PSAM 2026 | 2026.07.23

Generating Daily Load-Following Operation Scenarios by using Pre-trained Reinforcement Learning for Nuclear Power Plants

Junhyeong Bang
Jonghyun Kim*

Nuclear I&C and Autonomous Operation (NICA) Lab.
Department of Nuclear and Quantum Engineering, KAIST

Content

01

Introduction

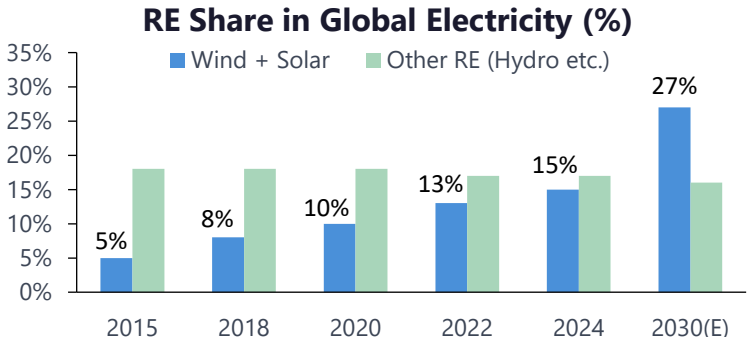
Background, motivation, and literature survey

Demand for Flexible Operation Driven by Renewable Energy

Increasing wind-solar penetration intensifies **grid supply-demand imbalance**, requiring flexible output modulation of conventionally baseload nuclear power plants (NPPs)

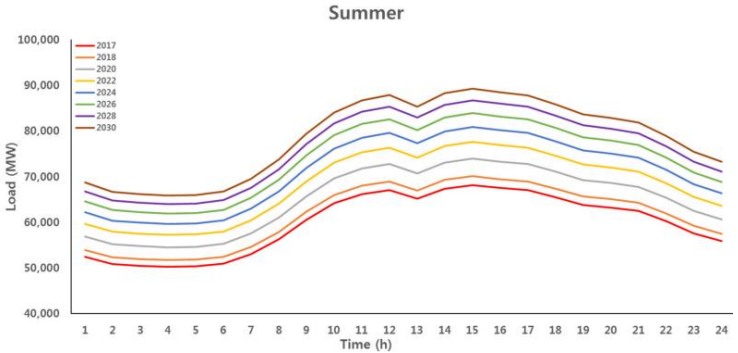
1 Rapid Increase in VRE Share

2030 wind+solar share projected at 27%, grid variability intensifying [1]



2 Domestic Grid Load Case

Annual widening of summer peak-to-through load gap; nuclear operation across 37~93% range unavoidable [2]



Transition required from baseload to flexible load-following opeartion

[1] IEA. (2025). Renewables 2025. International Energy Agency. [2] Kim et al. (2020). Analysis of challenges due to changes in net load curve in South Korea by integrating DERs. Electronics, 9(8), Article 1310.

Overview of Flexible Operation

- Flexible operation adjusts NPPs power output in **response to grid supply-demand** conditions
- Core reactivity is controlled via **control rods and boron concentration** adjustment to vary power output
- EUR and EPRI provide guidelines for **daily load-following operation (DLFO) with a 100%-50%-100%** diurnal power cycle
- However, dure to the nonlinear and coupled physical interactions within the core, **reactivity control remains challenging**

European Utility Requirements (EUR) Recommendations	• 100%-50%-100% • DLFO • 5 cycles per week • 200 cycles per year
EPRI Recommendations	• Ramp-down from 100% to low power within 2 h • Maintain low power for 2-10 h • Ramp-up from low power to 100% within 2 h • Hold 100% power for the remainder

	Mode A	Mode G	Mode X
Started	1970	End of 1970	1980
Variables	Boron Concentration	Mode A + Control Rod	Mode G + Axial Power Offset
Limitations	Very Slow Reaction	Core Power Imbalance	

French load-following case

Safety Guideline for DLFO

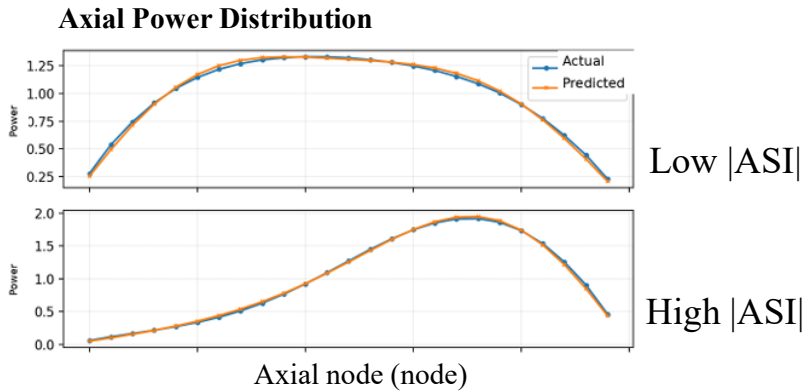
- **Guideline for DLFO**: DLFO must perform under the safety guideline such as **Axial Shape Index (ASI)**
- **Definition of ASI**: The index of imbalance of **axial power distribution** given in Eq. 1
- **The operating safety condition**: **|ASI| < 0.3** (which is |ASI| = 0.3 means ratio of bot and top power almost twice
  - Such large axial power imbalance reduces the safety margin such as DNBR and induces Xe-induced power oscillation

Eq 1. Definition of ASI

$$\mathrm{ASI} = \frac{P_{\mathrm{bot}} - P_{\mathrm{top}}}{P_{\mathrm{bot}} + P_{\mathrm{top}}}, \qquad P_{\mathrm{bot}} = \int_0^{H/2} P dz \, , P_{\mathrm{top}} = \int_{H/2}^{H} P dz$$

Eq 2. Power Imbalance at ASI = 0.3

$$\mathrm{ASI} = 0.3 \Rightarrow \frac{13}{7} \approx 2 = \frac{P_{\mathrm{bot}}}{P_{\mathrm{top}}}$$



DNBR: Departure from Nucleate Boiling Ratio

Operational Challenges of DLFO

- **Control methods:** Control Element Assemblies (CEAs) & Boron concentration
- **Challenges**: Power changes **induce the Xe-transient** and this cause non-linear ASI behaviour

Points

Xe absorb the neutron and this cause power oscillation

	Pros	Cons	
Boron Concentration	• Without affecting the ASI	• Slow reaction • Radioactive liquid wastes	→ Minimize or Boron free → i-SMRs
CEAs Maneuver	• Fast reaction	• Affecting the ASI	→ Main control variables (But within |ASI| < 0.3)

- **Efforts:** Various approaches have been explored in Korea
  - **i-SMR**: is designed to address these challenges by considering DLFO requirements from the initial design stage.
  - **Large Commercial Reactor**: 100%-80%-100% flexible-operation test has been conducted at Shin-Hanul Unit 1
  - **On-going Project:** Development of a Real-Time 3D Core Power Distribution Calculation and Operation-Support Integrated System (2025.06 – 2028.12)
    - **Objective 1:** 100-50-100 or 100-20-100 DLFO
    - **Objective 2:** 5 cycles per week, 200 cycles per year
    - **Objective 3:** Power ramp rate up to 30% per hour

    } 100% to 50% decrease [2h], sustain 50% [6h], 50%to 100% increase [2h], sustain 100% [rest]

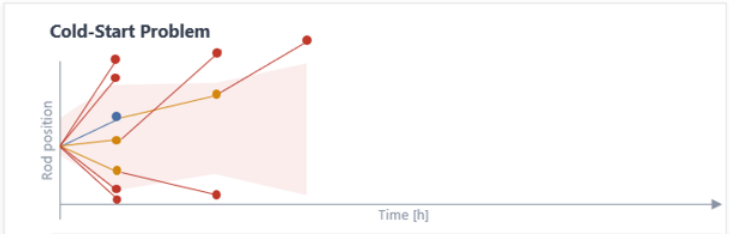
    - **Objective 4:** Generation of operation scenarios satisfying |ASI| < 0.3

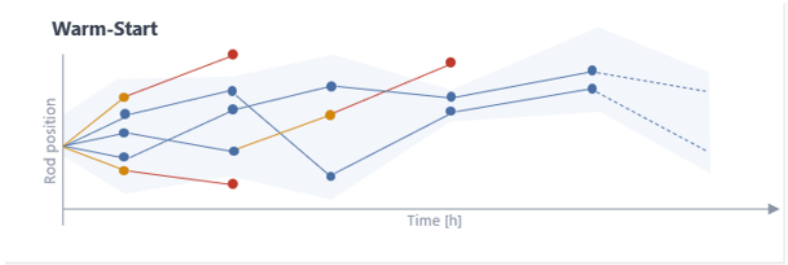
i-SMR: innovated small modular reactor

How to Solve This Problem

- **Problem:**
  - **Hard** to generate the DLFO scenarios **under the ASI** condition by the **Xe-transients**
- **Approach:**
  - With **Reinforcement Learning (RL)**, Search and generate the DLFO scenarios model itself while keeping ASI condition
  - **Problem in Approach (Cold-Start):**
    - In initial stage of training, RL conducts random search which cause the training collapse and delaying

- **Alternative (Warm-Start):** Using **pre-trained actor network** with Supervised Learning (SL)





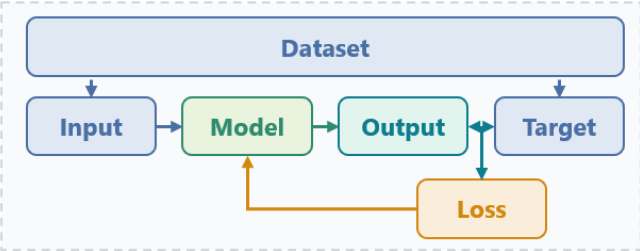
Overview of This Presentation

Objective

- Generation of load-following scenarios via **pre-trained-network-based reinforcement learning**

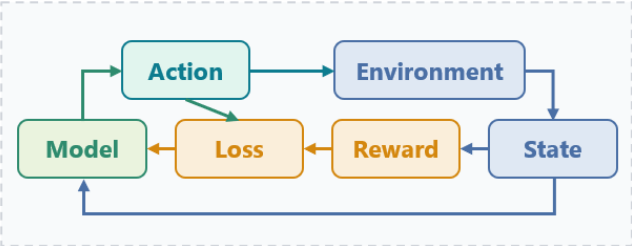
Step 1. Pre-training

- Objective of pre-training
- Design of dataset-generation algorithm
- Dataset generation for supervised learning (SL)
- Actor model configuration and training



Step 2. Reinforcement Learning (RL)

- Objective of RL
- Design of a PPO-based RL platform (coupled with core analysis code RAST-K)
- ASI-based reward function design
- Training procedure and results



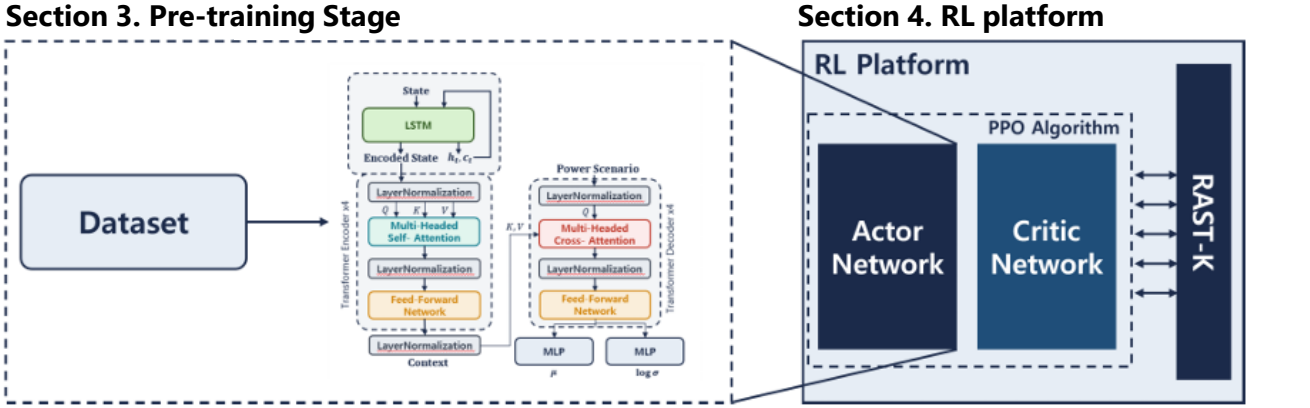
02

Method

For the feasibility of the control rod position

2.1. Proposed Framework

- **Pre-training Stage:** Perform SL to **agent network architecture** in RL platform by pre-generated dataset
  - **2.2. Methods for Agent Network Architecture:** LSTM based Transformer model
- **RL platform:** the **Proximal Policy Optimization (PPO)** agent interact the environment coupled **with RAST-K** which described in section 4. Reinforcement Learning
  - **2.3. Methods for RL platform**: PPO algorithm in RL



Methods: LSTM based Transformer model

Methods: PPO algorithm

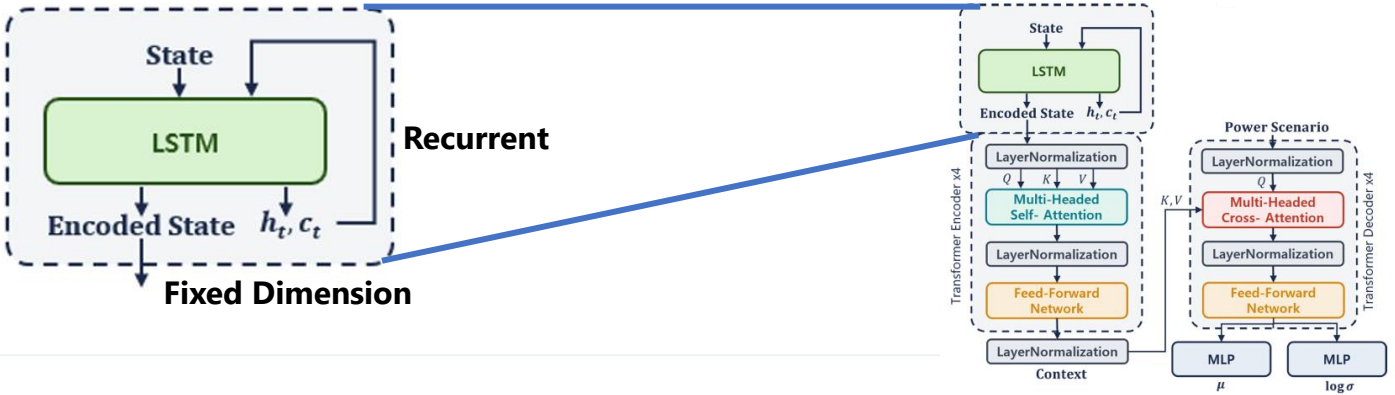
RAST-K: 3d modal the core analysis code developed in UNIST

2.2. Agent Network Architecture

- **Target of pre-training:** Actor network that **generates** action such as **CEAs speed**
  - **Component of Actor network 1**: **LSTM** which is input head to reflect the past core operating history
  - **Component of Actor network 2**: **Transformer** which extracts information and predict with attention mechanism

- **LSTM (Long-Short Term Memory):**
  - **Embedding the data history** with hidden (h_t) and cell states (c_t)
  - Connect the memory with recurrent the two states
  - Operating history is variable but the out state dimension is fixed so suggests the stable input to transformer model

- **Transformer**
  - **Encoder:** generate context with **self-attention algorithm**
    - **Self-attention:** find correlation inside of data itself
  - **Decoder:** predict results with **cross-attention algorithm**
    - **Cross-attention:** with query, find correlation from context and extract future feature



2.3. Method of Reinforcement Learning

- **Methodologies for Reinforcement Learning:**

	Methodology	Description
Value-based	Q-learning	Learning Q-matrix for discrete action • However, CEA speed is continuous variable.
Policy-based	Vanila policy gradient	Generates random action and backpropagate raw reward • Doesn't limit the magnitude of update which can cause diverge
	Trust Region Policy Optimization (TRPO)	Guarantees stable updates but requires explicit handling of the trust-region constraints
	Proximal Policy Optimization (PPO)	**Limits the size of updating** so that induce the stable training

- **In this study: we use PPO methods** in RL platform

03

Pre-training

For the feasibility of the control rod position

Pre-training Objective: Development of the Network Model for RL

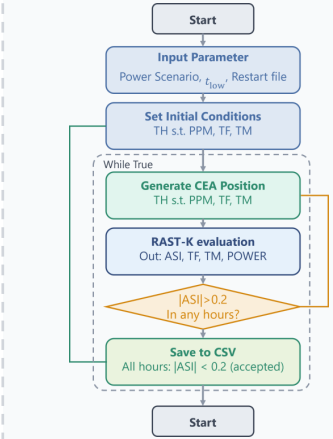
Dataset Generation

3.1 Purpose of Pre-training

- RL exploration may proceed randomly
- Random exploration is slow, unstable, and prone to training failure

3.2 Dataset Generation

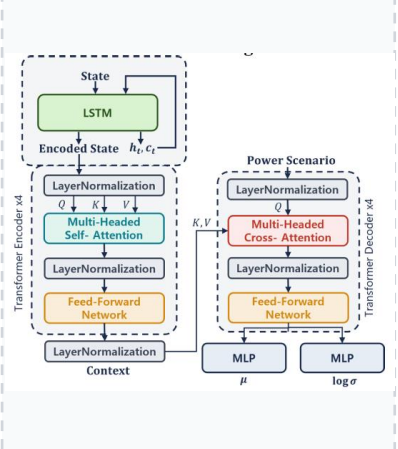
- Control rod motion constrained by operational limit conditions
- Generation of an hourly dataset spanning 1.5 years (BOC to EOC)



Pre-training Procedure

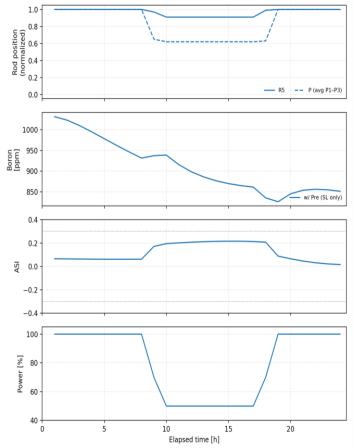
3.3 Model Architecture

- Transformer-based architecture serving as the RL actor Network
- LSTM module incorporating past operational history



3.4 Model Training

- SL with MSE loss and AdamW optimizer
- Validation MSE converges to ~0.023, satisfying pre-training objective



3.1. Purpose of Pre-training

- **Random Search Problem (Cold-Start):**
  - Random initial policy induces repeated meaningless exploration
  - Frequent constraint violations preclude effective exploration
  - **Cause time escalation and exploration may become infeasible**

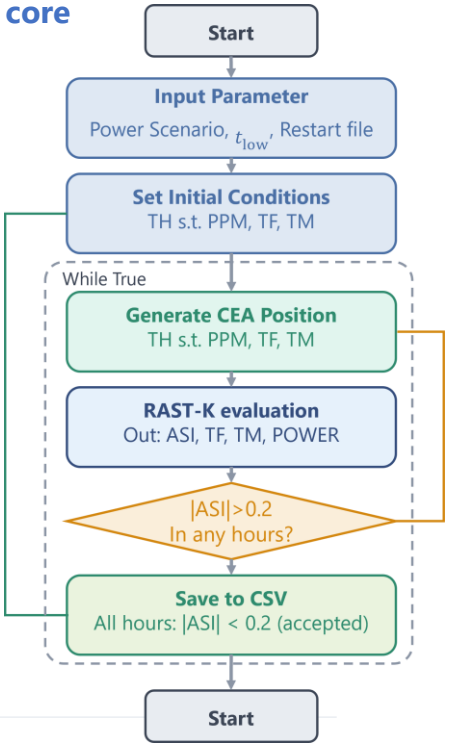
- **Regulating by Pre-training (Warm-Start):**
  - Pre-train the actor network to **regulate the learning space**
  - Reduces exploration space and provides learning directionality

Exploration Space Restricted to Generate Only Physically Meaningful Operating Scenarios

3.2. Dataset Generation

- **Algorithm Based Generation:**
  - With fixed scenario such as **100-50-100 everyday** from initial **BOC to EOC** in **APR1400 core**
  - **Step 1**: Generates CEAs position (**P, R5 bank**) with randomly
    - Convert to CEAs speed when input to the model
  - **Step 2**: evaluate ASI with RAST-K
  - **Step 3**: Determine the ASI
    - Set |ASI| < 0.2 as save condition
    - Since the **little difference of trained model may cause |ASI| threshold hitting** so that set |ASI| condition more conservatively

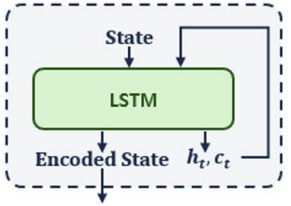
- **CEAs Position Generation:**
  - To make feasible dataset of CEA Position, we use **PDIL**, **order rule** in generation
  - **PDIL (Power Depend Insertion Limit)**:
    - PDIL **prescribes the insertion limit** as a function of power level
  - **Order rule**:
    - In Regulating Group (R5-R1), Rod inserted by the order of R5-R4-…-R1
    - The between gap of regulating group is **fixed**, all the bank positions are determined when **R5 (lead bank)** position is determined.
    - The part-strength group (P1-P3) is independent with R5, so we generate **only two bank position (R5, P)**



3.3. Model Architecture

- **Model Input Variables – Encoder (LSTM):**

Variables	Description	Normalization
Power	Core Power	Power / 100
Burnup_M/ Burnup_Z	Min-Max / Z-Score Burnup	Min-max / Z-score
PPM	Soluble Boron Concentration	Z-score
ASI	Axial Shape Index	ASI/0.3
TF_Min	Minimum Fuel Temperature	Z-score
TM_Avg/In/Out	Average/Inlet/Outlet Moderator Temperature	Z-score
R5/P	R5/P bank position	normed



$\tilde{D} \in \mathbb{R}^{24\times11} \Rightarrow \text{Linear}(\tilde{D}) = s_t \in \mathbb{R}^{64}$

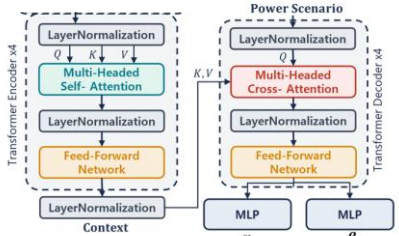
(stack for previous 24 hours)

$\Rightarrow \text{LSTM}(s_t; h_t, c_t) = h_t, c_t \in \mathbb{R}^{64}$

$\Rightarrow \text{Encoder}(s_t \oplus h_t) = \text{Context} \in \mathbb{R}^{2\times64}$

- **Model Input Variables – Decoder (Query):**

Variables	Description	Normalization
Start/End Power	Power at initial/final point	Power / 100
Delta Power	Initial to final power difference	Power / 100
Power Direction	Increase (+1), Decrease (-1), Consists (0)	Boolean
End Power Hold Time	Final power duration	
Start Burnup M/Z	Min-max/z-score normalized initial burnup	Min-max / Z-score



$Q \in \mathbb{R}^{1\times7}$

$\Rightarrow \text{Decoder}(\text{context} \oplus Q) = o \in \mathbb{R}^{1\times64}$

$\Rightarrow \text{Linear}_{\alpha,\beta}(o) = \alpha, \beta \in \mathbb{R}^2$

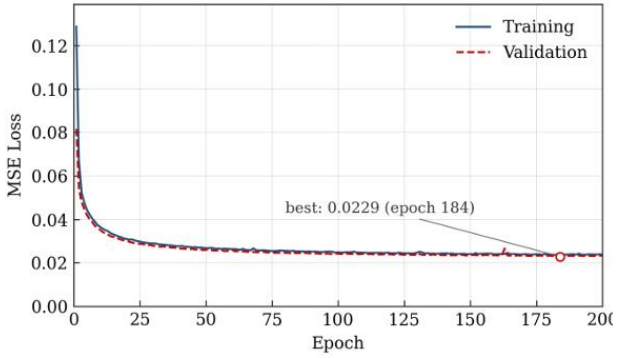
3.4. Model Training

- **Training Status:**
  - We reaches best **validation MSE at 0.02286** at epoch 184
  - With the model error, the validation |ASI| exceed 0.2 but under 0.3
    - In RL process, it keeps |ASI| under 0.3
  - CEA behaviour is not random shape, **become to feasible shape**

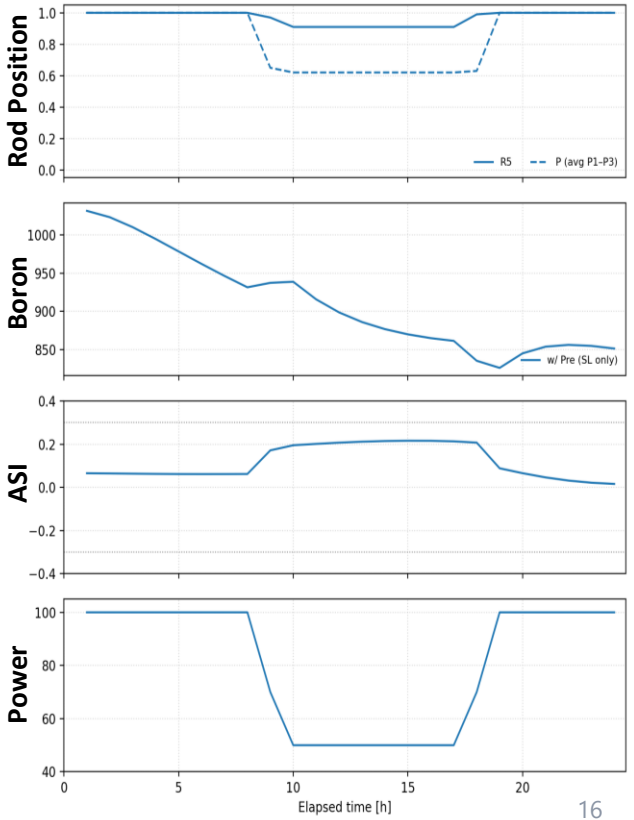
Training Configuration

Parameter	Description
Action Distribution	Beta function
Loss Function	MSE
Optimizer	AdamW, gradient clip = 1.0
Learning Rate	1e-4
Batch Size	32
Train/Validation/Test Split	0.7/0.2/0.1
Epochs / Best Epoch	200 / 184
Best Validation MSE	0.02286
Computing Platform	WSL2 (Ubuntu) in Window 11
CPU core/ RAM	24 / 128 GiB
GPU	NVIDIA L40S (48 GiB)

Learning Curve



Model Output Validation



04

Reinforcement Learning

For the feasibility of the control rod position

4. Reinforcement Learning

RL Objective: Development of a 100-50-100 CEA Operation Scenario Satisfying |ASI| < 0.3

RL Fine-tuning Workflow

4.1: Platform Development

- Coupling between core analysis code and Actor

4.2: RL Design

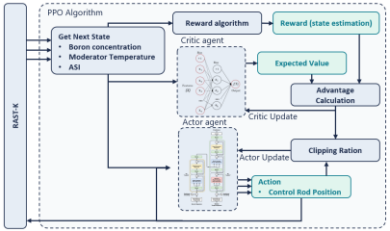
- Loss design; PPO algorithm
- 3-day scenario generation and validation via core analysis code

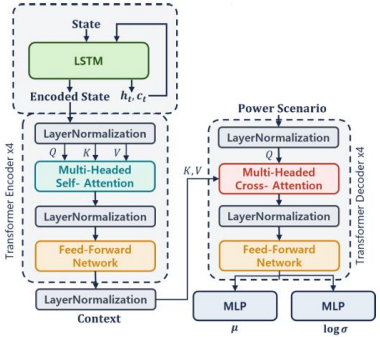
4.3: Reward Function Design

- |ASI|-centric reward function design

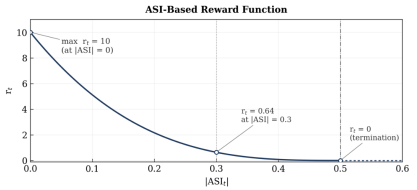
4.4: Training

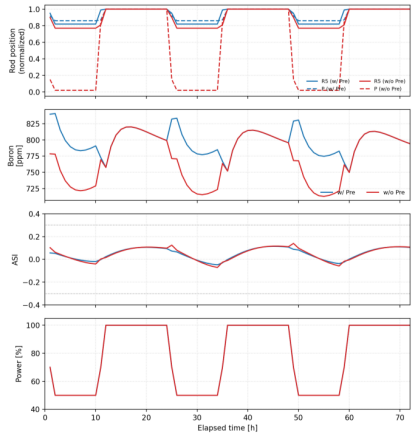
- Training reward tendency analysis
- Power profile analysis with the generated CEA position





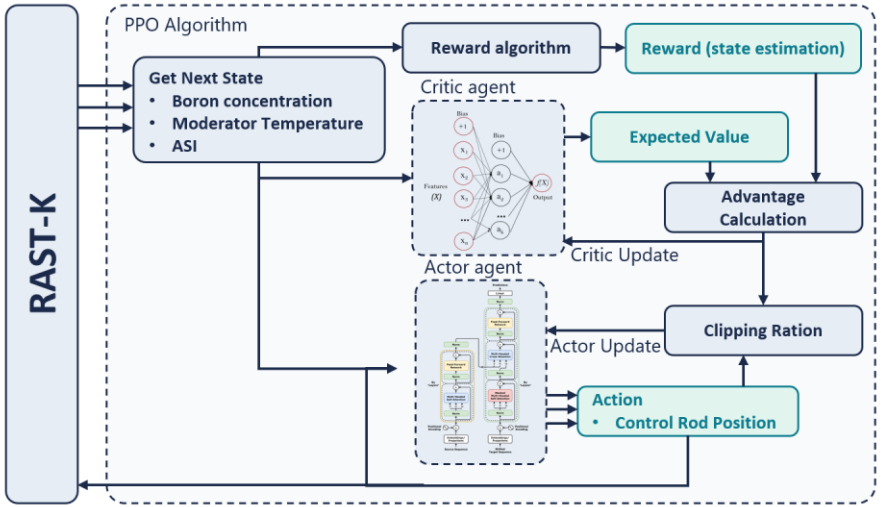
$$r_t = 10\left(\max\left(0, \frac{0.5 - |\mathrm{ASI}_t|}{0.5}\right)\right)^3$$





4.1. Platform Development

- **Platform Configuration:** Platform is based on **PPO algorithm** which is consist with **Actor-Critic agent**
  - **Actor Network**: produce action with state, **actions are the rod speed (R5, P)**
  - **Critic Network**: produce critic value which determine the **standard of reward value**
  - **Reward Function**: produce reward with state, current reward function only depends on ASI
  - **Environment**: we use **RAST-K** the 3D-modal core analysis code as an environment, it **receive action** from actor network, **produce next state**

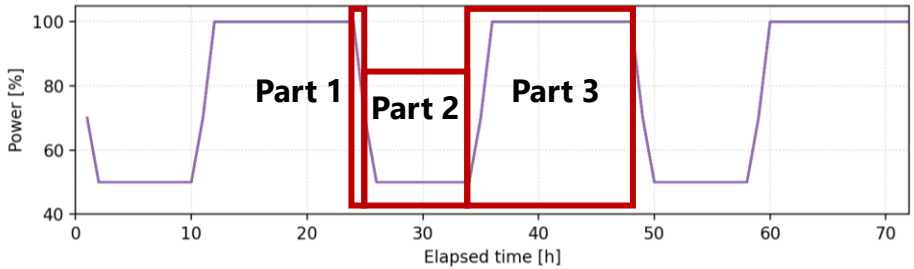


- **Action:** Rod speed (R5, P) for one step
- **State**: Diagnosis the core status from given action (rod speed)
  - Power, Burnup, ASI, Fuel/Moderator Temperature
- **Reward**: Depends on reward function of ASI

4.2. RL Design

- **RL Scenario:** Generate rod scenario of **APR1400 100-50-100 DLFO for 3 days** after 1 day in All Rod Out (ARO) state
  - **Part1**: 100% to 70% for 1 hour
  - **Part2**: 70% to 50% for 1 hour and consists 50% for 5 hours
  - **Part3**: 50% to 70% for 1hour, 70% to 100% for 1 hour and consists 100% for rest of 24 hours

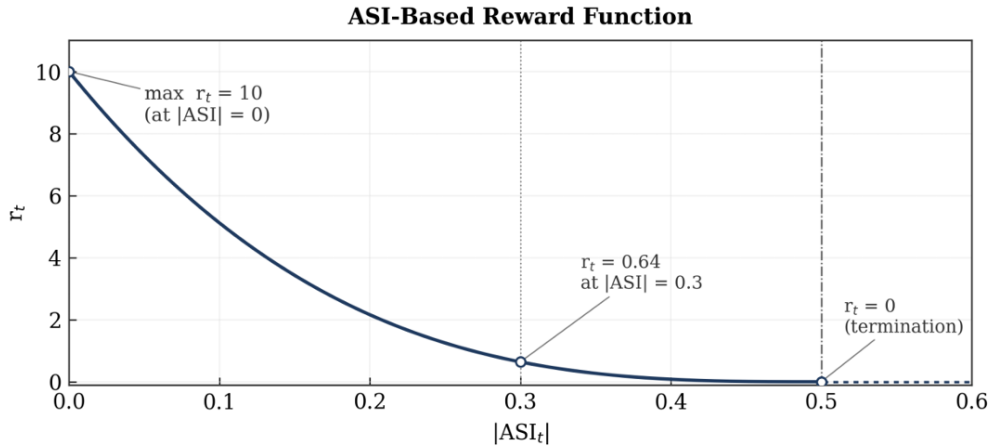
  } **One cycle = 1 day**



- **PPO Description:** PPO conduct **policy update clipping** to prevent excessive single-step deviation
  - **Loss**: $\mathcal{L}_{\text{actor}} = -\mathbb{E}\left[\min\left(\rho_t \hat{A}_t,\ \text{clip}(\rho_t, 1-\varepsilon, 1+\varepsilon)\hat{A}_t\right)\right], \quad \mathcal{L}_{\text{critic}} = \mathbb{E}[(V(s_t) - G_t)^2], \quad \mathcal{L} = \mathcal{L}_{\text{actor}} + 0.5 \cdot \mathcal{L}_{\text{critic}}$
    - A_t denotes **how much better an action is than the average**
    - PPO performance depends on the quality of A_t and stably **estimated via GAE**
  - **Generalized Advantage Estimation (GAE)**
    - $G_t = A_t + V(s_t), \quad A_t = \delta_t + \gamma\lambda A_{t+1}, \quad \delta_t = r_t + \gamma V(s_{t+1}) - V(s_t),$ γ: Discount Factor, V: Critic Value
    - Early in training, V is random and easily destabilizes learning and late in training, V becomes noisy and slows learning ⇒ Mitigated by interpolating between the two extremes

4.3. Reward Function Design

- **Reward Function Input:** Only |ASI|, all positive range
  - Since, a single reward function considers various control variables, **reward value vanishes each other** so that might reduce the learning efficiency and accuracy
  - If reward function get **negative range**, AI agent might be **chosen the termination** since avoiding the accumulation of negative reward is wors than termination (reward = 0)
  - Therefore, **negative reward cause the learning unstably**
- **Terminate condition**: we set the termination condition when |ASI| ≥ 0.5 which cause the simulator termination

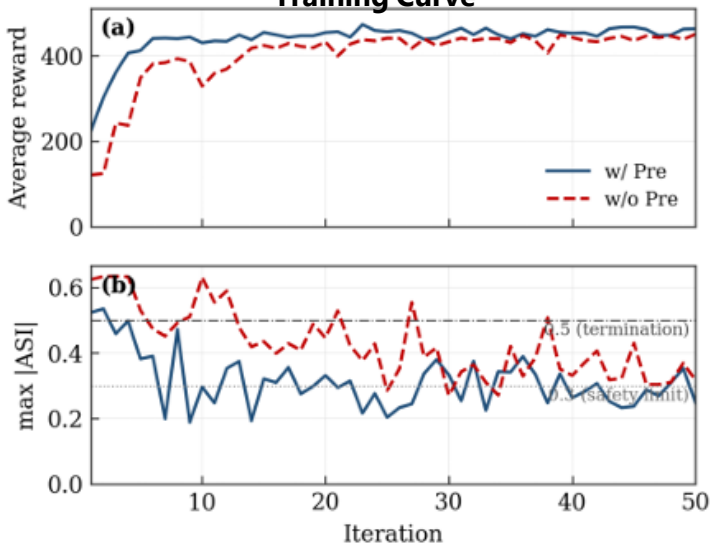


$$r_t = 10\left(\max\left(0, \frac{0.5 - |\mathrm{ASI}_t|}{0.5}\right)\right)^3$$

4.4. Training

- **Reward:**
  - Pre-trained model **[w]**: best reward is 463 at 24 iteration
  - Without pre-trained model **[w/o]**: best reward is 451 at 38 iteration
    - **[w]** get **best reward 3% higher** that **[w/o]** and reaches **37% faster**
- **Stability**:
  - Over the first five iterations, the termination rate average is 10% of **[w]** and 40% of **[w/o]**
  - At iteration 50, max |ASI| is 0.247 of **[w]** and 0.316 of **[w/o]** where the safety threshold is 0.3

Training Curve



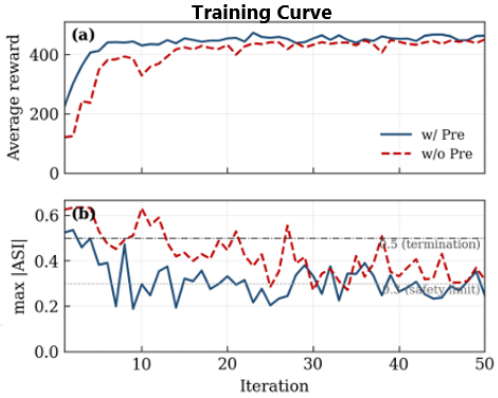
05

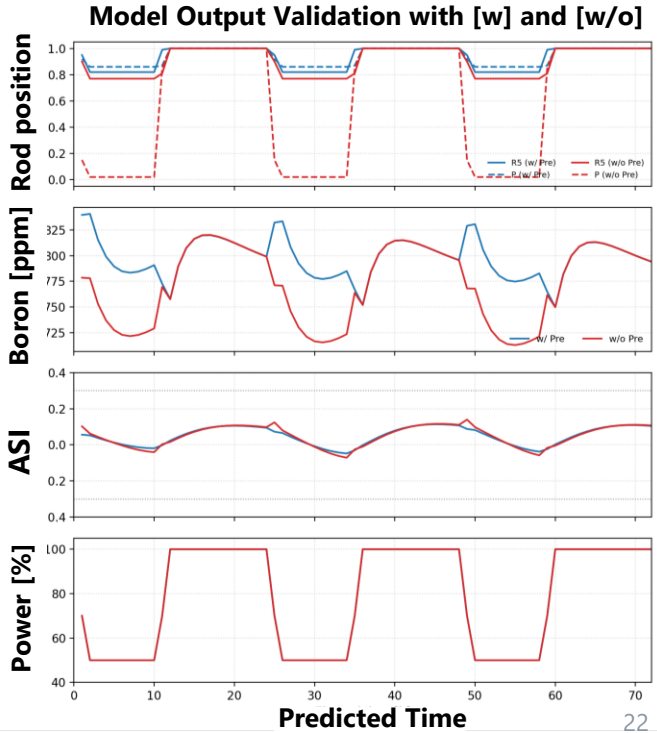
Results

For the feasibility of the control rod position

Model Output

- **At Training Process:**
  - As a results, we could check pre-trained model **[w]** has 37% fast learning speed than non-pre-trained model **[w/o]**
  - Furthermore, **[w]** has more stable in point of max|ASI|
- **At Output**:
  - **[w]** tend to use less CEA maneuver and more boron usage
  - **[w/o]** tend to use more CEA maneuver and less boron usage
  - **[w]** has smoother ASI trajectory compared to **[w/o]**
    - ASI has local peak in **[w/o]**
- **Advantages for Pre-trained model:**
  - There exists acceleration of learning speed
  - There exists stabler ASI generation





06

Conclusion & Future Works

For the feasibility of the control rod position

6. Conclusion & Future Work

Conclusion

- We shows the benefits for using pre-training model in reinforcement learning which is learning speed acceleration
- We shows pre-trained model generates more stable scenario such as less max |ASI| than non-pre-trained model
- We shows pre-trained model chose different strategy for perform DLFO compared with non-pre-trained model

Future Work

Scenario Expansion

We expand scenario length for comparing the maximum reward

Reward Function

Add more dependency such as boron concentration or CEA maneuver in reward function

Thank You

Questions & Discussion

Junhyeong Bang & Jonghyun Kim* | bjun201gosegulv@kaist.ac.kr

NICA Lab · Dept. of Nuclear and Quantum Engineering · KAIST

KAIST NICA

Nuclear I&C and Autonomous Operation Lab