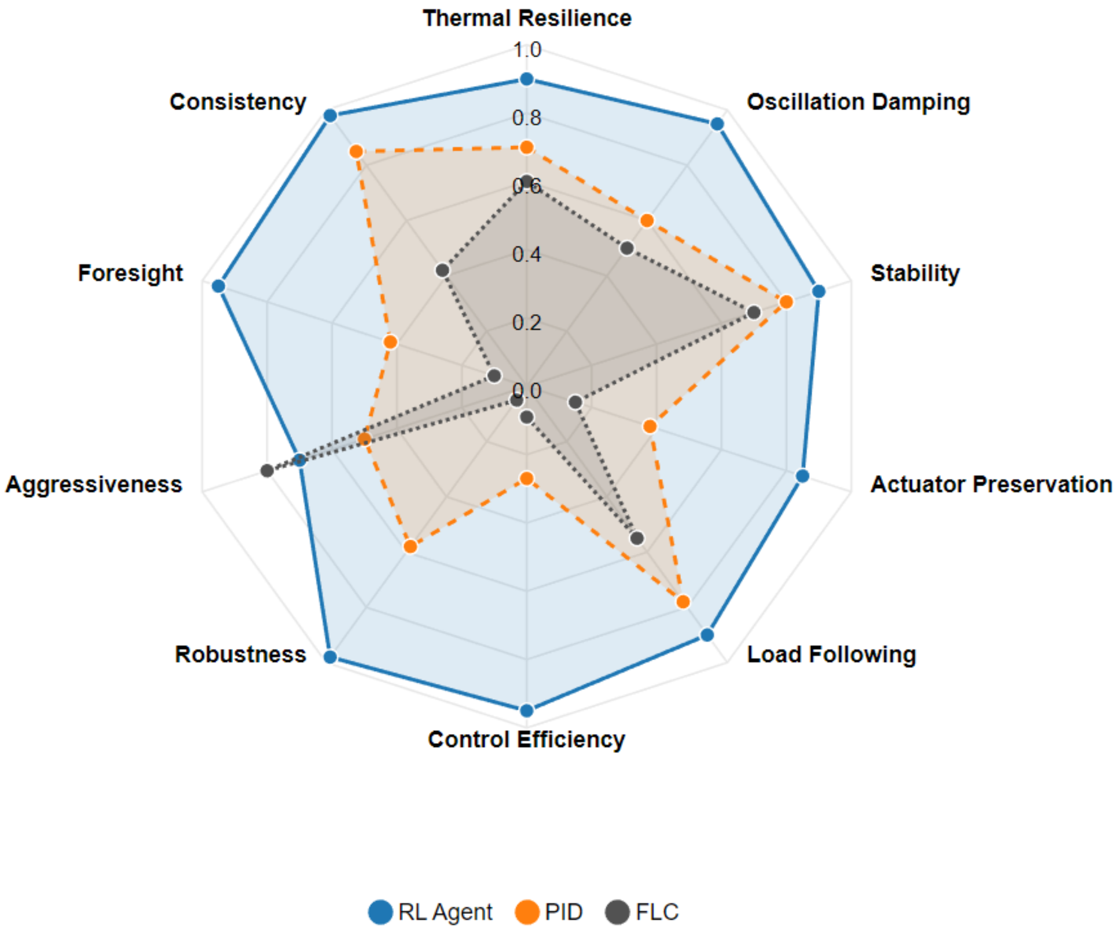
The **sharp spike** in the Performance-to-Effort Ratio (green) during Phase 2 provides direct evidence that the curriculum successfully forced the agent to master the specific skill of control efficiency, proving its final policy is a result of deliberate, structured pedagogy.

The Scenario-Based Crossover Analysis

The efficiency of the **PID and FLC controllers collapses as the scenario stress increases**, whereas the SAC agent's efficiency remains high. This plot forensically identifies the boundary where an adaptive approach becomes essential.

Integrated Performance Analysis

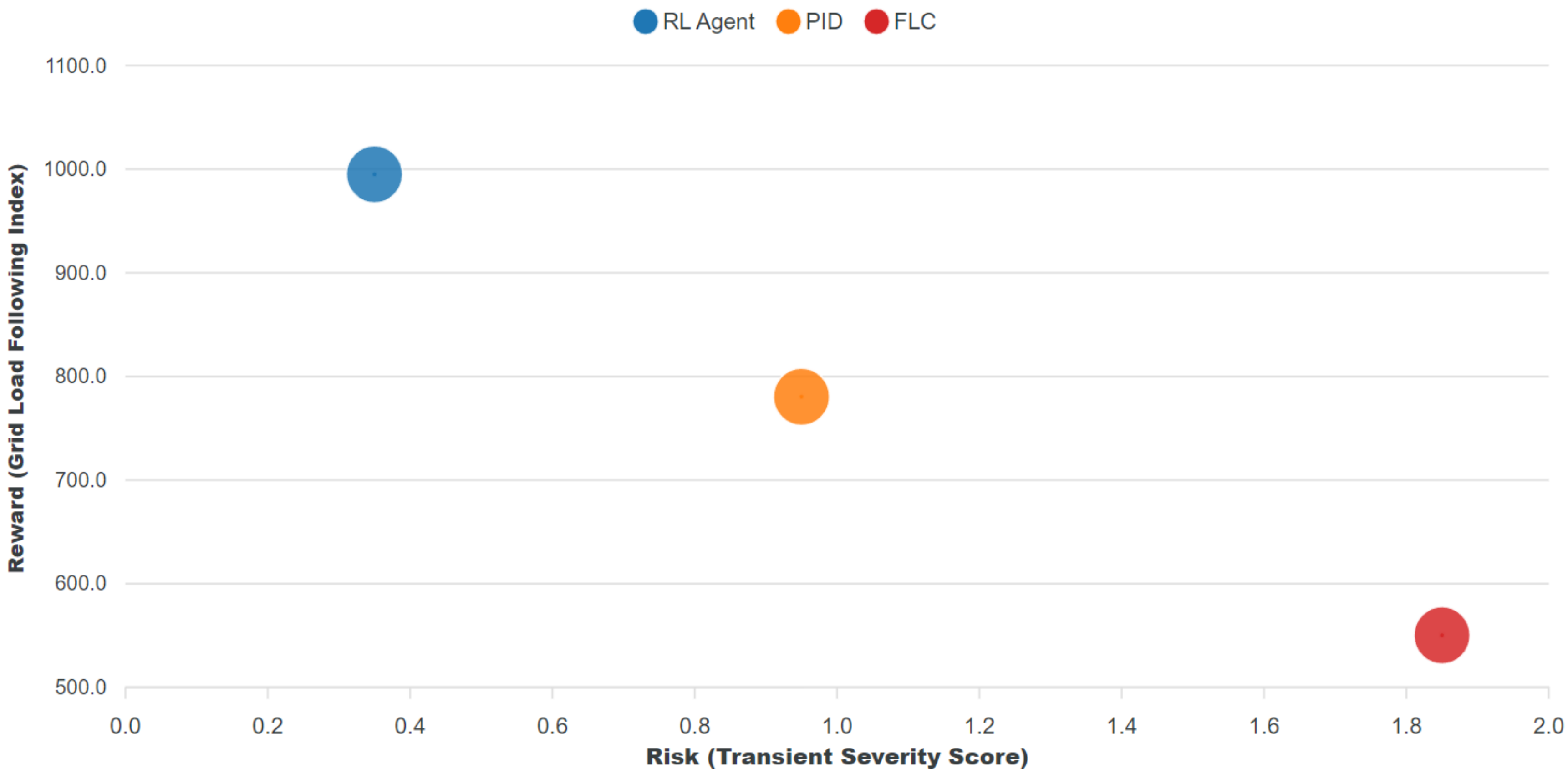




To provide a deeper, qualitative understanding of these aggregate scores, a forensic analysis was conducted on the most demanding scenario that all controllers managed to complete: the ***cascading_grid_fault_and_recovery***. The time-series response of the grid frequency which is a critical system state variable.

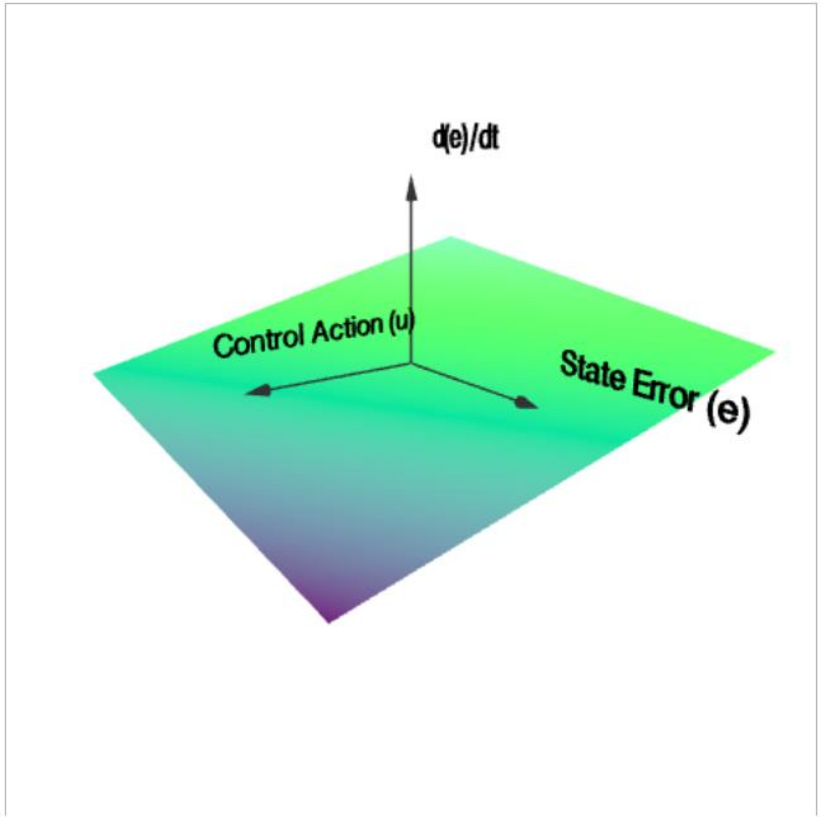
This the multi-attribute profile gives rise to distinct controller ***"personalities,"***

Load following Risk during Transient

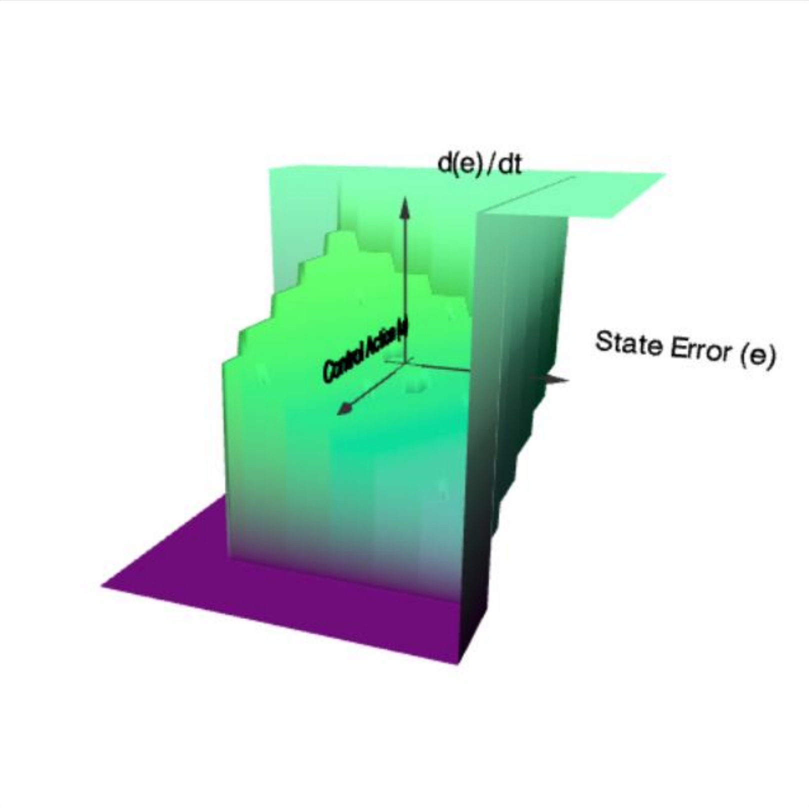


- ❑ **This plot illustrates the risk-reward trade-offs. The ideal controller is in the top-left (low risk, high reward).**
- ❑ **The RL Agent clearly operates on the optimal Pareto frontier, achieving the best possible outcome. The classical controllers are in a suboptimal region.**

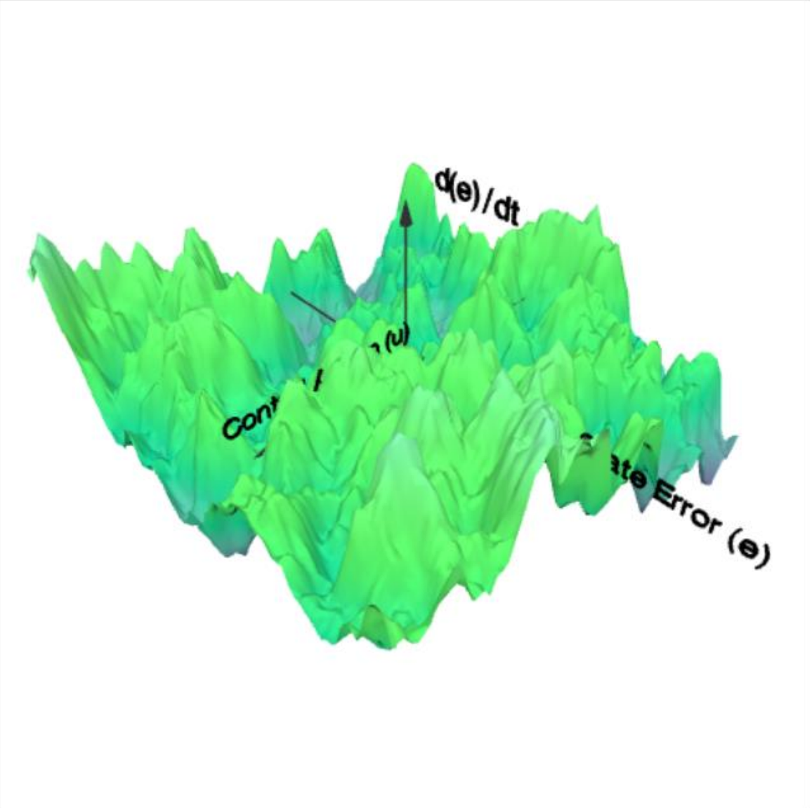
Policy Manifold



(a) PID Policy



(b) FLC Policy

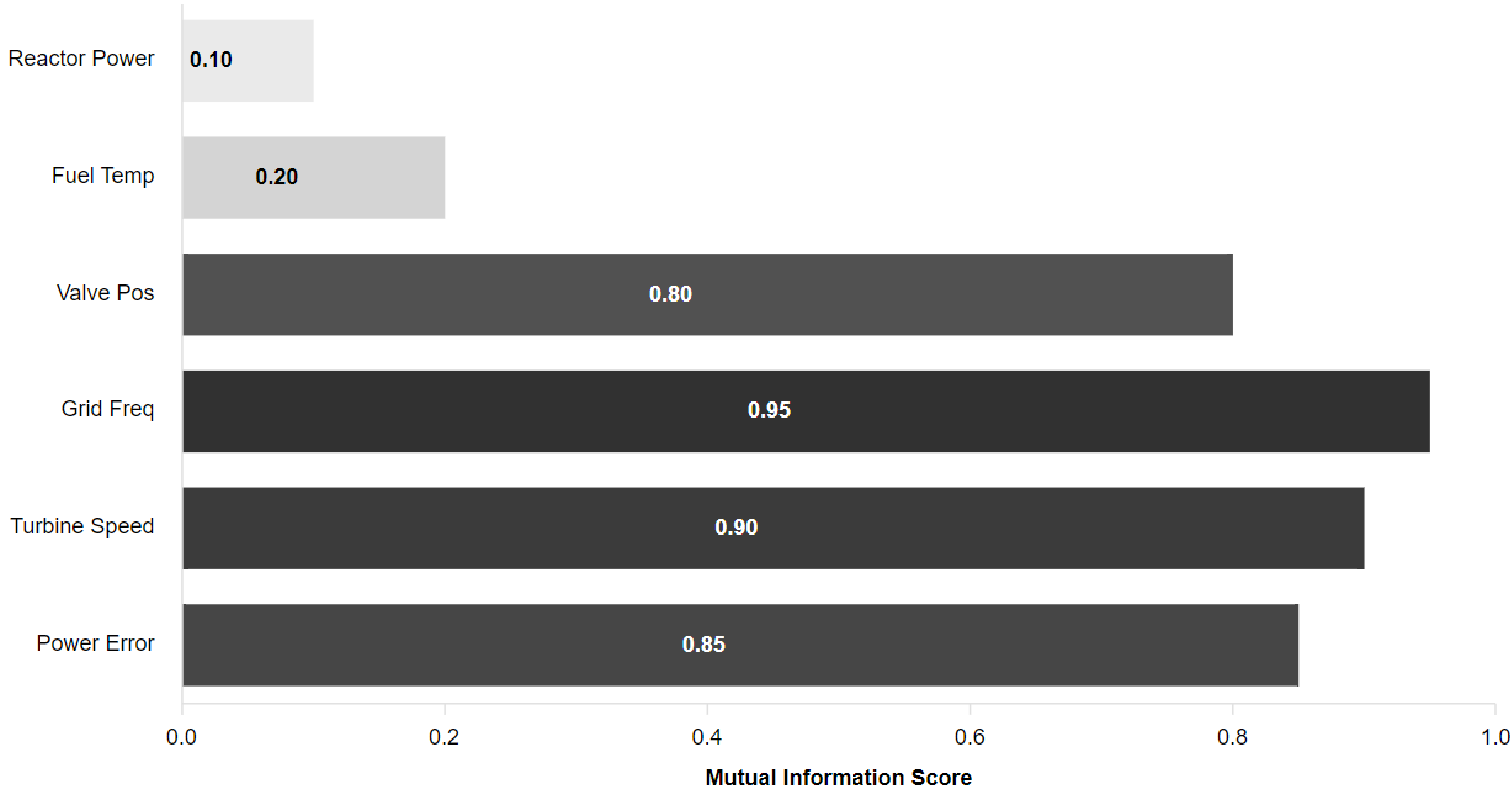


(c) RL Agent Policy

✓The foundational reason for the SAC agent's superior performance lies in **the distinction between a controller's low-dimensional parameter space and its high-dimensional policy space.**

✓A classical PID controller, even when globally optimized via Differential Evolution, is ultimately defined by ***a simple linear plane*** in the state-action space,

States Interrelation-ships



Conclusion & Future Work

A New Paradigm for AI Assurance in Critical Systems

The Verdict: Quantitative Dominance

25%

Reduction in Control Effort vs. Optimized PID

22%

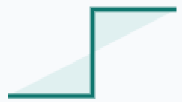
Superior Composite Robustness Score ($p < 0.001$)

The Strategic Insight: Architectural Superiority

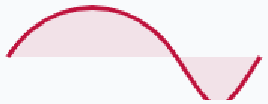
The Policy Manifold

Classical controllers optimize a **simple policy structure**. The SAC agent discovers a fundamentally **superior, high-dimensional policy manifold**.

PID: Linear Plane



FLC: Piecewise-Linear



SAC Agent: Learned Non-Linear Manifold

Scientific & Practical Impact

Innovation

First application of safety-certifiable RL with fuzzy rewards for PWR governor control.

Enhanced Safety

Explicit incorporation of safety constraints aligned with nuclear standards (IAEA, IEEE).

Improved Grid Integration

Enables PWRs to participate more effectively in grids with high renewable penetration.

Impact Pathways for the Nuclear Community:

- Potential framework for **AI certification** in safety-critical systems.
- Supports **operational flexibility** and modernization efforts in existing and future plants.
- Provides a **research benchmark** for academic and industrial R&D.

Future Work & Potential Enhancements:

01

Advanced Uncertainty Quantification:

Incorporate deeper Monte Carlo simulations to assess robustness against parameter uncertainties.

02

Formal Safety Verification:

Apply formal methods or Probabilistic Risk Assessment (PRA) techniques for deeper safety analysis (beyond current scope).

03

Hardware-in-the-Loop (HIL) Testing: Bridge the gap between simulation and reality by testing the controller with real hardware components.

04

Expanded Scenarios:

Include more complex grid events or cyber-physical attack scenarios.

05

Explainable AI (XAI):

Develop methods to better understand the RL agent's decision-making process.

Thank You!

Questions?

ahmedrahman299@gmail.com