Evolving reactor core simulation software through AI: Approaches and insights

Siarhei Dzianisau, Ph. D.

Postdoctoral Researcher at UNIST CORE Laboratory (Prof. Deokjung Lee)

Presented at Korean Nuclear Society Spring Meeting
AI-Accelerated Nuclear R&D: Latest Trends and Applications Workshop

CONTACT

Ulsan National Institute of Science and Technology
Address 50 UNIST-gil, Ulju-gun, Ulsan, 44919, Korea
Tel. +82 52 217 0114 **Web.** www.unist.ac.kr

Computational Reactor physics & Experiment lab
Tel. +82 52 217 2940 Web. reactorcore.unist.ac.kr

한국원자력학회
Korean Nuclear Society

Contents

1. Basics of reactor simulation software 3

2. How AI is used in reactor software 8

3. AI techniques developed at UNIST CORE 17
- Part 1 – Barcode CNN model 18
- Part 2 – RAST-AI, XSNET 27

4. Future of AI in reactor simulation 53

1. Basics of reactor simulation software

Explain about various kind of reactor software

- Simulating a reactor core is a complex process that should account for sophisticated movement of very high number of neutral particles – neutrons – that are produced during the fission reaction and drive the fission chain reaction further to keep the reactor critical.
  - The movements of neutrons have a stochastic (random) nature, especially once they slow down enough to become thermal — meaning they reach thermal equilibrium with the surrounding medium and their energy distribution resembles that of gas molecules at the same temperature.
- To approximate the effects of neutron movements, there are two main approximation principles employed in modern reactor simulation software:
  - Neutron transport solves the Boltzmann transport equation and explicitly tracks the angular distribution of neutrons — accounting for all travel directions in 3D space.
  - Neutron diffusion applies Fick's law as an approximation to the Boltzmann equation, collapsing the full angular description into a scalar neutron flux.
    - It assumes weakly anisotropic scattering, making it computationally much cheaper than transport but less accurate near boundaries and in strongly absorbing regions.

Explain about various kind of reactor software

- **Neutron transport codes have two main approaches:**
  - **Deterministic neutron transport codes that use deterministic methods to approximate the neutron transport equation, such as the Method of Characteristics (MOC) and Discrete Ordinates (S_N).**
  - **Probabilistic (Monte Carlo) methods simulate the full stochastic lifecycle of individual neutrons — from birth to absorption or leakage — across a large population, converging to a statistical solution.**
  - **Deterministic methods are faster but rely on geometry discretization and multi-group energy approximations; Monte Carlo is more accurate in complex geometries but computationally far more expensive.**
  - **Deterministic methods rely on fine spatial, energy, and angular discretization, while Monte Carlo methods track particles through continuous geometry.**
    - **Both result in high computational demand — deterministic from discretization resolution, Monte Carlo from the large number of particle histories required for statistical convergence.**

Explain about various kind of reactor software

- **Neutron diffusion codes work with a homogenized, coarse-mesh geometry — each fuel assembly is represented as a single node with spatially averaged (homogenized) material properties — allowing efficient nodal methods to solve the diffusion equation over the full reactor core.**
  - **The result is a reasonably accurate solution, especially in the interior of the core (far from boundaries and material interfaces), where the diffusion approximation holds well.**
    - **Accuracy degrades near boundaries, control rod tips, and strongly absorbing regions.**
  - **Much lower computational burden compared to neutron transport approach – a single CPU core produces results in seconds or minutes, depending on the task and settings.**

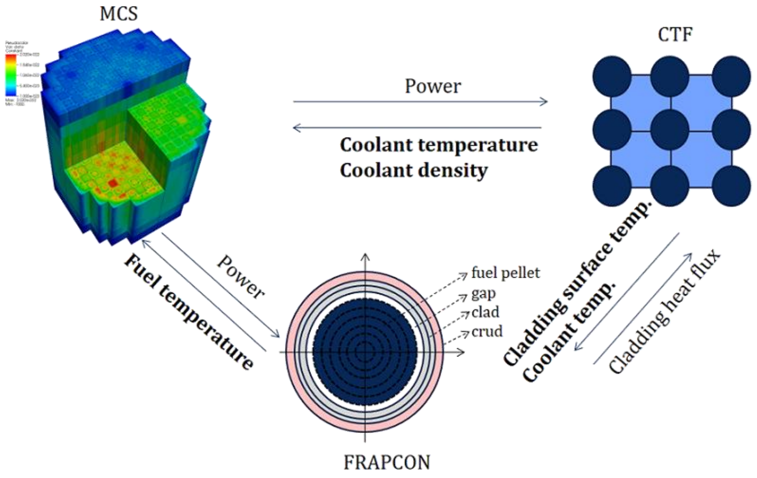
Reactor simulation software at UNIST CORE

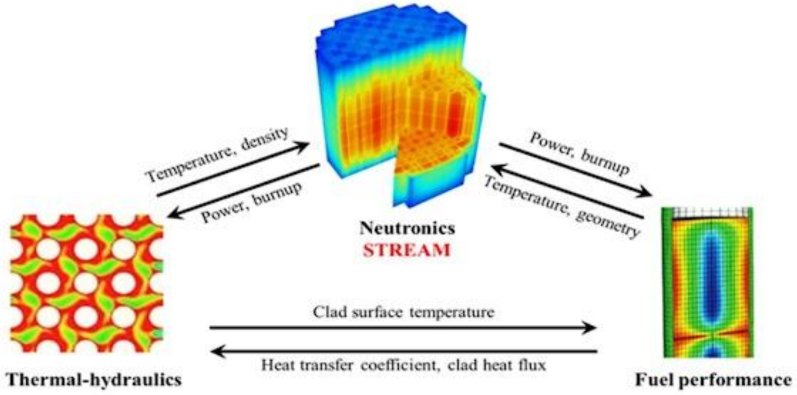
- **UNIST CORE Lab** is specialized in **developing reactor simulation software** for high-fidelity steady-state and transient analysis.

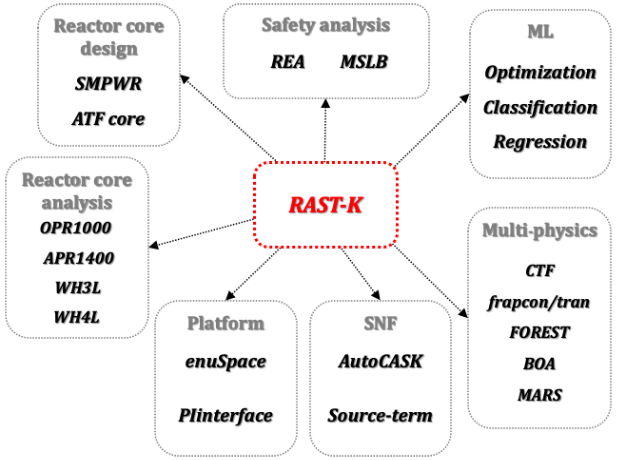
MCS is a general-purpose Monte-Carlo code with TH feedback and capability to run steady state-depletion and time-dependent problems.

STREAM is a deterministic neutron transport code for high-fidelity steady-state depletion and transient reactor analysis based on MOC solver.

RAST-K is a nodal diffusion code with fast solving schemes for thermal and fast reactors.







2. How AI is used in reactor software

General overview of different techniques for reactor simulation

- **There are a few worthy publications in the reactor simulation software industry.**
- **One way AI could speed up our (researchers) workflow is to help us find relevant literature for our studies.**
- **To find relevant works, I used Perplexity app with Claude Sonnet 4.6 model.**
  - **ChatGPT and Gemini were also tested but performed much worse compared to Claude.**
- **In the following slides, a brief overview of notable studies is to be presented.**

Radaideh et al. (2021-2024)

- NeoRL algorithm library and its coupling with OpenMC Monte Carlo transport code, enabling Reinforcement Learning (RL) and evolutionary optimization of reactor configurations with live full-physics transport evaluations at each step.
  - Optimization loop (fuel loading, detector placement, burnup targets) calls real OpenMC transport — not a surrogate — at each step.
  - Supports RL, genetic algorithms, and neuroevolutionary methods through a unified API.
  - Open-source availability makes it a reproducible community reference architecture for AI-integrated transport-code optimization.

X. Gu, M. Radaideh, J. Liang, OpenNeoMC: A framework for design optimization in particle transport simulations based on OpenMC and NEORL, **Ann. Nucl. Eng.**, Volume 180, 109450, 2023.

Shriver & Watson et al. (2021–2024)

- **LatticeNet is a Convolutional Neural Network (CNN) framework that predicts pin-level neutron flux distributions, kinetic parameters (Doppler/moderator temperature coefficients), and differential boron worth directly from assembly inputs — replacing CASMO-4E branch calculations with sub-millisecond inference.**
  - **All-parameter network predicts 1,100 branch points in 236 ms vs. 4.95 min for CASMO-4E — a ~1,260× speedup — with differential boron worth mean error of only 0.30%.**
  - **Developed in collaboration with Framatome/ANP on the CASMO/SIMULATE ecosystem, with errors within acceptance criteria for industrial licensing workflows.**
  - **Extended to Monte Carlo-trained variants (OpenMC) and atypical assembly configurations, demonstrating robustness beyond standard training conditions.**

A. Furlong, J., F. Shriver, Investigation of Monte Carlo trained CNNs for neutronics predictions of typical and atypical PWR assemblies, **Prog. Nucl. En.**, Volume 166, 104961, 2023.

Calvin et al. (2025)

- **A neural network (NN) is deployed inside a production reactor physics code (INL's Griffin) through a compiled C++ backend (LibTorch), running natively within the same binary as the physics solver.**
  - **NN replaces the resonance self-shielded microscopic cross-section lookup from ACE libraries at runtime during multiphysics depletion — not as a pre-processing wrapper.**
  - **Validated against full Serpent2/MCNP reference calculations for a pebble bed reactor (PBR) configuration.**
  - **LibTorch integration means NN inference runs inside the same compiled simulation binary — architecturally distinct from all surrogate approaches.**

O. W. Calvin, Y. Che et al., Deployment of neural-network-based neutron microscopic cross sections in the Griffin reactor physics application, **Ann. Nucl. Eng.**, Volume 220, 111509, 2025.

Schiassi et al. (2022)

- **Replaces the ODE-based point kinetics equation (PKE) solver with a Physics-Informed Neural Network (PINN) X-TFC, embedding the governing equations with delayed neutrons directly in the loss function, achieving analytical-quality solutions without time-stepping.**
  - **Solves stiff PKE systems with reactivity-driven transients — a notoriously difficult ODE class — validated against benchmark datasets.**
  - **X-TFC converges significantly faster than standard collocation PINNs; mesh-free formulation eliminates time-stepping discretization entirely.**
  - **Targets the kinetics module of reactor simulation codes directly, not a post-hoc surrogate of an existing solver.**

E. Schiassi, M. De Florio et al., Physics-informed neural networks for the point kinetics equations for nuclear reactor dynamics, **Ann. Nucl. Eng.**, Volume 167, 108833, 2022.

Leniau et al. (2015)

- **The earliest confirmed case of a NN becoming a permanent shipped component of a production fuel cycle code CLASS, replacing the Bateman equation depletion solver for UOX and MOX fuel cycles.**
  - **Two multi-layer perceptron (MLP) embedded in CLASS: one predicting fresh fuel loading composition, the other predicting mean cross-sections as a function of burnup and fuel state.**
  - **Maximum deviation of 3% on major nuclide inventories vs. reference MURE coupled transport/depletion calculations.**
  - **Enables full fuel cycle scenario simulation in under one minute, opening scenario classes impractical with coupled Monte Carlo.**

B. Leniau, B. Mouginot et al., A neural network approach for burn-up calculation and its application to the dynamic fuel cycle code CLASS, **Ann. Nucl. Eng.**, Volume 81, Pages 125-133, 2015.

Dzianisau et al. (2022/2023)

- **RAST-AI is the first standalone reactor code that runs the full steady-state and transient simulation pipeline using a deep neural network (DNN) in place of all lattice physics calls — no external STREAM execution required.**
  - **The DNN generates macroscopic cross-sections internally, eliminating the lattice code from the two-step pipeline entirely rather than just wrapping it.**
  - **Transient scenarios (rod ejection) are validated against the full STREAM/RAST-K reference system within 0.5% mean relative error on power distribution.**
  - **Explicitly designed for loading pattern optimization workflows where the 20–30 min/case STREAM cost was the bottleneck.**

S. Dzianisau, K. Saeju, H. C. Lee, D. Lee, "Development of an Artificial Neural Network Model for Generating Macroscopic Cross-Sections for RAST-AI," **Ann. Nucl. Energy**, Volume 186, 109777, 2023.

Dzianisau et al. (2024)

- **XSNET is a drop-in replacement for STREAM that generates the complete .STN parameter file consumed by RAST-K, with zero changes required to the downstream nodal solver workflow.**
  - **Produces the full .STN file (300+ parameter conditions per assembly state) in seconds vs. a multi-minute STREAM execution.**
  - **Supports 16×16 and 17×17 PWR assembly types with depletion and transient branch coverage.**

S. Dzianisau, D. Lee, "Application of Artificial Neural Network for Assembly Homogenized Equivalence Parameter Generation," **Prog. Nucl. Energy**, 2024.

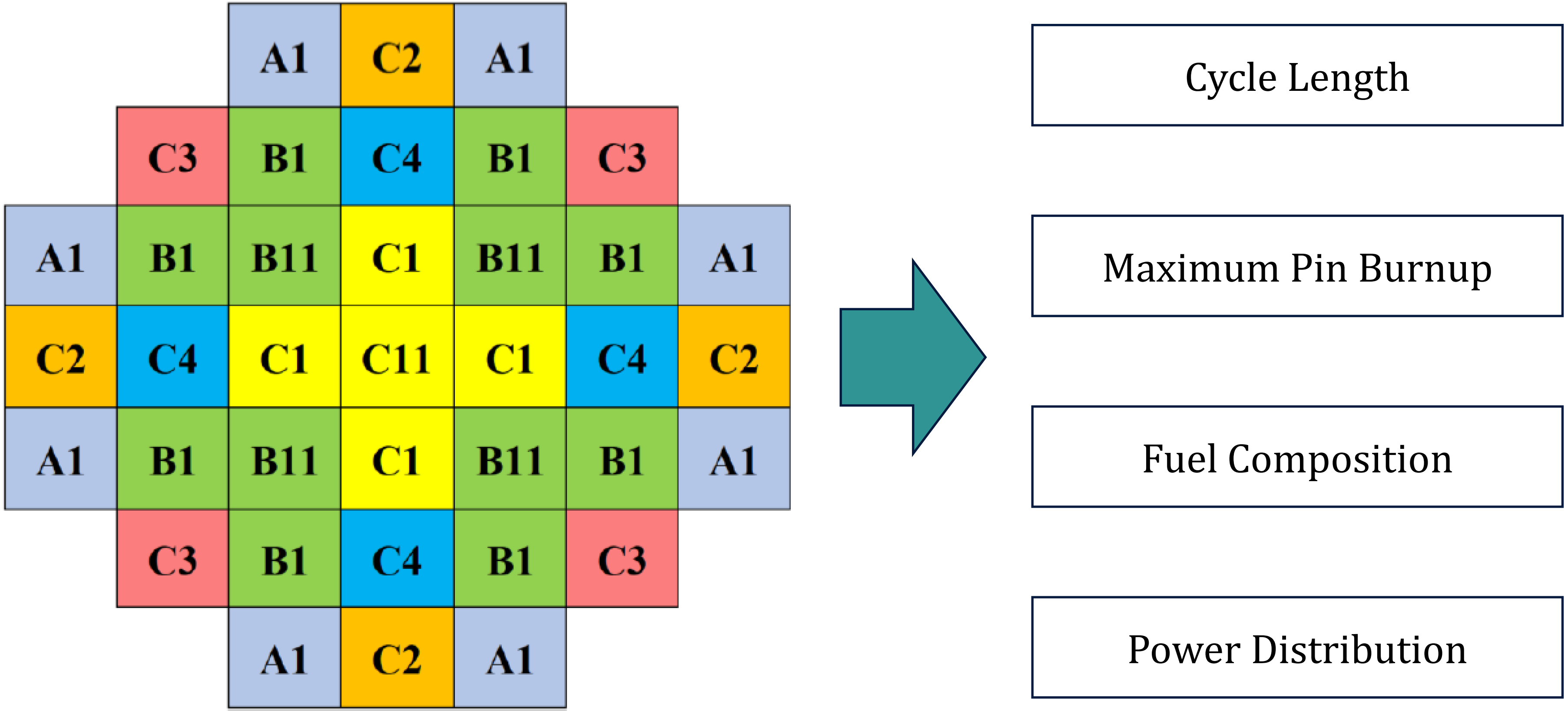
3. AI techniques developed at UNIST CORE

Part 1 – Accelerating a routine nuclear simulation task via AI

- **Problem**: to find an optimal or close to optimal loading pattern configuration
- **Reason**: with the same fuel types, it is possible to get longer cycle time, which results in bigger profit.
- **Obstacle**: complexity of the task requires testing many fuel loading pattern configurations.
  - AI can help filter out some of the configurations before evaluation, which will save a lot of time.
- For the optimization, Simulation Annealing algorithm was applied.
  - **Downside**: slow convergence, which means many code runs are required.
- **With AI model**: much fewer actual code runs as most LP designs are rejected by the AI model.

Applicability of CNN for Reactor Analysis

- **Features of CNN could be used in Reactor Core Design and Analysis.**
  - **A regular Core Loading Pattern (LP) looks like an image.**



		A1	C2	A1		
	C3	B1	C4	B1	C3	
A1	B1	B11	C1	B11	B1	A1
C2	C4	C1	C11	C1	C4	C2
A1	B1	B11	C1	B11	B1	A1
	C3	B1	C4	B1	C3	
		A1	C2	A1		

Cycle Length

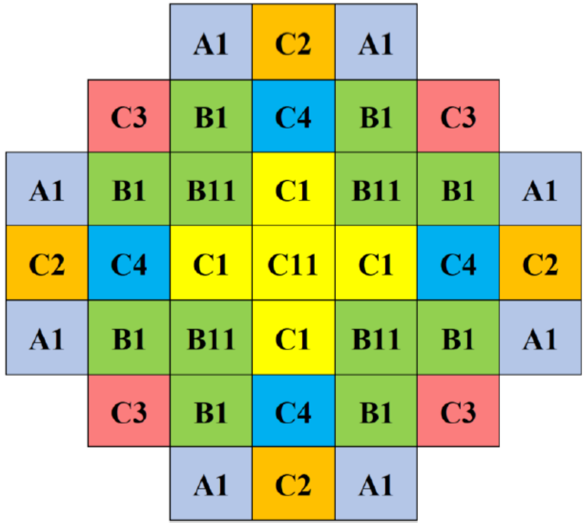
Maximum Pin Burnup

Fuel Composition

Power Distribution

Comparing a LP layout to a digital image

- **A LP layout offers some similarity to a digital image.**



		A1	C2	A1		
	C3	B1	C4	B1	C3	
A1	B1	B11	C1	B11	B1	A1
C2	C4	C1	C11	C1	C4	C2
A1	B1	B11	C1	B11	B1	A1
	C3	B1	C4	B1	C3	
		A1	C2	A1		

Take one pixel of an “image”

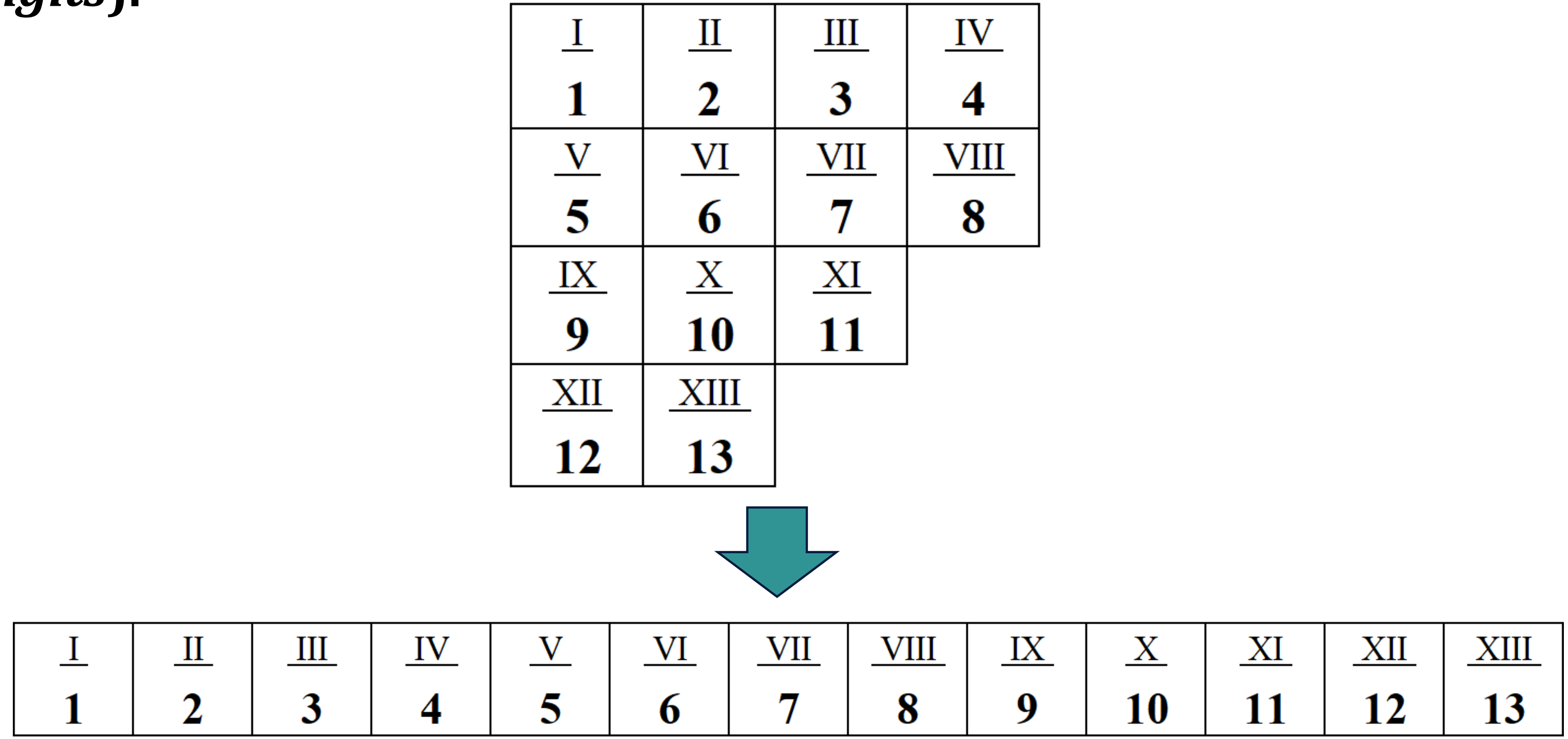
B1

Σ_{a1}	Σ_{a2}	Σ_{f1}	Σ_{f2}	. . .	etc.

R	G	B

Introduction of the Barcode Model approach

- **Enumerate all Fuel Assembly (FA) positions in certain strict sequence (*Roman digits*).**



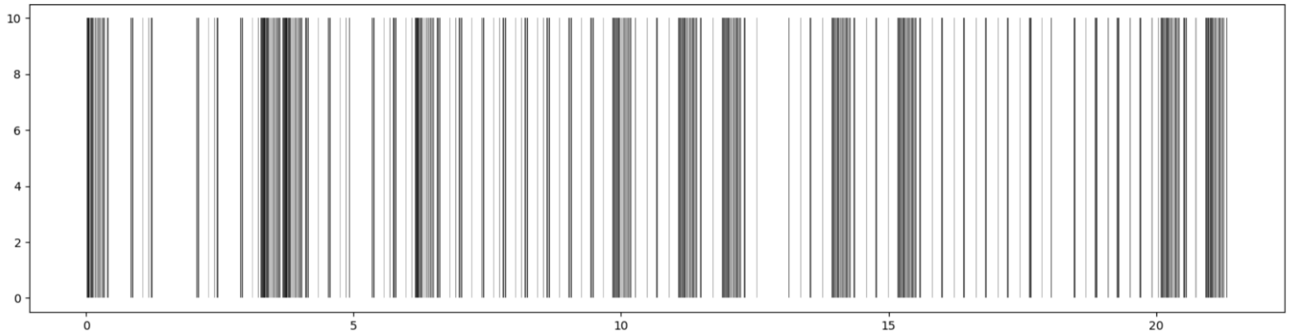
I 1	II 2	III 3	IV 4
V 5	VI 6	VII 7	VIII 8
IX 9	X 10	XI 11	
XII 12	XIII 13		

I 1	II 2	III 3	IV 4	V 5	VI 6	VII 7	VIII 8	IX 9	X 10	XI 11	XII 12	XIII 13

  - Restate all FAs in 1D format.

Origin of the name

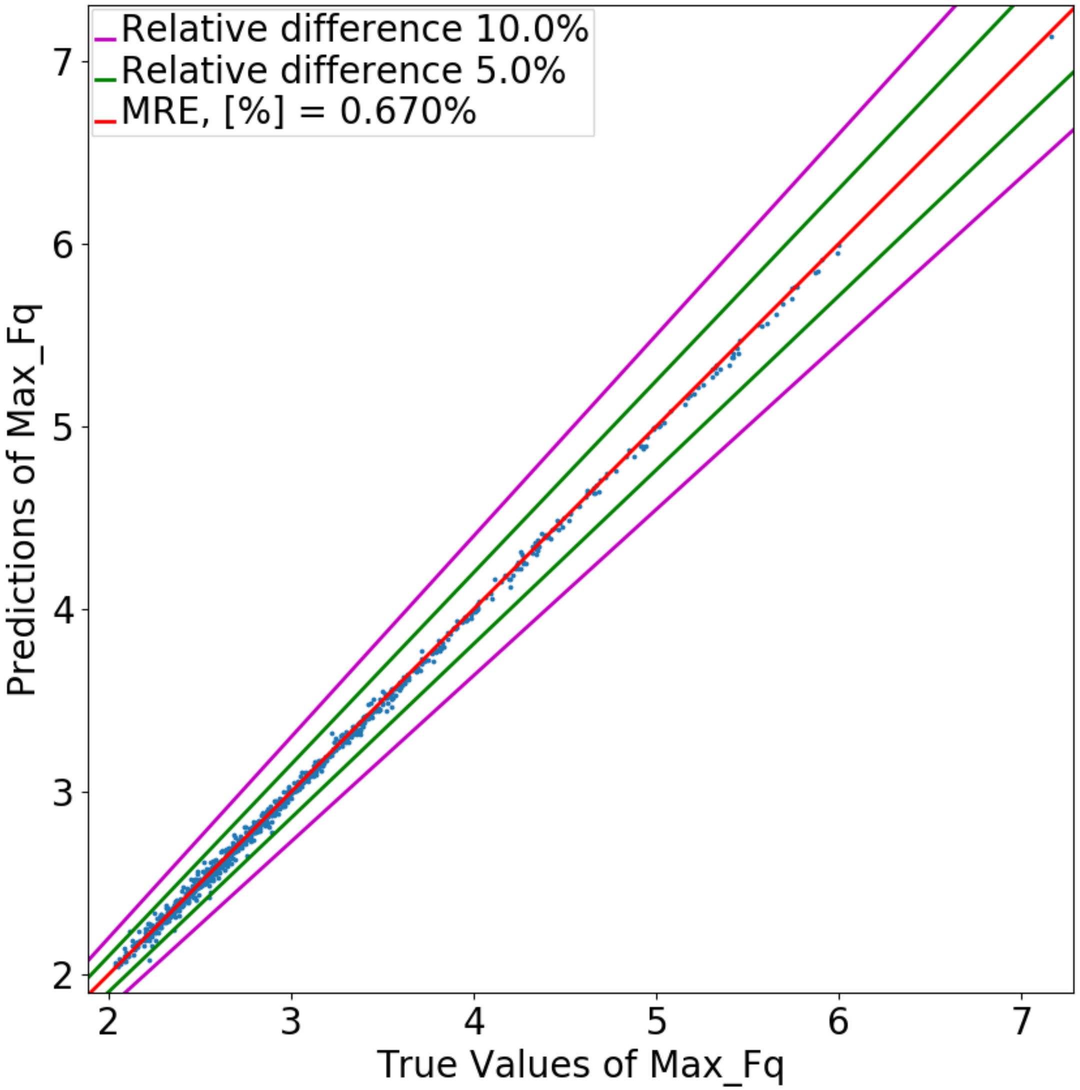
- **The model was called “Barcode” because the printed data of this model looks like a typical barcode.**



3 4 4 B A 8 7 6 Y 8 9

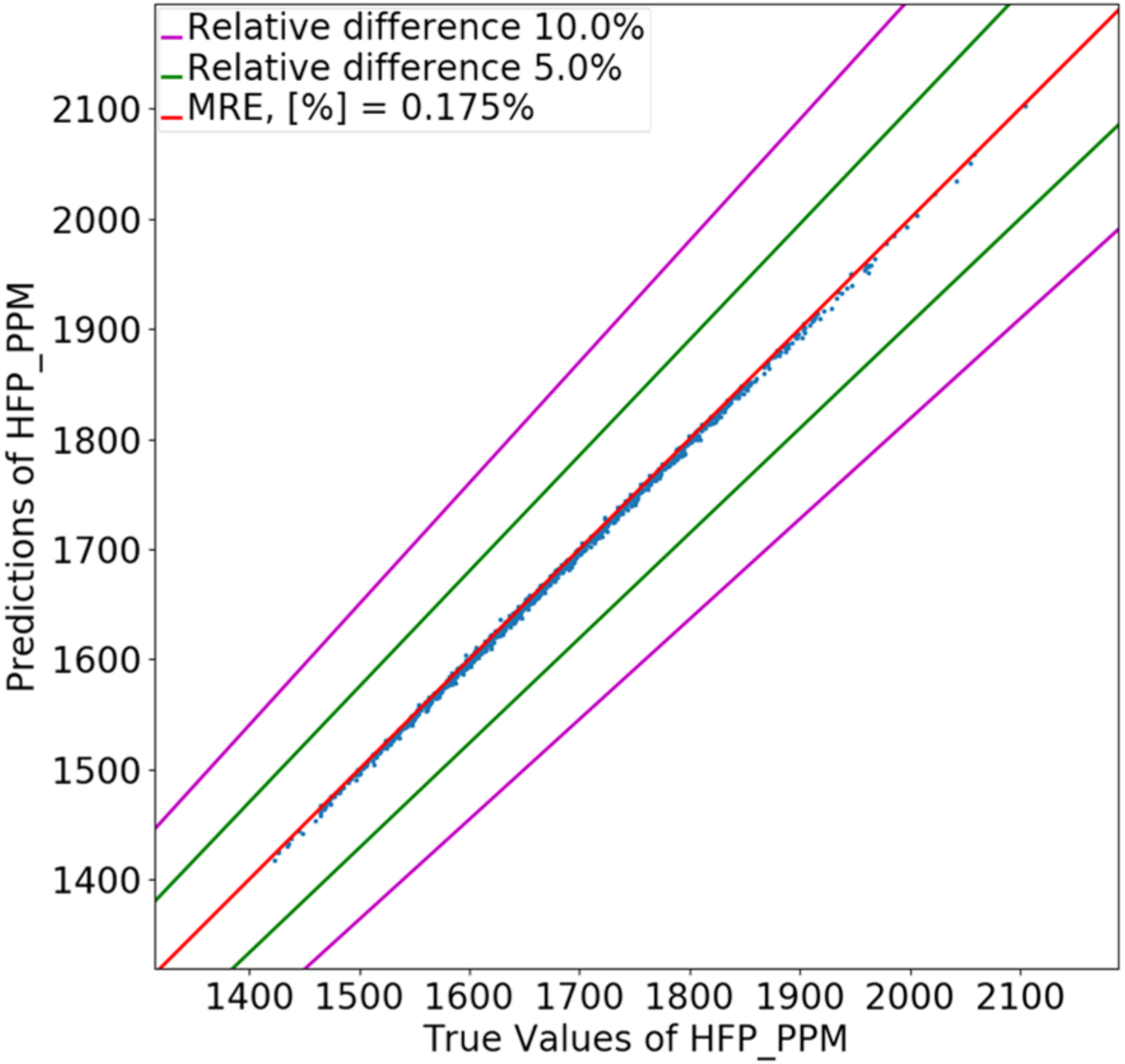
Barcode model – model outputs

- **Optimized model**
  - **Scatter plot (Target vs. Prediction) for Fq**



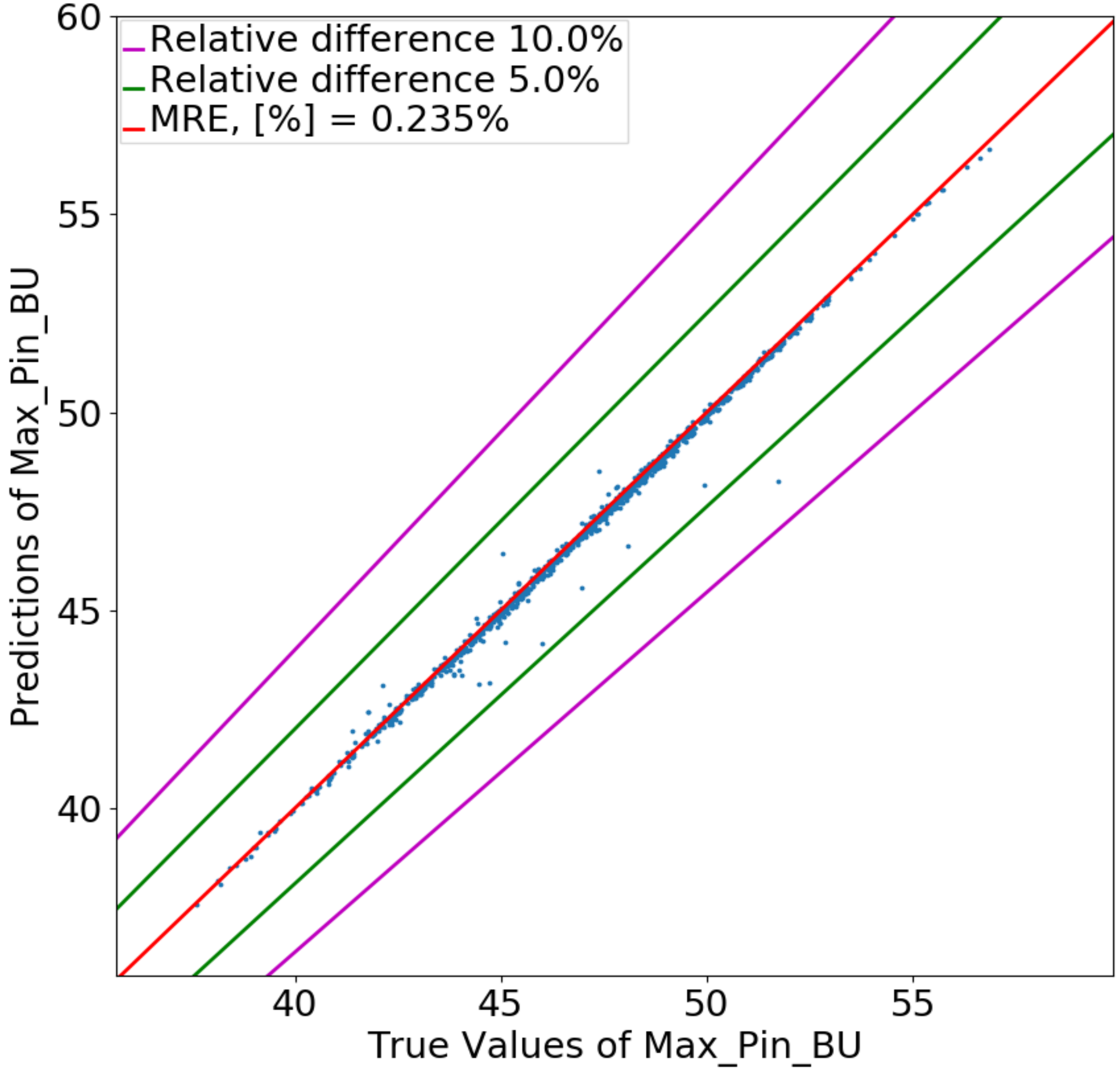
Barcode model – model outputs

- **Optimized model**
  - **Scatter plot (Target vs. Prediction) for HFP CBC**



Barcode model – model outputs

- **Optimized model**
  - **Scatter plot (Target vs. Prediction) for Max. Pin BU**



What is the advantage

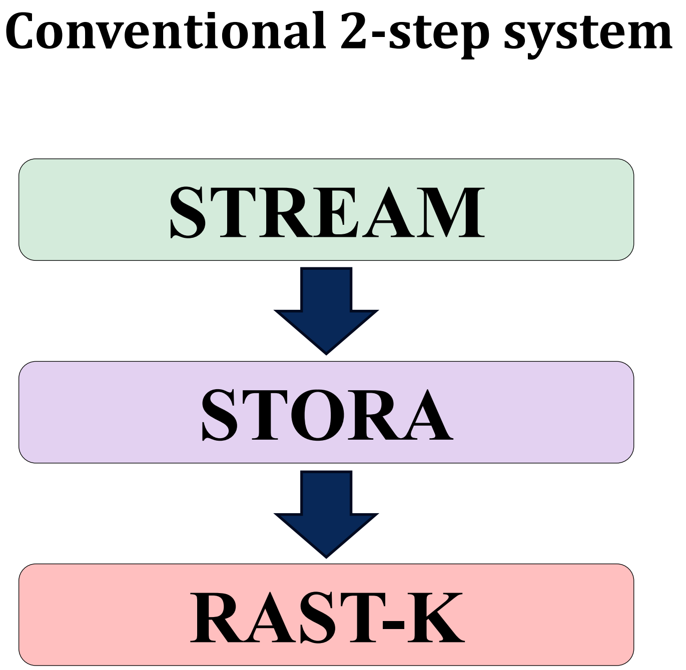
- Traditional manual search of the optimal LP could take up to 1 month and involve a highly qualified human researcher.
- Outsourcing this process to a combination of an optimization algorithm (such as Simulated Annealing – SA) and a reactor simulation code could take much longer via a single CPU core, but reasonably shorter when using multiple CPU cores.
  - This is because SA generates hundreds of thousands candidate LPs, whereas human operator would require hundreds to thousands of LPs, with the analysis of each new LP in between.
- Filtering out obviously bad LP candidates before simulating them with a reactor simulation code saves up to hundreds of thousands runs, can be fully automated, and eventually makes the LP search much faster compared to a manual search.
  - It reduces the required time from weeks to hours, compared to manual search.
- For the final optimized LP, conventional reactor simulator code is still executed to ensure accuracy and compliance to the regulator standards.

Part 2 – Coupling an AI model with a reactor simulation code

- **Problem**: to run a conventional 2-step calculation to assess the properties of a reactor core.
- **Reason**: this is a routine procedure for finding an optimal LP or designing a new reactor core.
- **Obstacle**: lattice physics code calculation – XS homogenization – is slow if required to be done many times (when designing new core or optimizing existing fuel configuration).
- An AI model can substitute the lattice physics code to speed up the process.
- **With AI model**: near-instant homogenization, which allows faster inference between the nodal code and the XS generation code.
  - Some accuracy is reasonably sacrificed.

What is RAST-AI

- **RAST-AI is a hybrid between our in-house nodal diffusion code RAST-K and a trained ANN.**
  - **Unlike in conventional 2-step systems, there is no need to run additional codes for generating macroscopic XS data as this data is generated by the ANN model.**
  - **The idea behind RAST-AI is not to compete with other codes in terms of accuracy but to be easy to learn and to use, especially for beginners, as well as provide reasonable quality of results at low computation costs.**

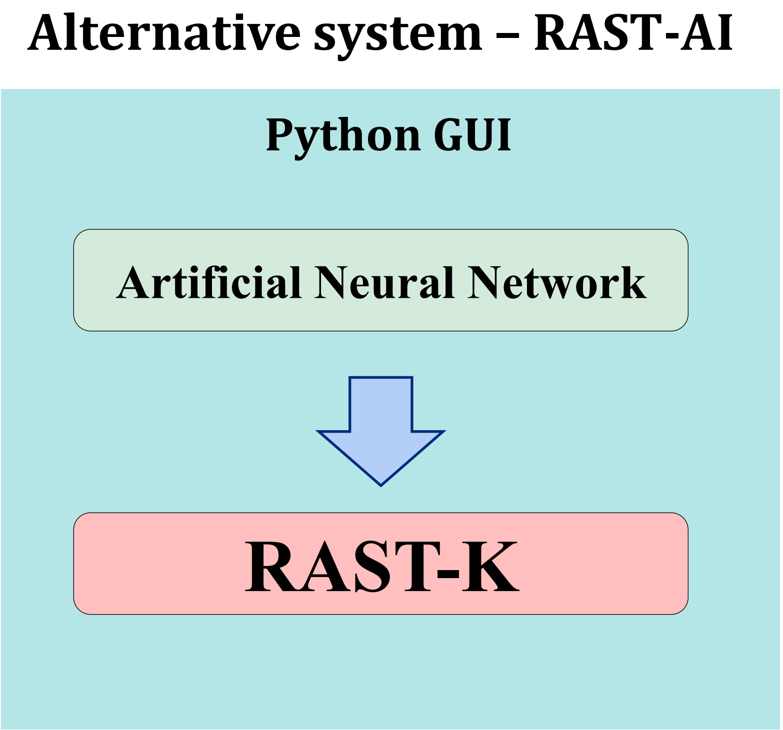


Conventional 2-step system

STREAM

STORA

RAST-K



Alternative system – RAST-AI

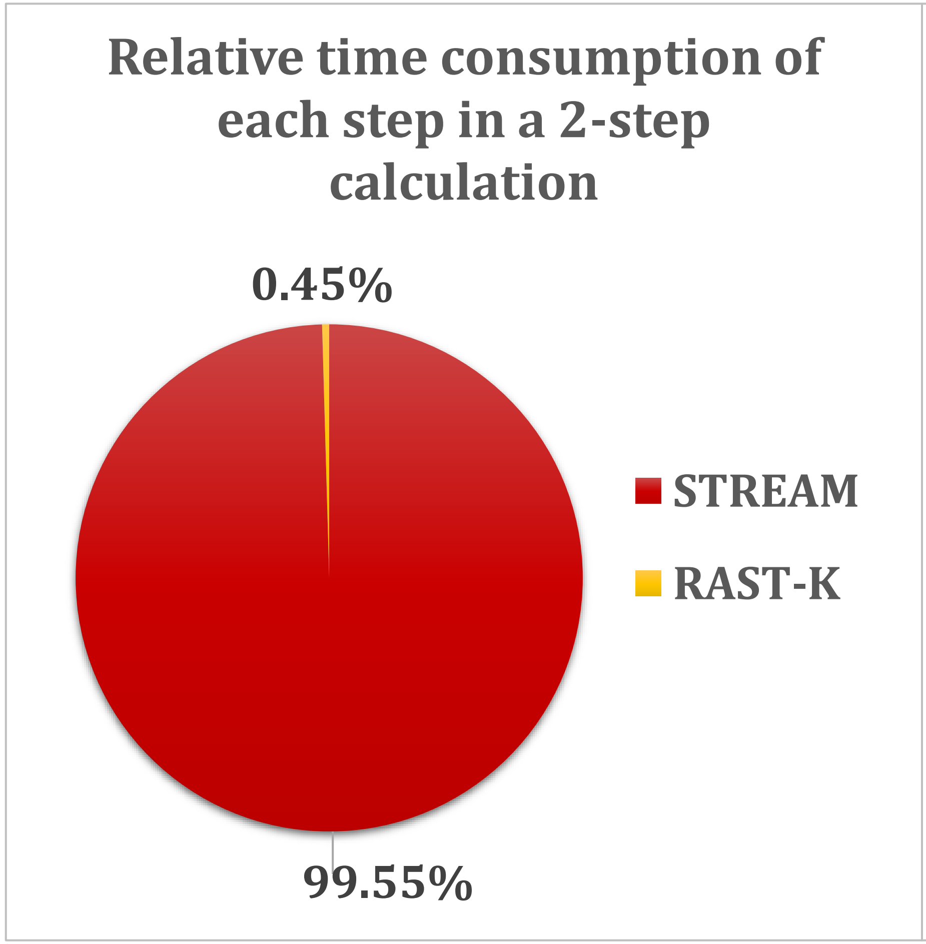
Python GUI

Artificial Neural Network

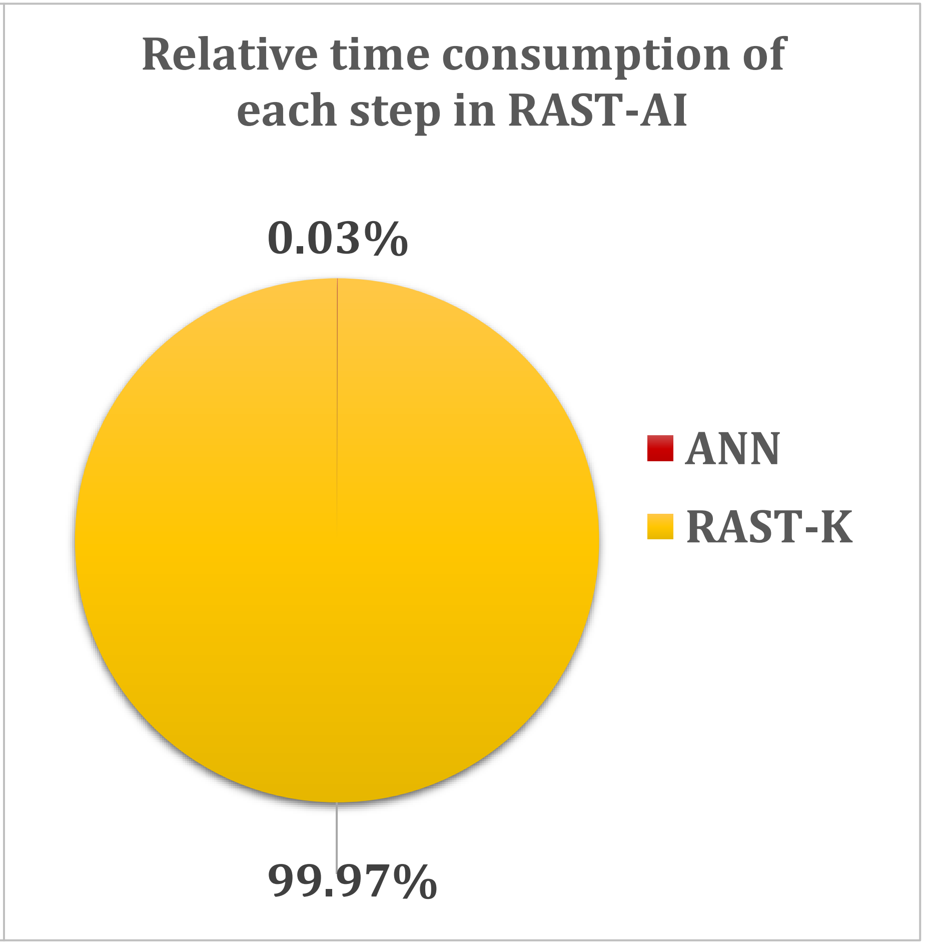
RAST-K

Time Consumption of a 2-Step Method

- **XS generation takes a significant amount of time compared to nodal calculation.**
  - **The Figure below shows the result of 20-step Depletion calculation in RAST-K with 10 FA types generated using STREAM.**



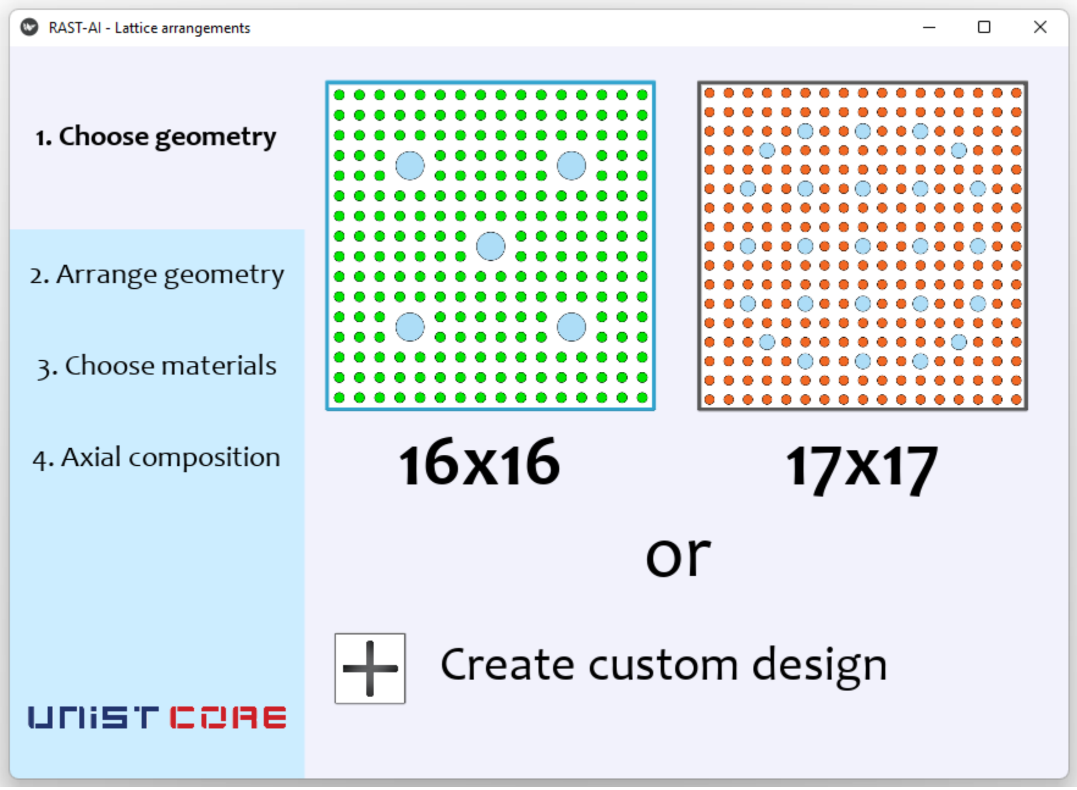
Relative time consumption of each step in a 2-step calculation

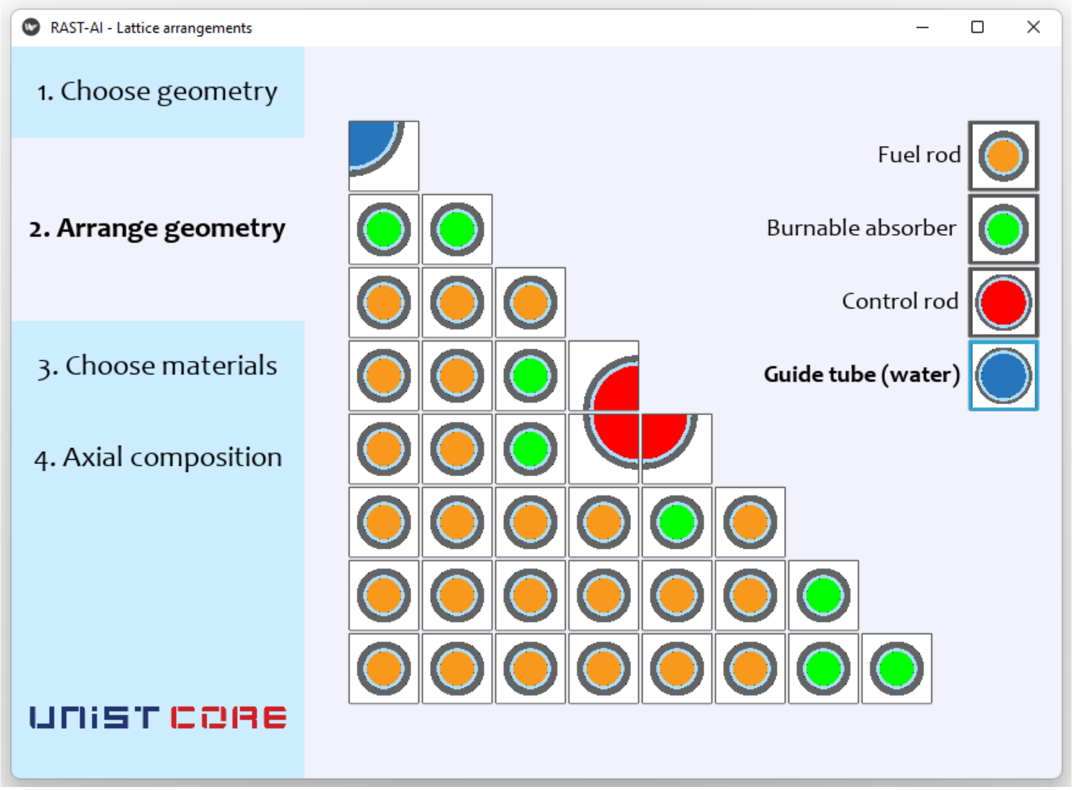


Relative time consumption of each step in RAST-AI

Graphical User Interface

- Python-based GUI, coupling through RAST-K input/executable, supported by both Windows and Linux (cross-platform).





S. Dzianisau, K.Saeju, H. C. Lee, D. Lee, "Development of an Artificial Neural Network Model for Generating Macroscopic Cross-Sections for RAST-AI," **Ann. Nucl. Energy**, 186:109777, 2023.

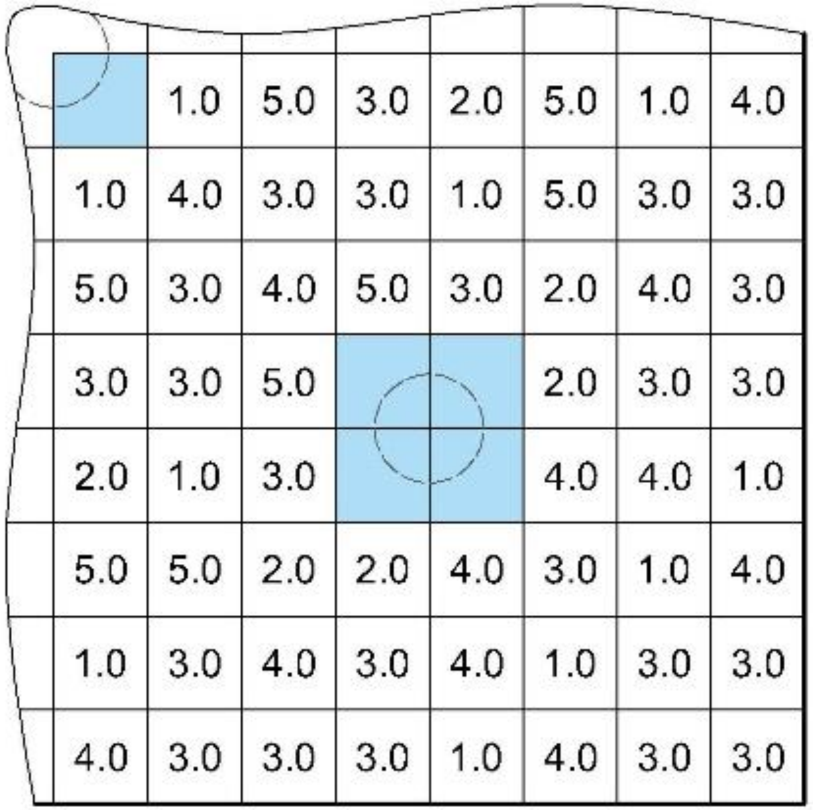
Branching to XSNET

- RAST-AI in the implemented form is a useful tool for evaluation fresh fuel LP, but due to the coupling strategy and limitations of RAST-K, it could not perform more advanced work, such as depletion calculation.
    - This was in part due to limited output of the ANN model used in RAST-AI.
- Therefore, the decision was made to first improve the ANN model and branch it out into a separate standalone code – XSNET.
    - XSNET acted as a 100% replacement of the lattice physics code STREAM.
    - It generates the same .STN file as STREAM, which is accepted by STORA coupling code.
    - The .STN file unlocks full functionality of RAST-K and makes it able to run depletion calculation and time-dependent transient calculation.
- In further slides, recent advances of XSNET and XSNET/RAST-K are presented.

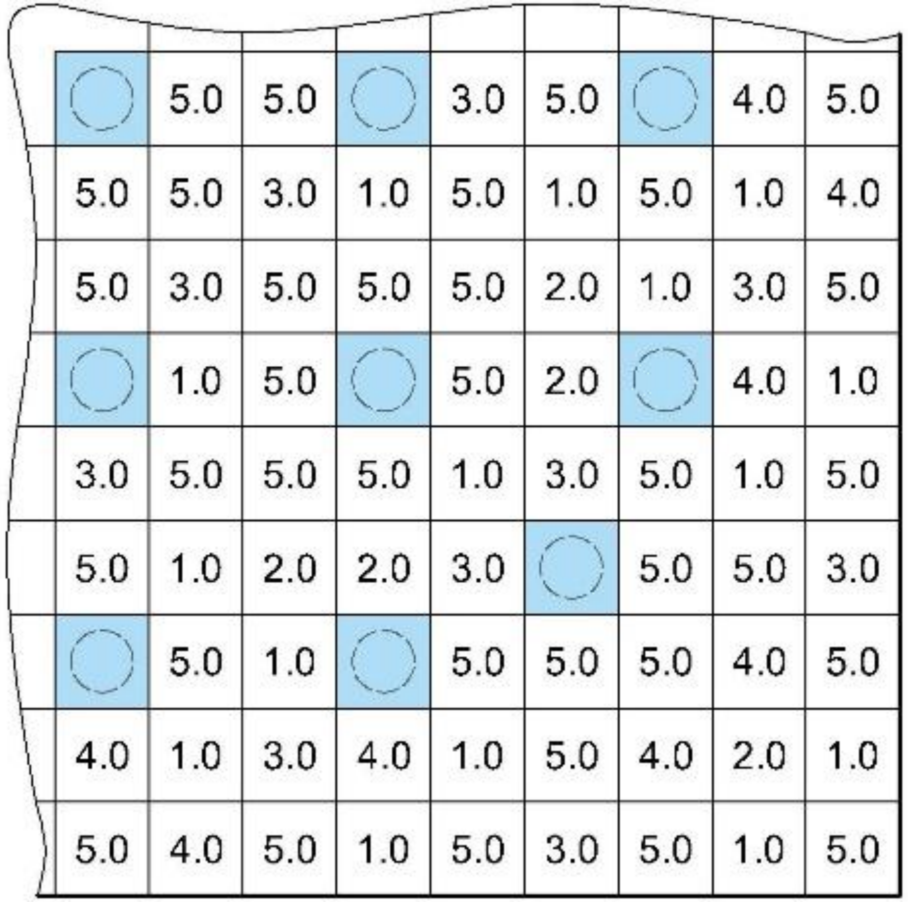
S. Dzianisau, D. Lee *, "Application of Artificial Neural Network for Assembly Homogenized Equivalence Parameter Generation," **Prog. Nucl. Energy**, 2024.

Input structure of the ANN model

- **Fuel enrichments – random order in a range 1-5 wt% U-235.**



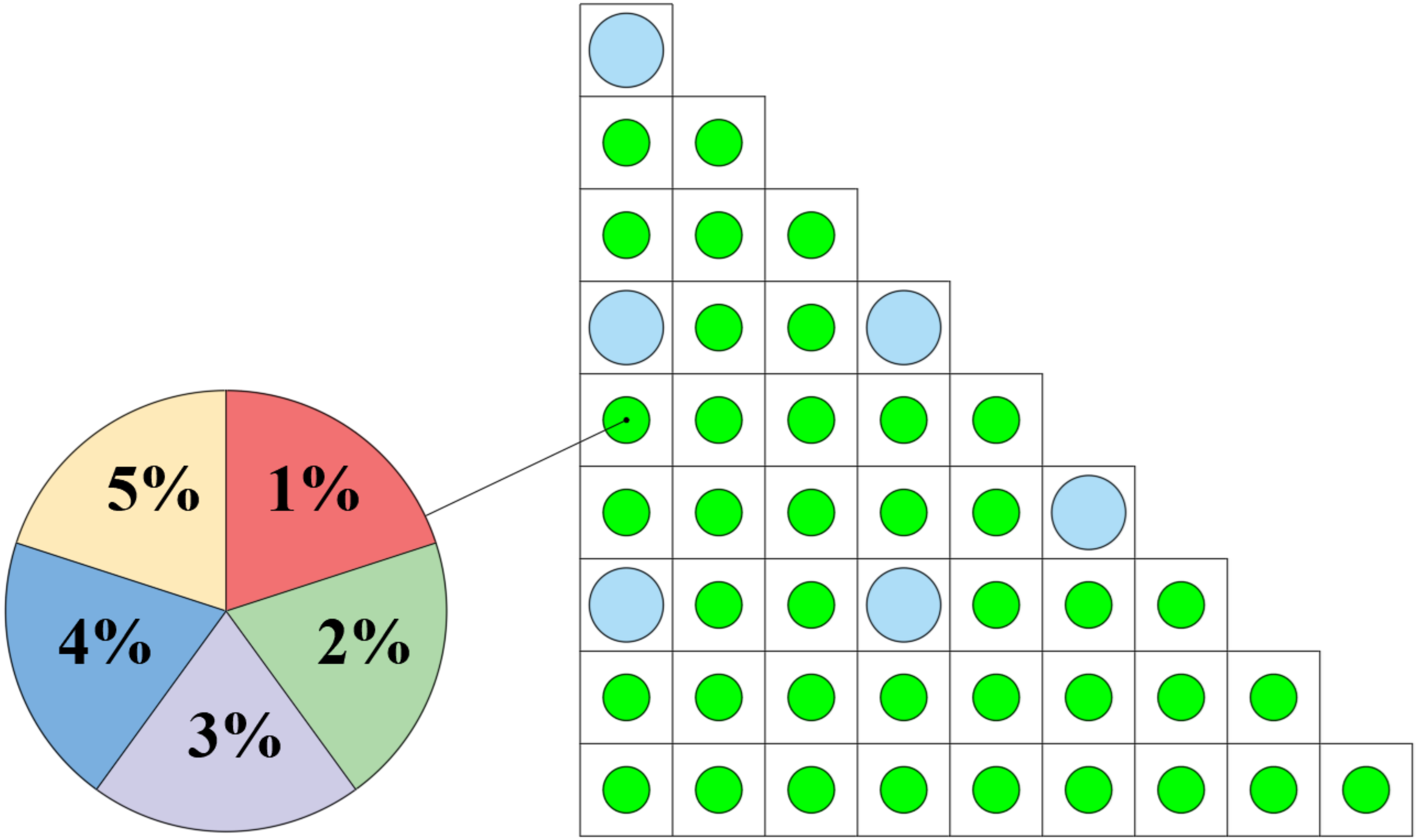
	1.0	5.0	3.0	2.0	5.0	1.0	4.0
1.0	4.0	3.0	3.0	1.0	5.0	3.0	3.0
5.0	3.0	4.0	5.0	3.0	2.0	4.0	3.0
3.0	3.0	5.0			2.0	3.0	3.0
2.0	1.0	3.0			4.0	4.0	1.0
5.0	5.0	2.0	2.0	4.0	3.0	1.0	4.0
1.0	3.0	4.0	3.0	4.0	1.0	3.0	3.0
4.0	3.0	3.0	3.0	1.0	4.0	3.0	3.0



	5.0	5.0		3.0	5.0		4.0	5.0
5.0	5.0	3.0	1.0	5.0	1.0	5.0	1.0	4.0
5.0	3.0	5.0	5.0	5.0	2.0	1.0	3.0	5.0
	1.0	5.0		5.0	2.0		4.0	1.0
3.0	5.0	5.0	5.0	1.0	3.0	5.0	1.0	5.0
5.0	1.0	2.0	2.0	3.0		5.0	5.0	3.0
	5.0	1.0		5.0	5.0	5.0	4.0	5.0
4.0	1.0	3.0	4.0	1.0	5.0	4.0	2.0	1.0
5.0	4.0	5.0	1.0	5.0	3.0	5.0	1.0	5.0

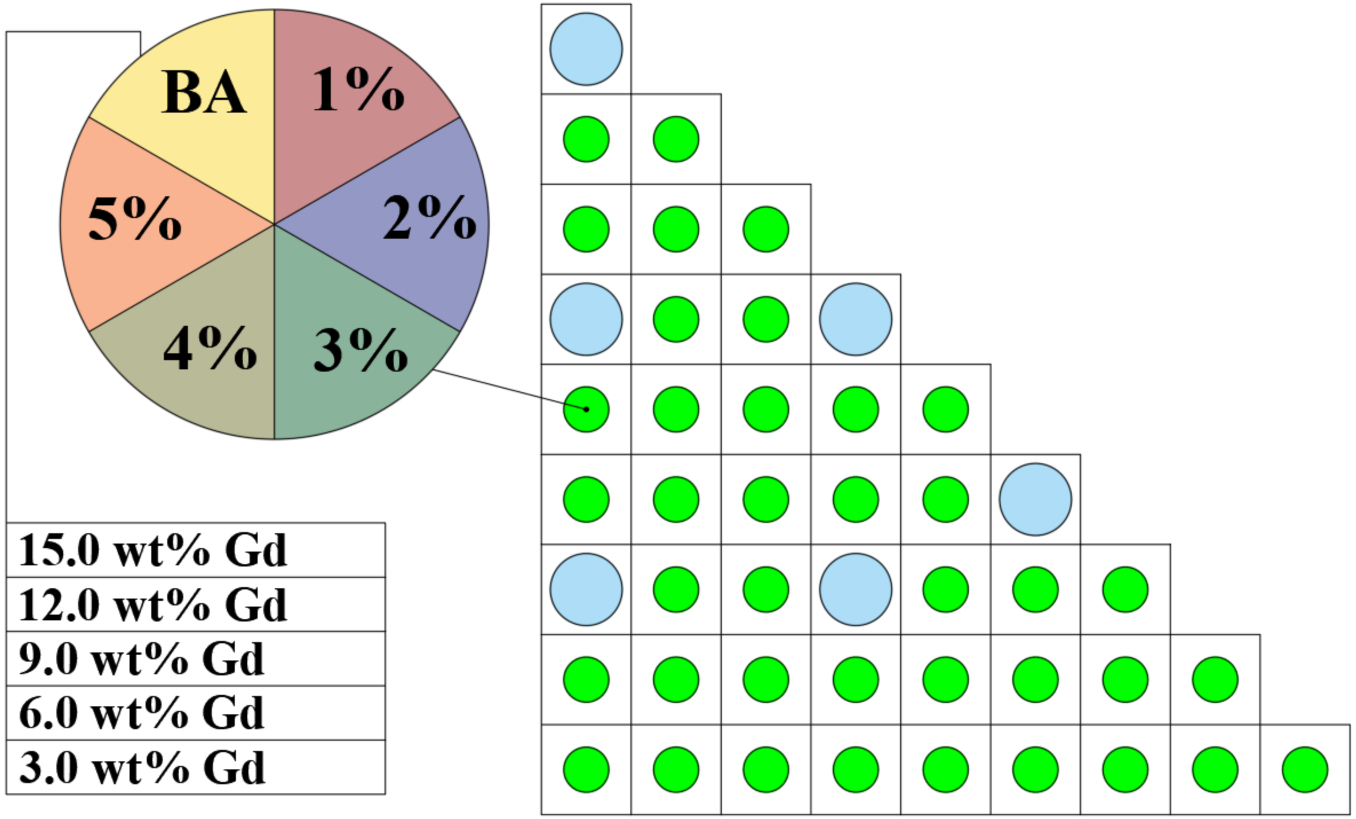
Uniformity of pin enrichment

- **In the training dataset, each fuel pin has the same number of occurrences for each enrichment:**
  - **1%**
  - **2%**
  - **3%**
  - **4%**
  - **5%**



Data with Gd fuel

- **Equal numbers of each burnable absorber concentration were used in generating training data.**
  - **The testing data was based on a wider scope non-fixed points in between 1% and 18% Gd.**

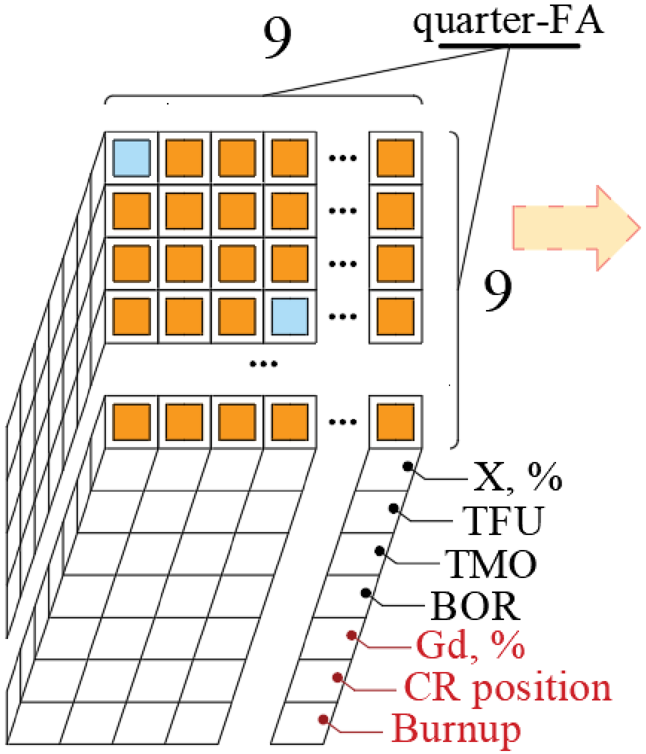


15.0 wt% Gd
12.0 wt% Gd
9.0 wt% Gd
6.0 wt% Gd
3.0 wt% Gd

Neural Network Model

- **A 2D Convolutional Neural Network (CNN)**

Input shape



ANN Model

Convolution layer 2D
ReLU, 128 filters

Convolution layer 2D
ReLU, 128 filters

Flattening Layer

Dense Layer #1
eLU, 6296* neurons

Dense Layer #2
eLU, 6296 neurons

Dense Layer #3
eLU, 6296 neurons

Dense Layer #4
eLU, 6296 neurons

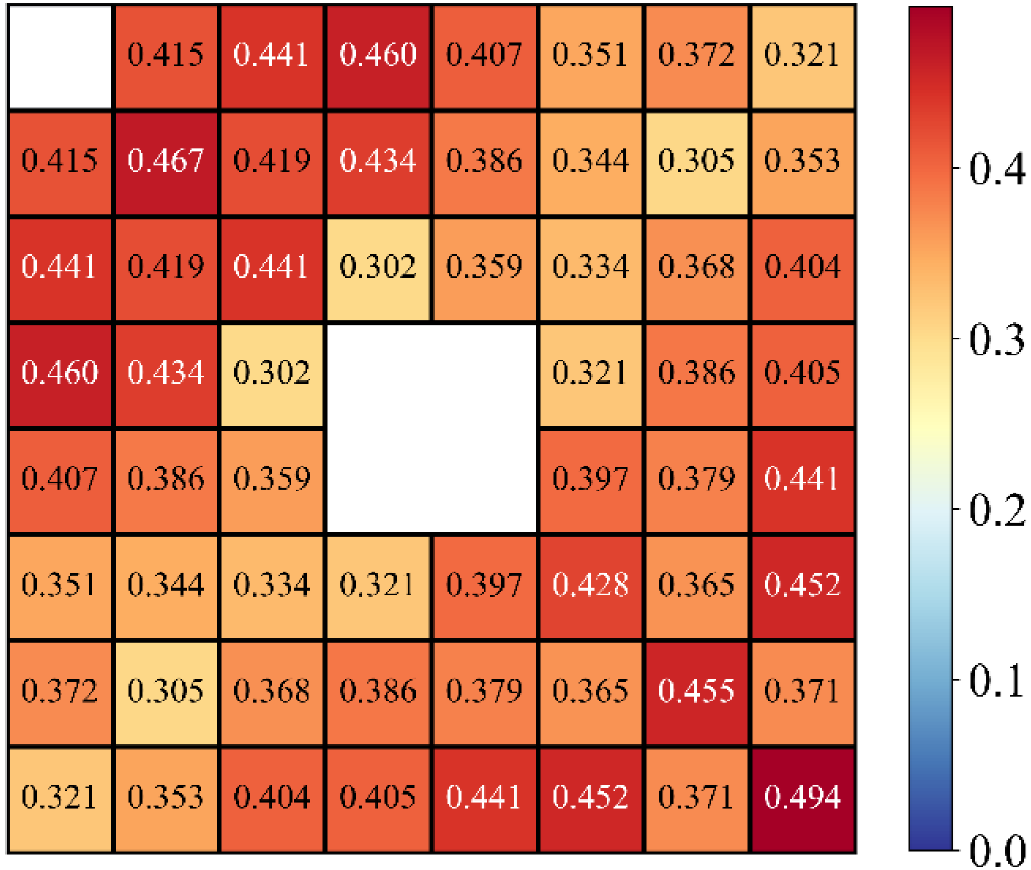
Dense Layer #5
ReLU, 1574 neurons

* 6,296 neurons per layer is 1,574 neurons in the output layer multiplied by 4

Macroscopic XS performance – heterogeneous form function

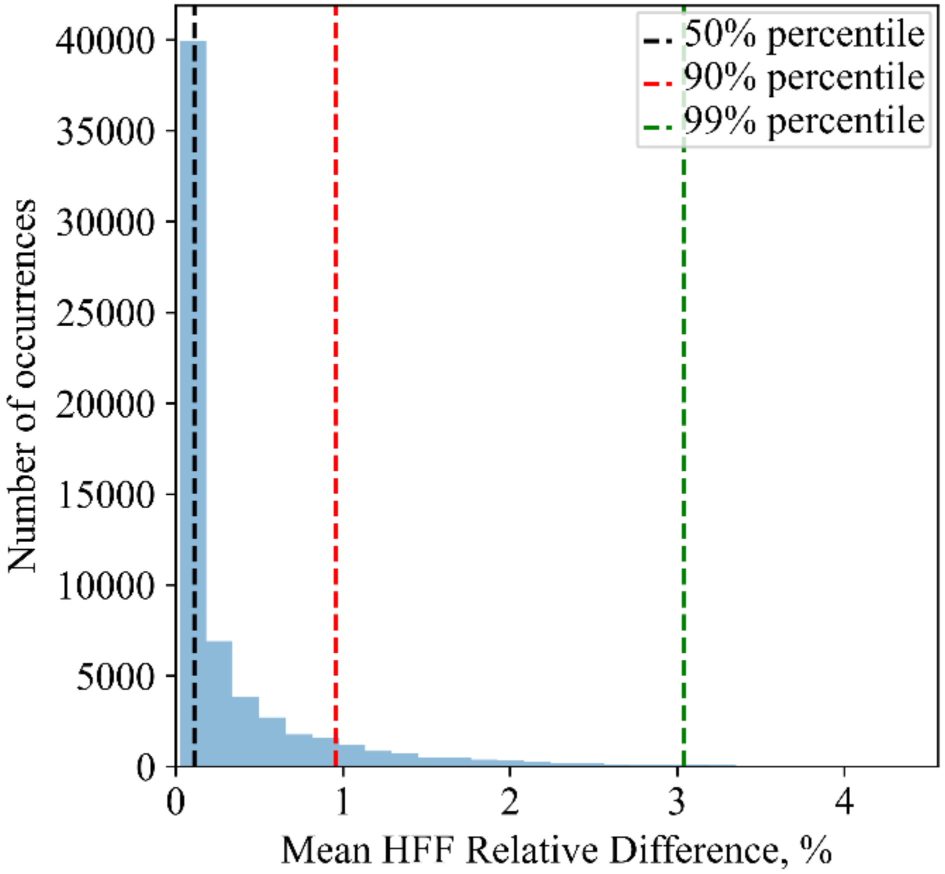
- **Statistics for XSNET output for all 62,806 test samples.**

Average RD values, %



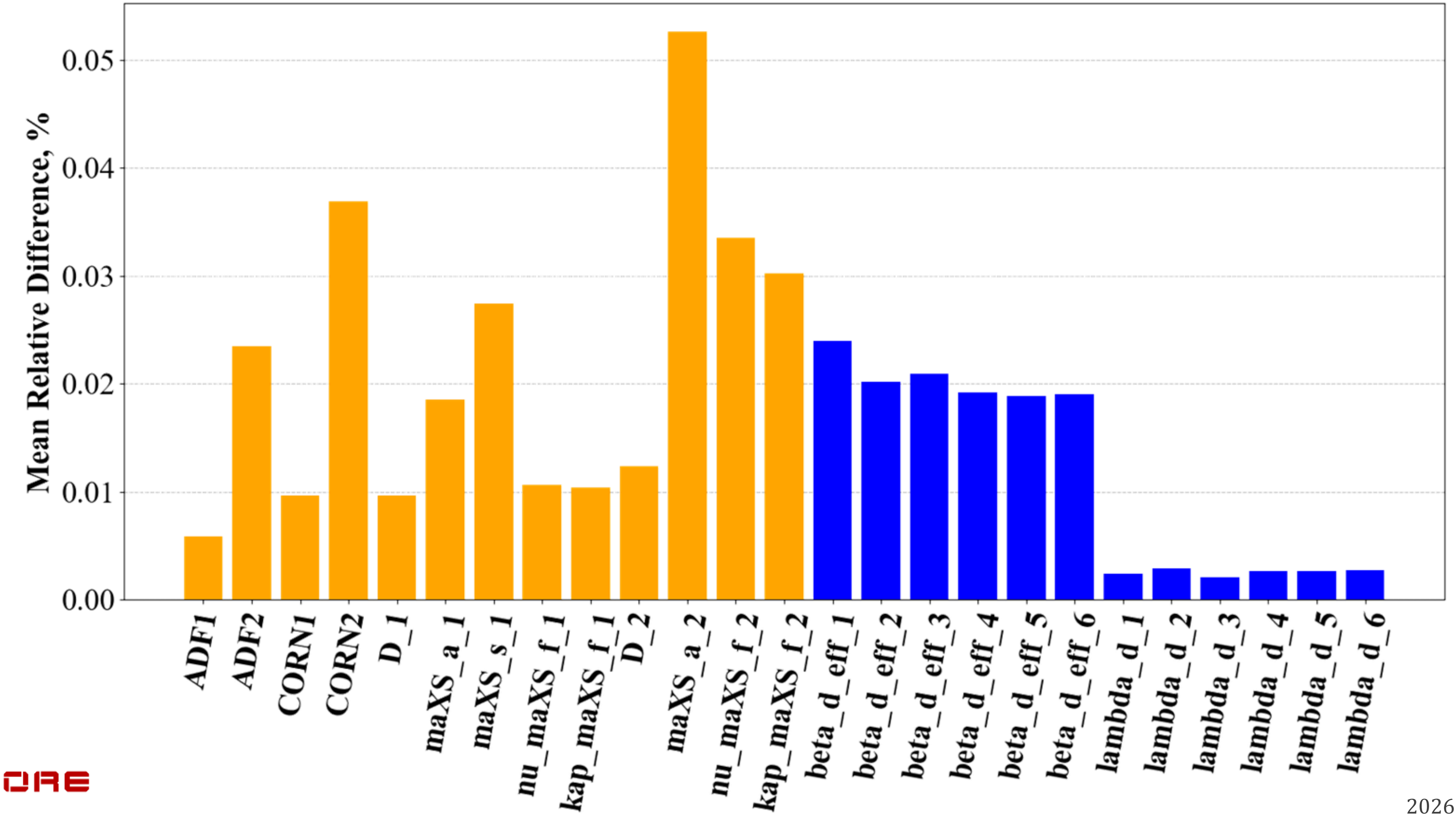
	0.415	0.441	0.460	0.407	0.351	0.372	0.321
0.415	0.467	0.419	0.434	0.386	0.344	0.305	0.353
0.441	0.419	0.441	0.302	0.359	0.334	0.368	0.404
0.460	0.434	0.302			0.321	0.386	0.405
0.407	0.386	0.359			0.397	0.379	0.441
0.351	0.344	0.334	0.321	0.397	0.428	0.365	0.452
0.372	0.305	0.368	0.386	0.379	0.365	0.455	0.371
0.321	0.353	0.404	0.405	0.441	0.452	0.371	0.494

Distribution of FA-averaged pin power (HFF) RD



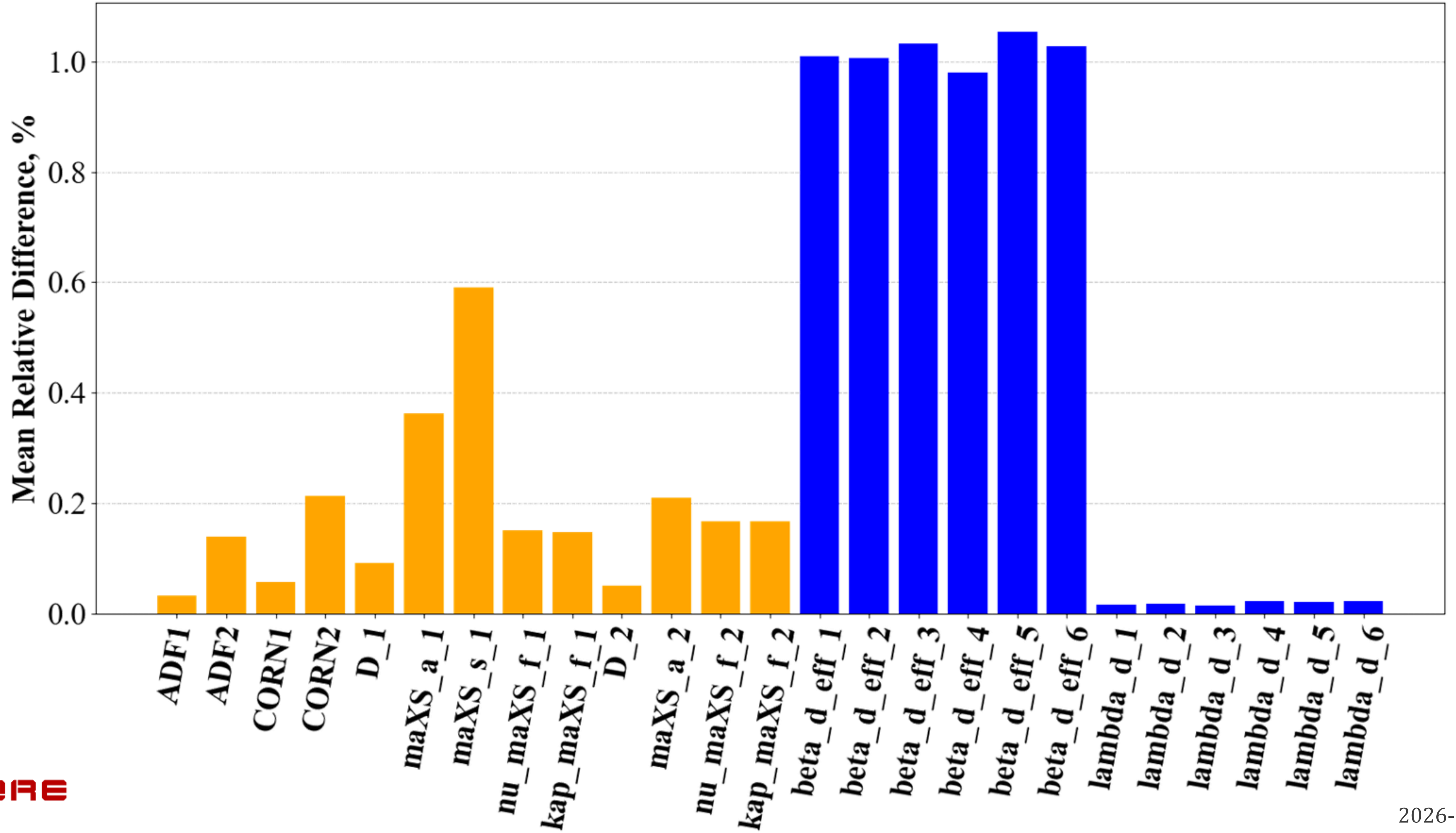
Cross-section data and kinetic parameters

- **16x16 FA geometry without Gd pins - Mean Relative Difference (%).**



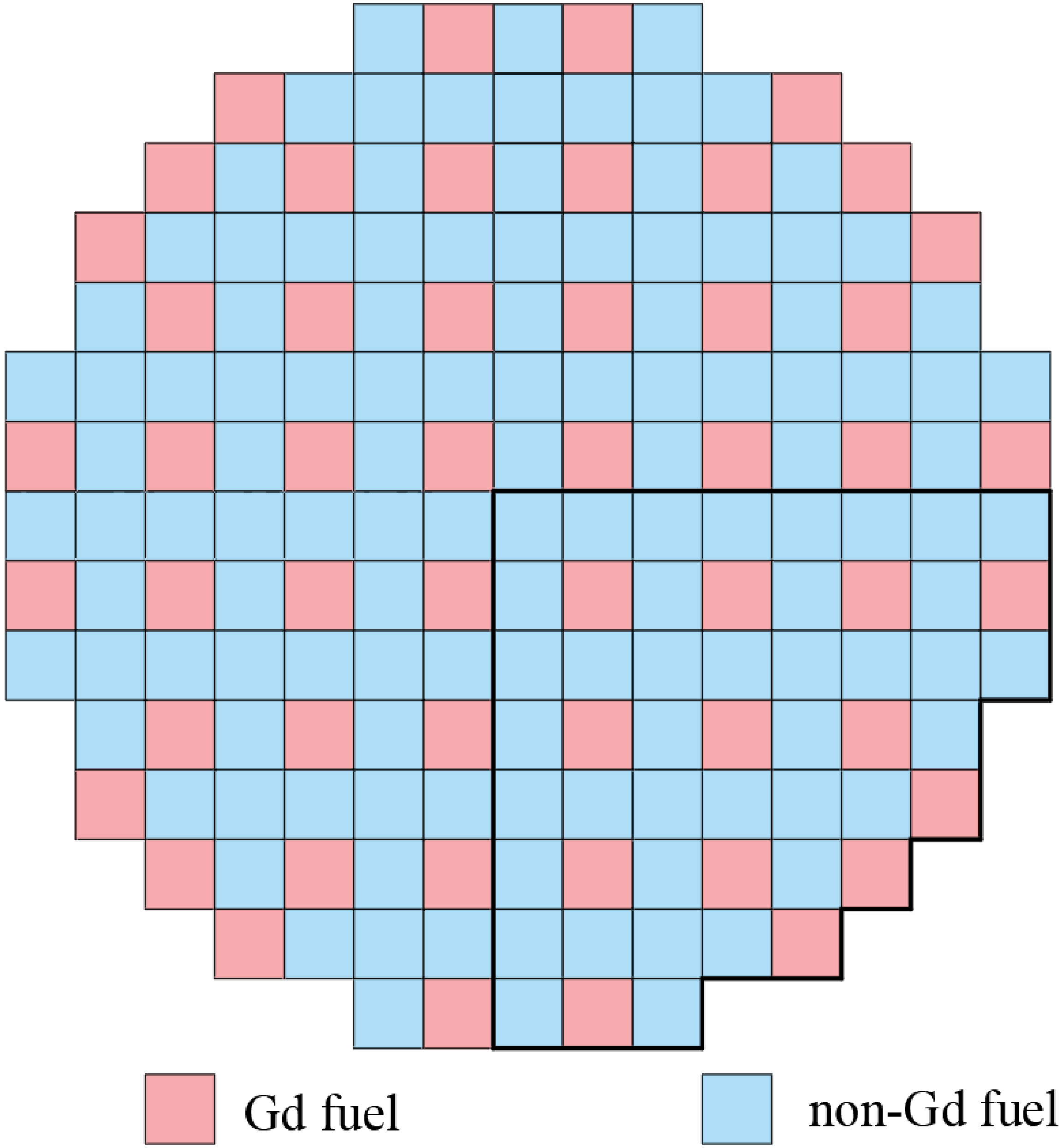
Cross-section data and kinetic parameters

- **16x16 FA geometry with Gd pins - Mean Relative Difference (%).**



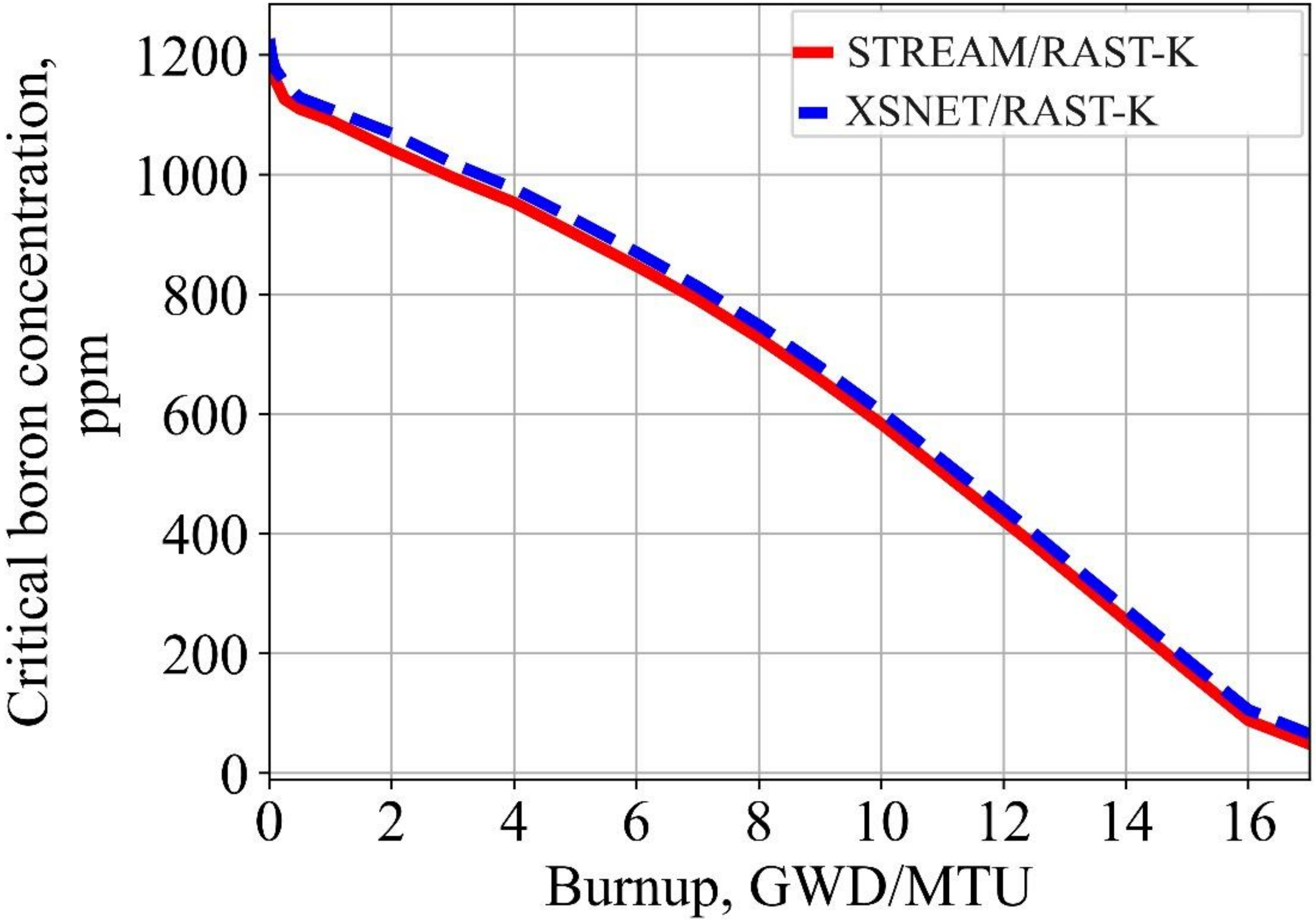
Steady-state analysis – OPR-1000 reactor

- **A full-core 3D OPR-1000 model was prepared with the following general arrangement.**



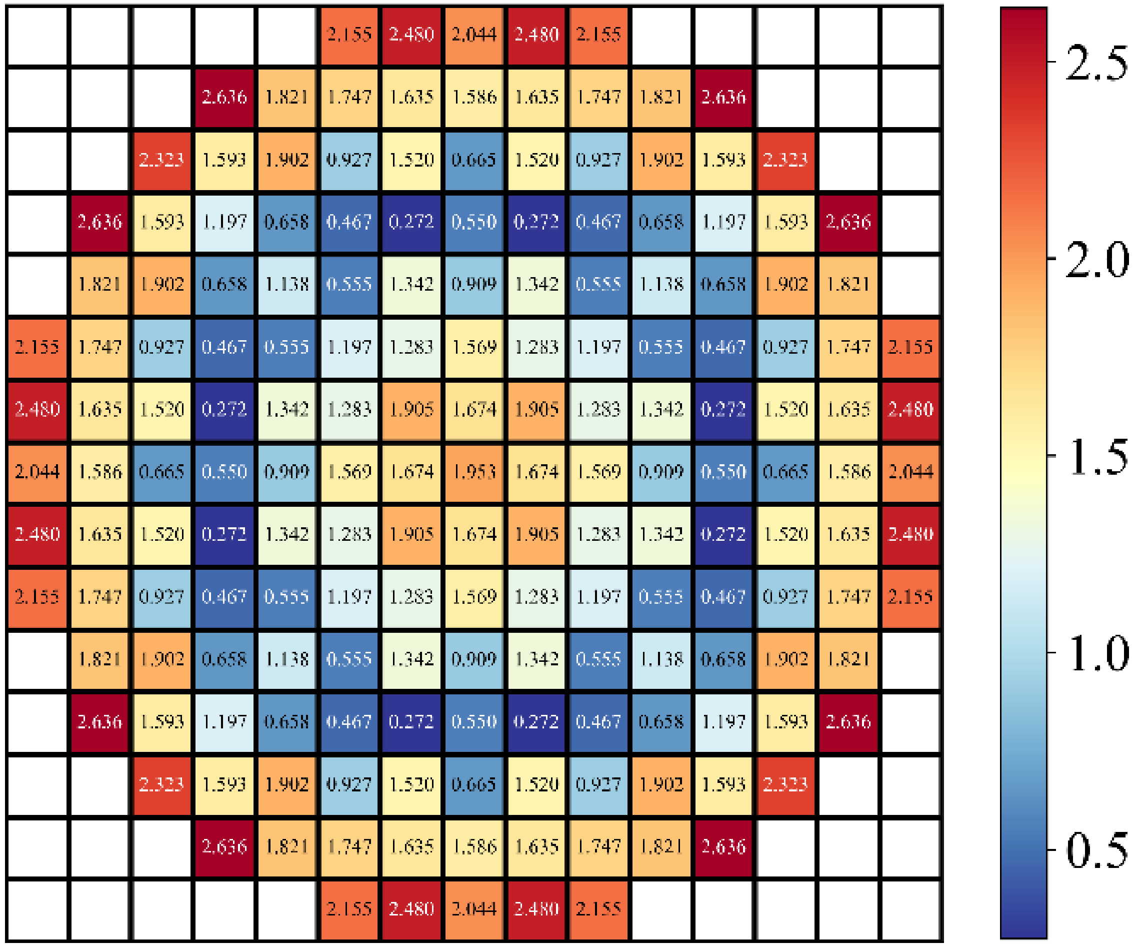
Core-wise stats – critical boron concentration

- **Averaged across all tested samples.**



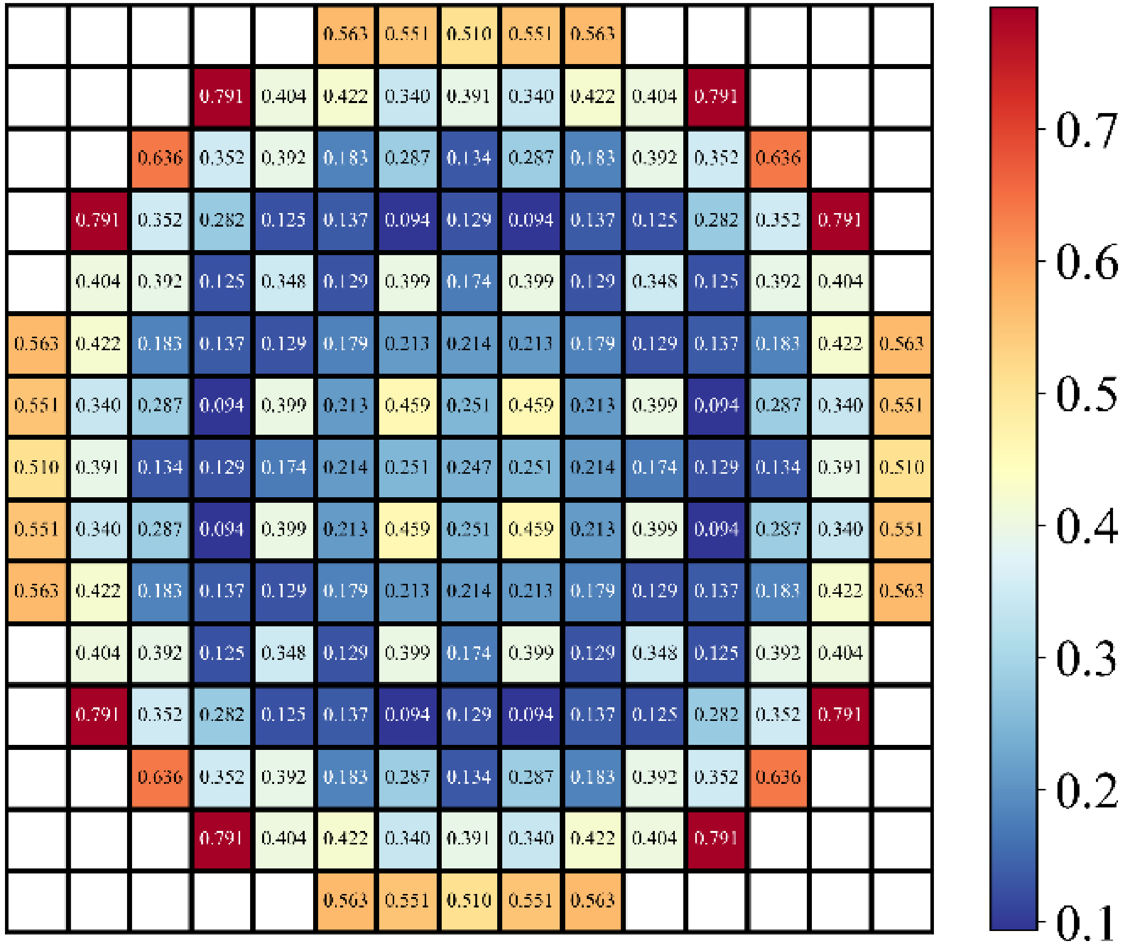
Core-wise stats – assembly power (axially-averaged), MAPE, %

Assembly **power at BOC** (0.0 GWd/MTU)



					2.155	2.480	2.044	2.480	2.155					
			2.636	1.821	1.747	1.635	1.586	1.635	1.747	1.821	2.636			
		2.323	1.593	1.902	0.927	1.520	0.665	1.520	0.927	1.902	1.593	2.323		
	2.636	1.593	1.197	0.658	0.467	0.272	0.550	0.272	0.467	0.658	1.197	1.593	2.636	
	1.821	1.902	0.658	1.138	0.555	1.342	0.909	1.342	0.555	1.138	0.658	1.902	1.821	
2.155	1.747	0.927	0.467	0.555	1.197	1.283	1.569	1.283	1.197	0.555	0.467	0.927	1.747	2.155
2.480	1.635	1.520	0.272	1.342	1.283	1.905	1.674	1.905	1.283	1.342	0.272	1.520	1.635	2.480
2.044	1.586	0.665	0.550	0.909	1.569	1.674	1.953	1.674	1.569	0.909	0.550	0.665	1.586	2.044
2.480	1.635	1.520	0.272	1.342	1.283	1.905	1.674	1.905	1.283	1.342	0.272	1.520	1.635	2.480
2.155	1.747	0.927	0.467	0.555	1.197	1.283	1.569	1.283	1.197	0.555	0.467	0.927	1.747	2.155
	1.821	1.902	0.658	1.138	0.555	1.342	0.909	1.342	0.555	1.138	0.658	1.902	1.821	
	2.636	1.593	1.197	0.658	0.467	0.272	0.550	0.272	0.467	0.658	1.197	1.593	2.636	
		2.323	1.593	1.902	0.927	1.520	0.665	1.520	0.927	1.902	1.593	2.323		
			2.636	1.821	1.747	1.635	1.586	1.635	1.747	1.821	2.636			
					2.155	2.480	2.044	2.480	2.155					

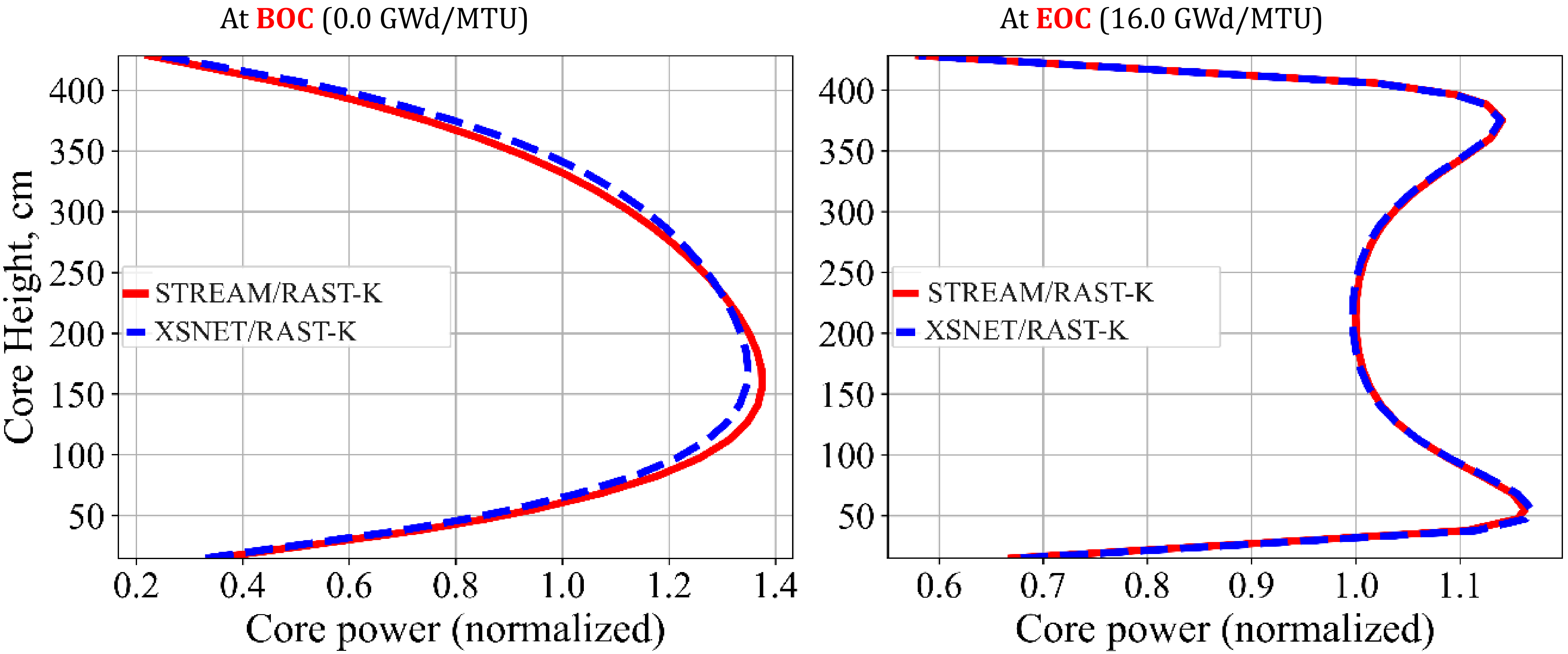
Assembly **burnup at EOC** (16.0 GWd/MTU)



					0.563	0.551	0.510	0.551	0.563					
			0.791	0.404	0.422	0.340	0.391	0.340	0.422	0.404	0.791			
		0.636	0.352	0.392	0.183	0.287	0.134	0.287	0.183	0.392	0.352	0.636		
	0.791	0.352	0.282	0.125	0.137	0.094	0.129	0.094	0.137	0.125	0.282	0.352	0.791	
	0.404	0.392	0.125	0.348	0.129	0.399	0.174	0.399	0.129	0.348	0.125	0.392	0.404	
0.563	0.422	0.183	0.137	0.129	0.179	0.213	0.214	0.213	0.179	0.129	0.137	0.183	0.422	0.563
0.551	0.340	0.287	0.094	0.399	0.213	0.459	0.251	0.459	0.213	0.399	0.094	0.287	0.340	0.551
0.510	0.391	0.134	0.129	0.174	0.214	0.251	0.247	0.251	0.214	0.174	0.129	0.134	0.391	0.510
0.551	0.340	0.287	0.094	0.399	0.213	0.459	0.251	0.459	0.213	0.399	0.094	0.287	0.340	0.551
0.563	0.422	0.183	0.137	0.129	0.179	0.213	0.214	0.213	0.179	0.129	0.137	0.183	0.422	0.563
	0.404	0.392	0.125	0.348	0.129	0.399	0.174	0.399	0.129	0.348	0.125	0.392	0.404	
	0.791	0.352	0.282	0.125	0.137	0.094	0.129	0.094	0.137	0.125	0.282	0.352	0.791	
		0.636	0.352	0.392	0.183	0.287	0.134	0.287	0.183	0.392	0.352	0.636		
			0.791	0.404	0.422	0.340	0.391	0.340	0.422	0.404	0.791			
					0.563	0.551	0.510	0.551	0.563					

Core-wise stats – axial power distribution

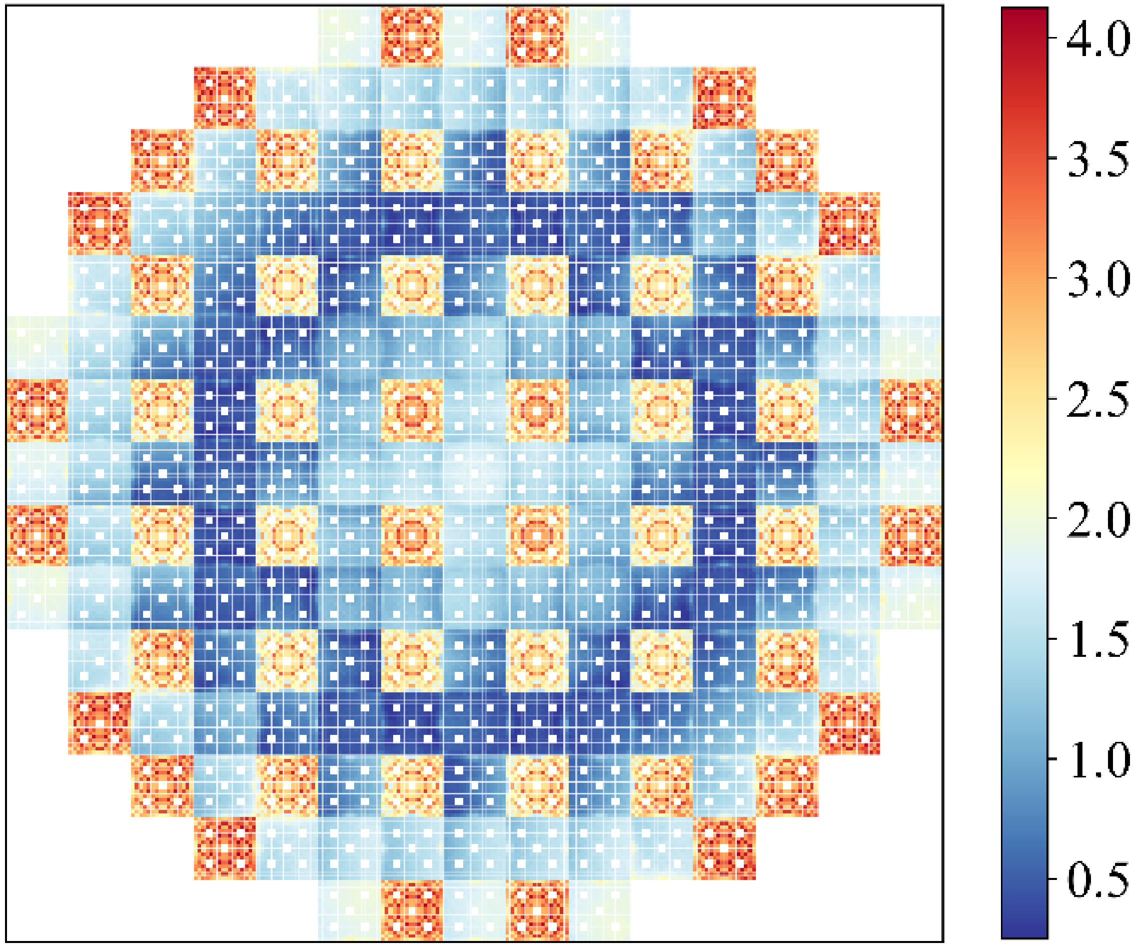
- **Statistics for axial FA power profile.**



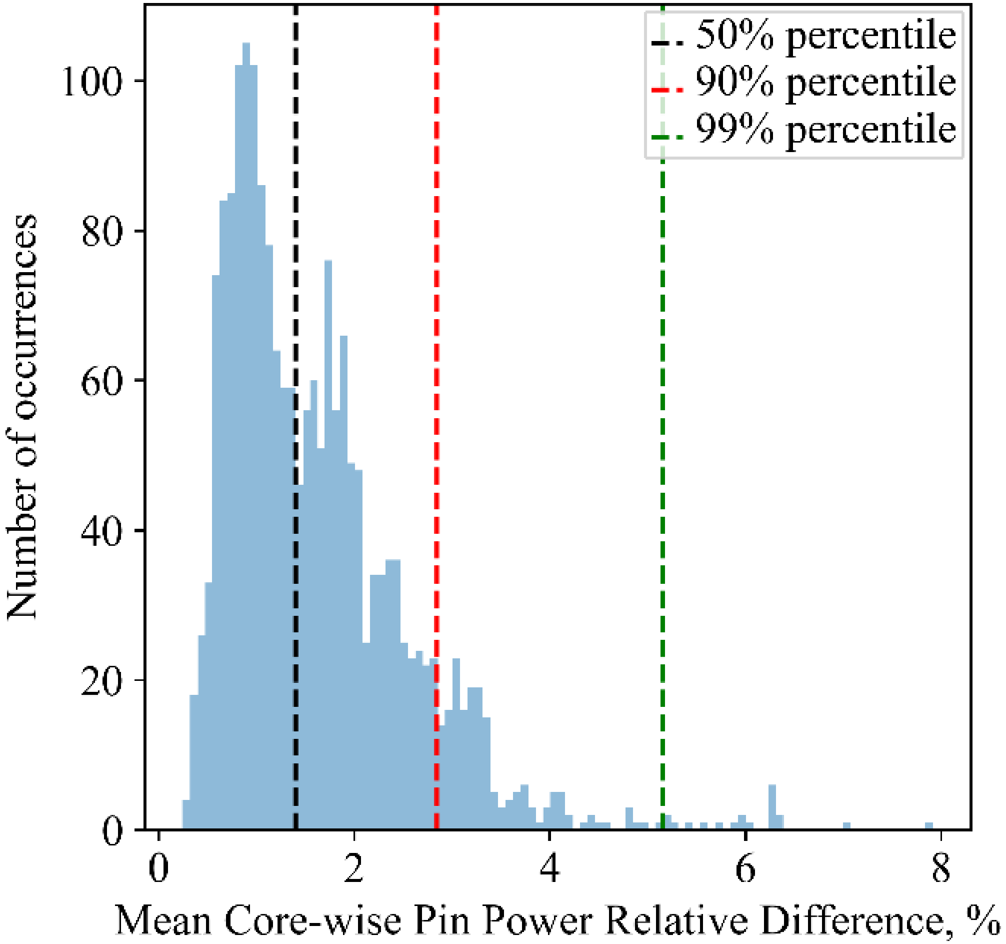
Core-wise stats – pin power (axially-averaged)

- **Statistics for pin power at BOC (0.0 GWd/MTU).**

Distribution of LP-averaged pin power RD, %

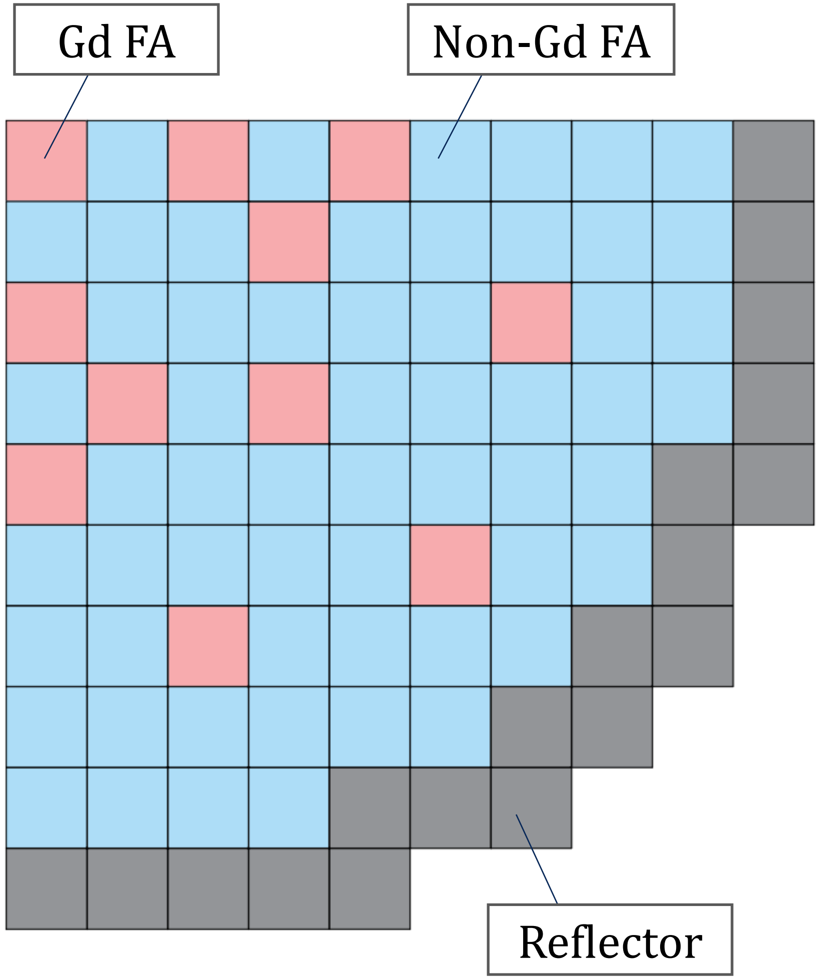


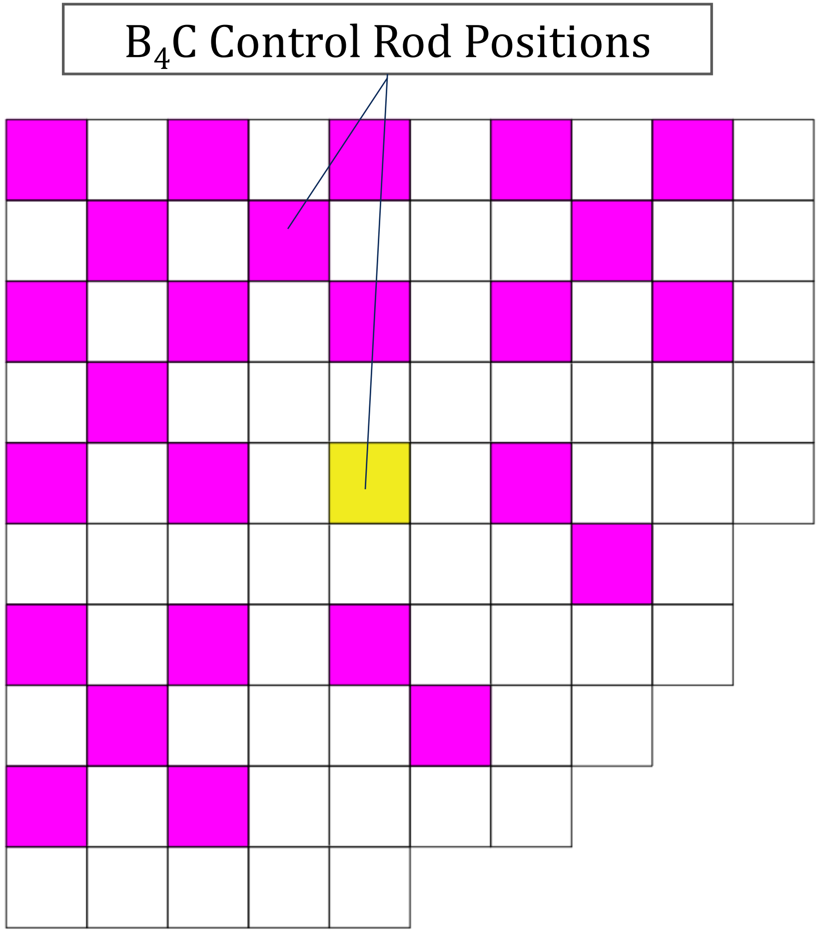
Percentiles of LP-averaged pin power RD



Test Reactor Geometry: APR-1400, fresh 16x16 FA

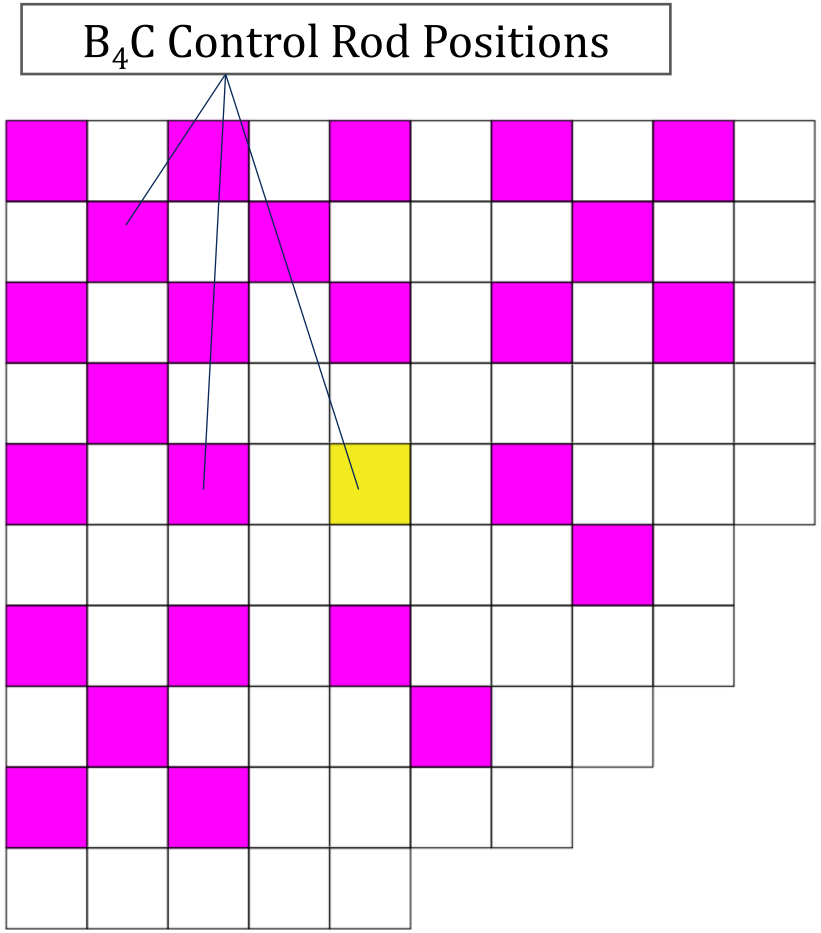
- 3 transient scenarios were modeled and compared with the reference STREAM/RAST-K code system.





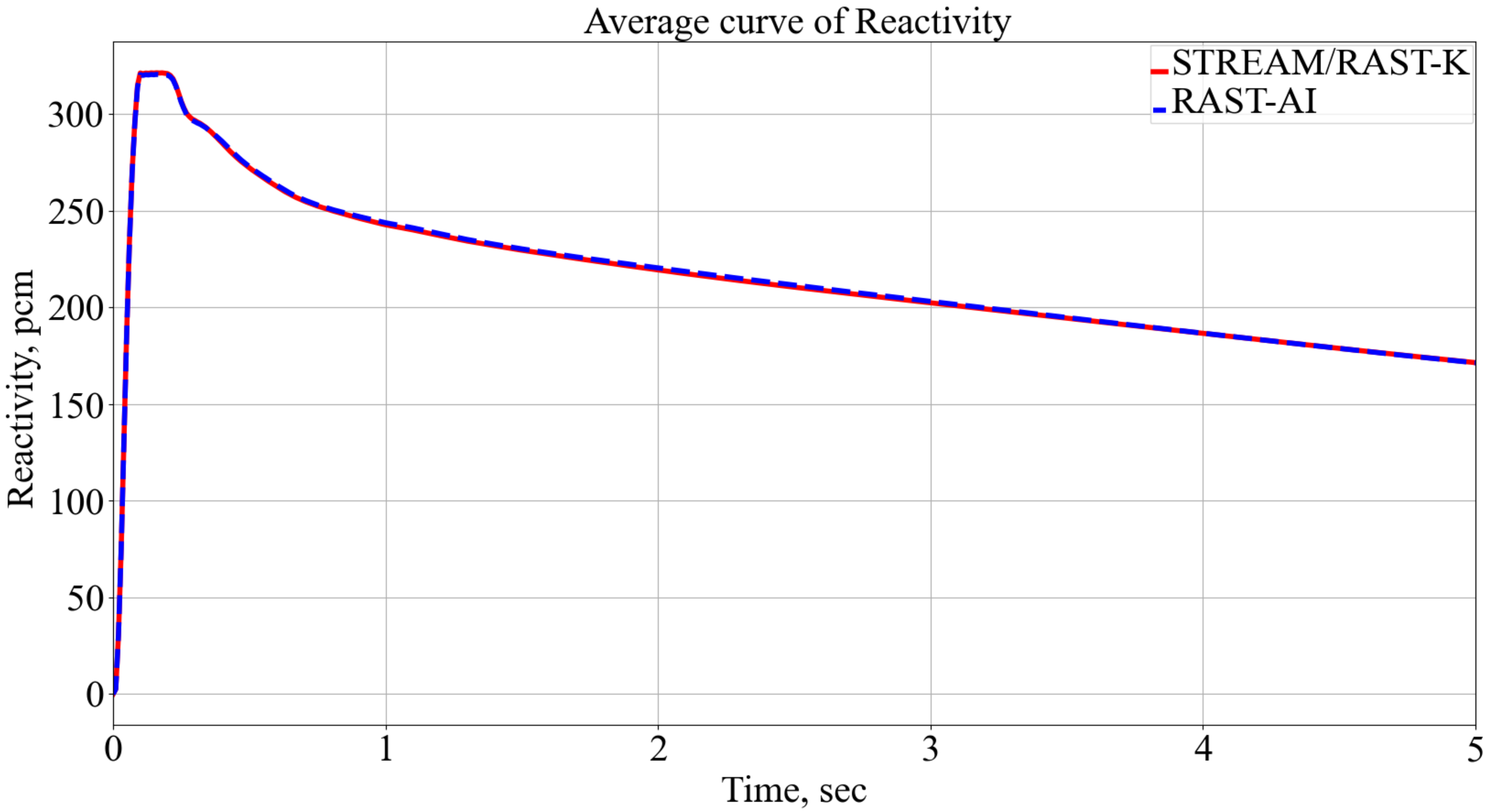
Summary for all tested cases

- **Summary of all cases**



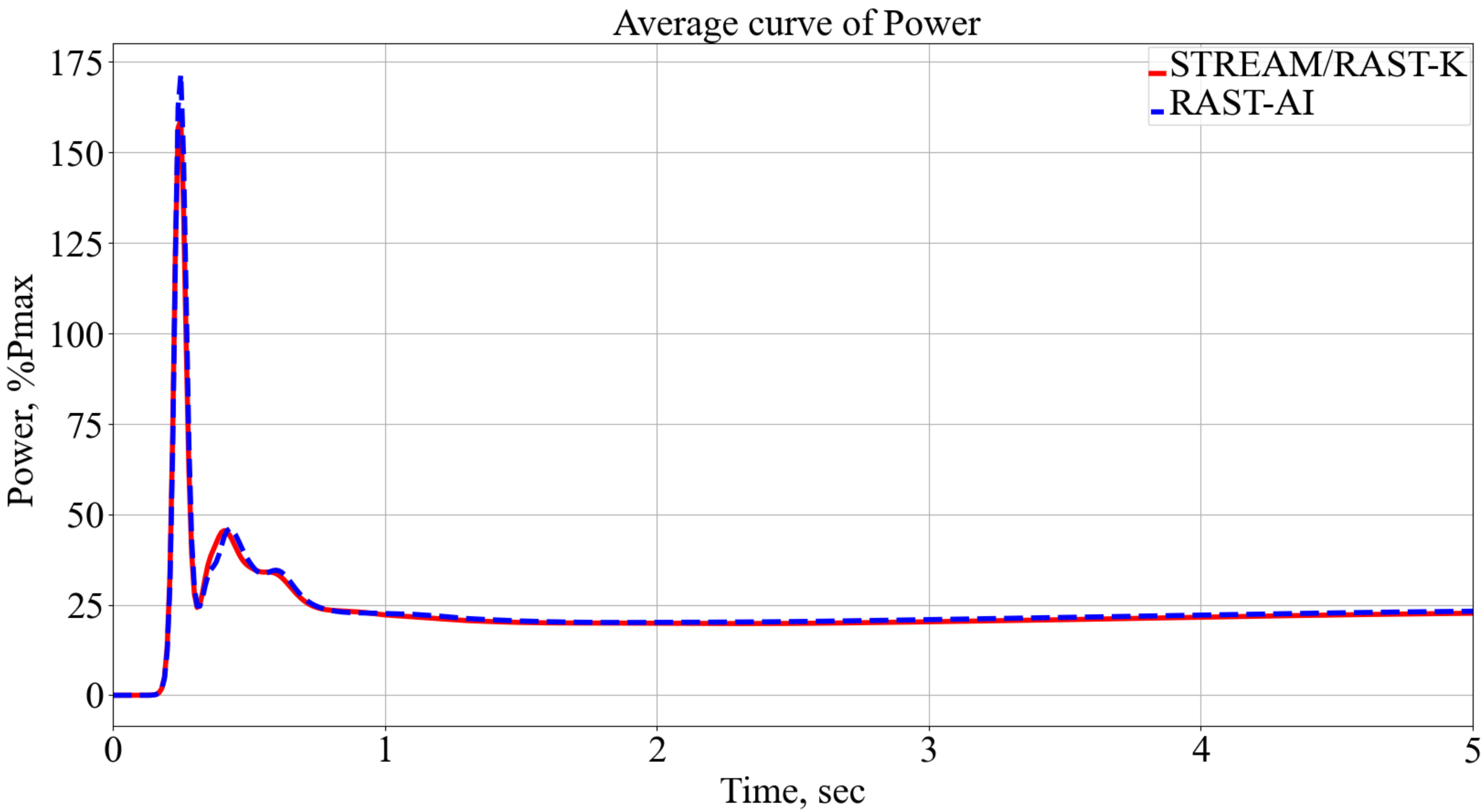
Metric	Reactivity, Δ pcm	Power, Δ $\%P_{max}$	Centerline Fuel Temperature, Δ K
Scenario #1 – **HZP Ejection** of the "yellow" rod			
Mean	0.642	1.175	1.964
Maximum	1.141	14.16	4.085
Median	0.693	0.53	2.03
Scenario #2 – **HFP Ejection** of the "yellow" rod			
Mean	0.951	4.478	4.764
Maximum	1.177	19.275	8.986
Median	0.991	3.547	4.581
Scenario #3 – **HFP Trip** (rapid insertion of all rods)			
Mean	0.788	0.004	1.436
Maximum	1.924	0.028	2.203
Median	0.54	0.002	2.01

Scenario 1 – Averaged output curves



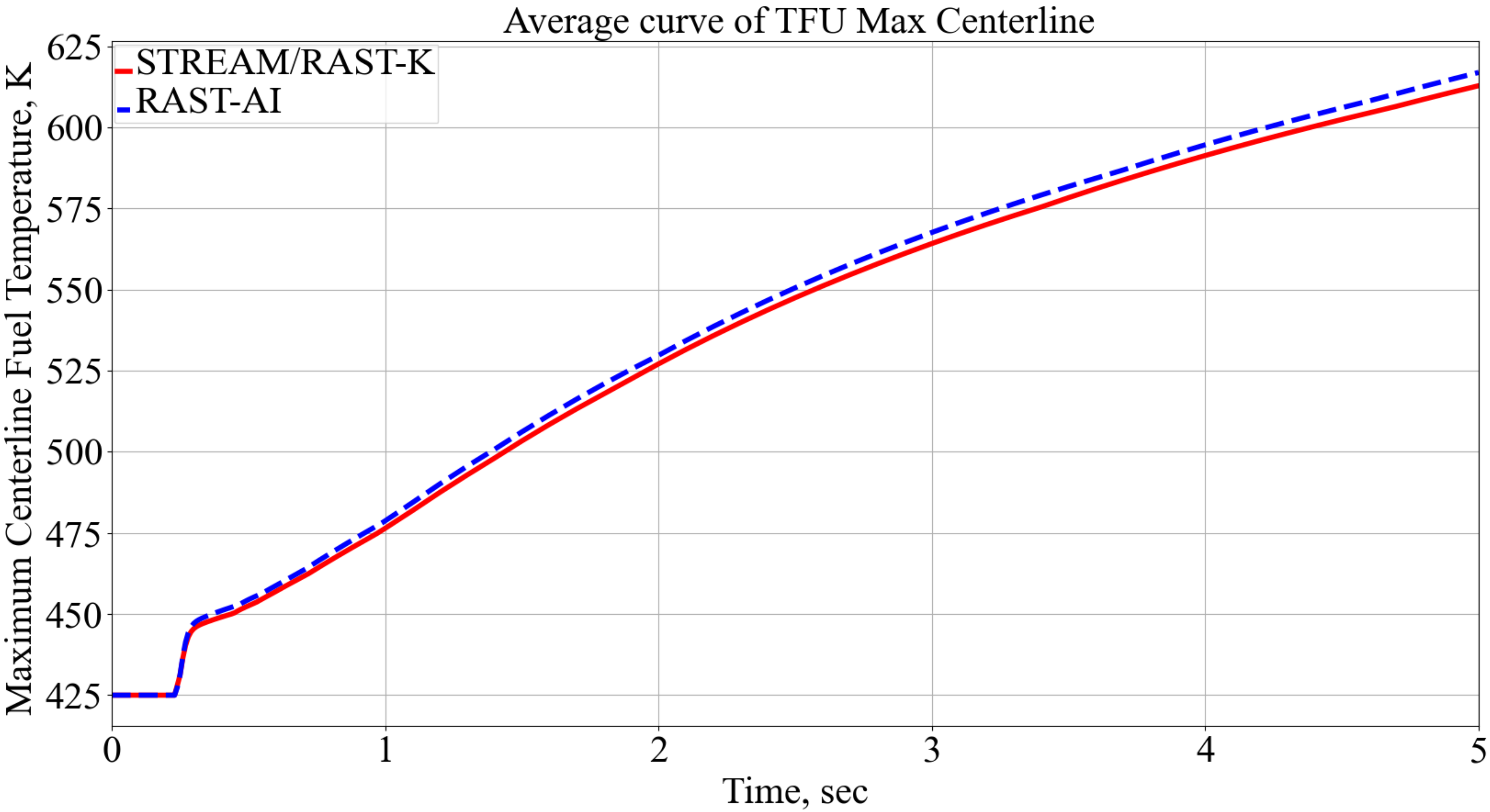
*24 full-core solutions

Scenario 1 – Averaged output curves



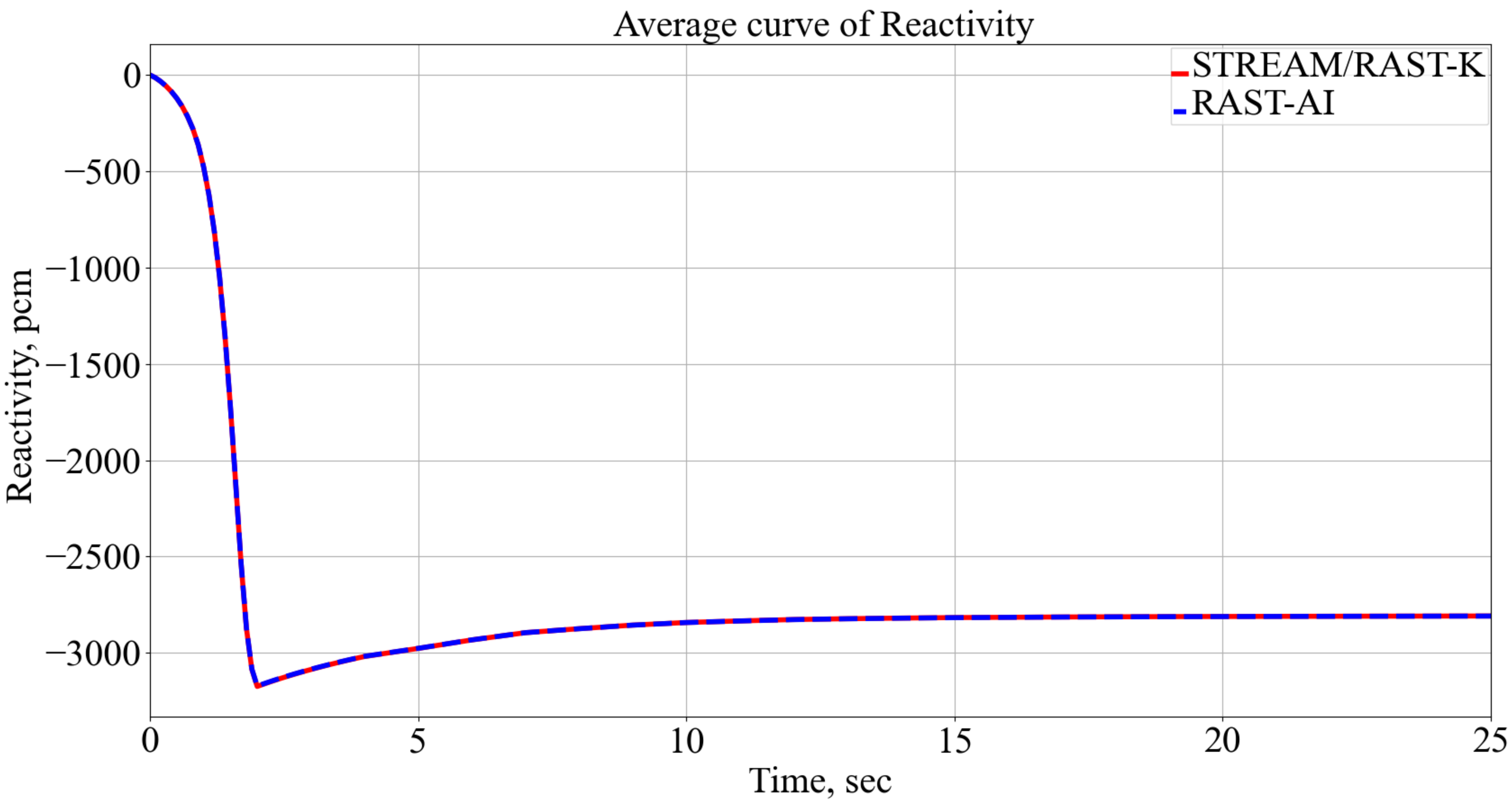
*24 full-core solutions

Scenario 1 – Averaged output curves



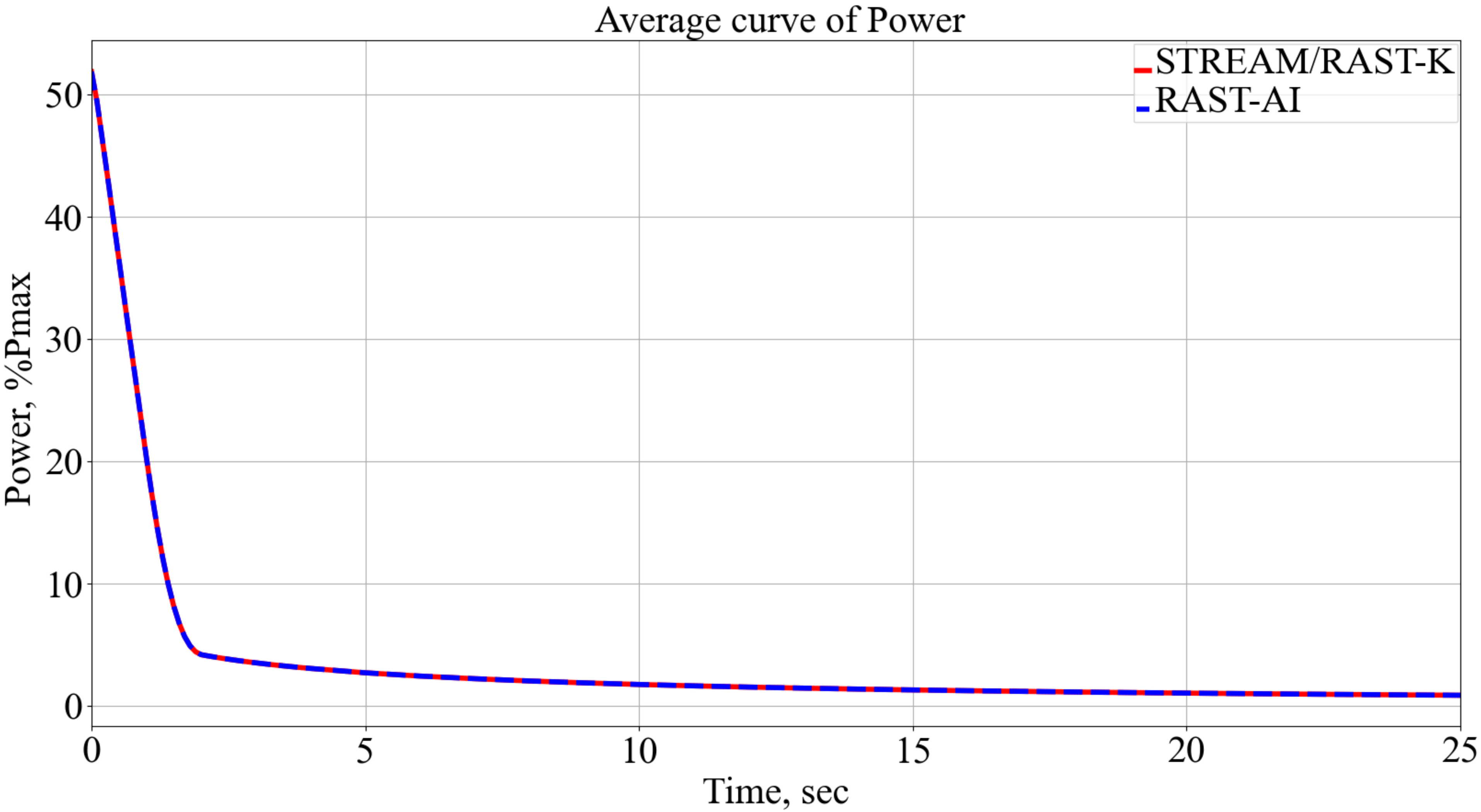
*24 full-core solutions

Scenario 3 – Averaged output curves



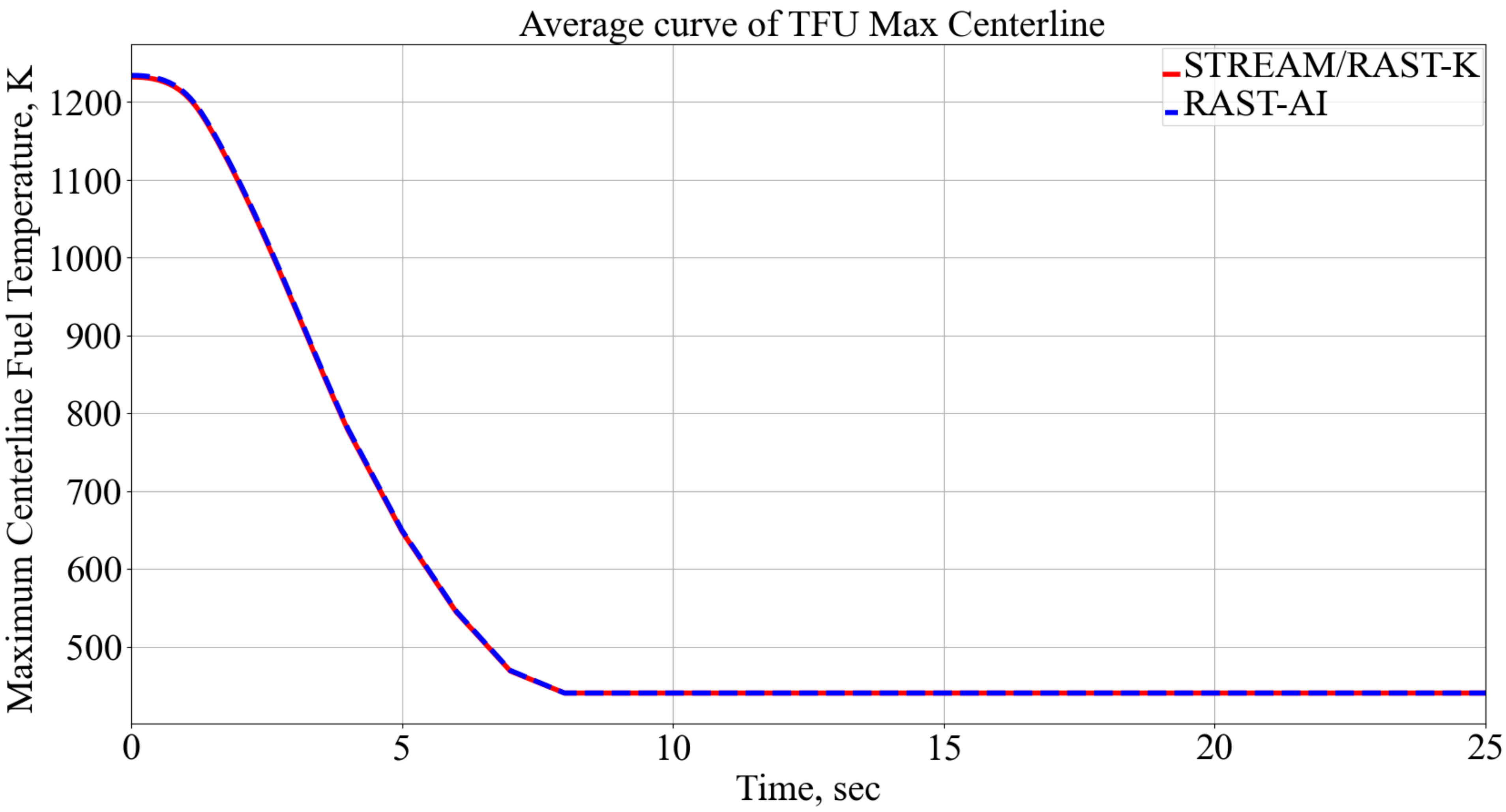
*28 full-core solutions

Scenario 3 – Averaged output curves



*28 full-core solutions

Scenario 3 – Averaged output curves



*28 full-core solutions

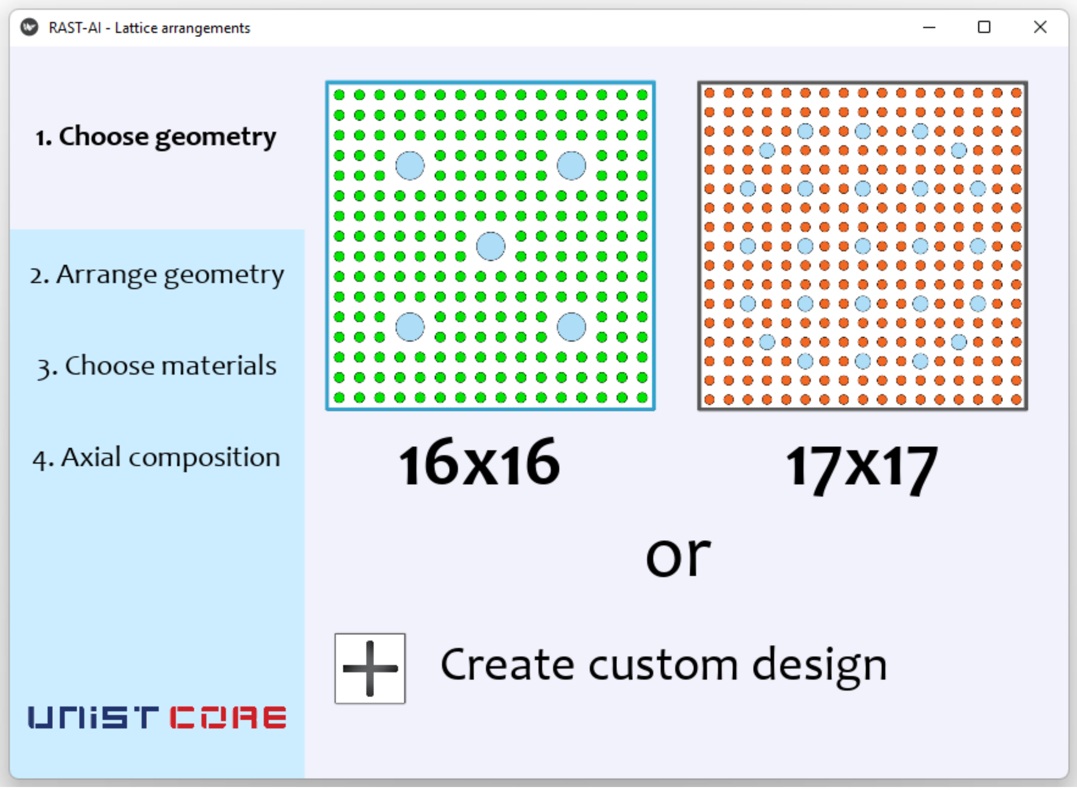
Summary of the achievement

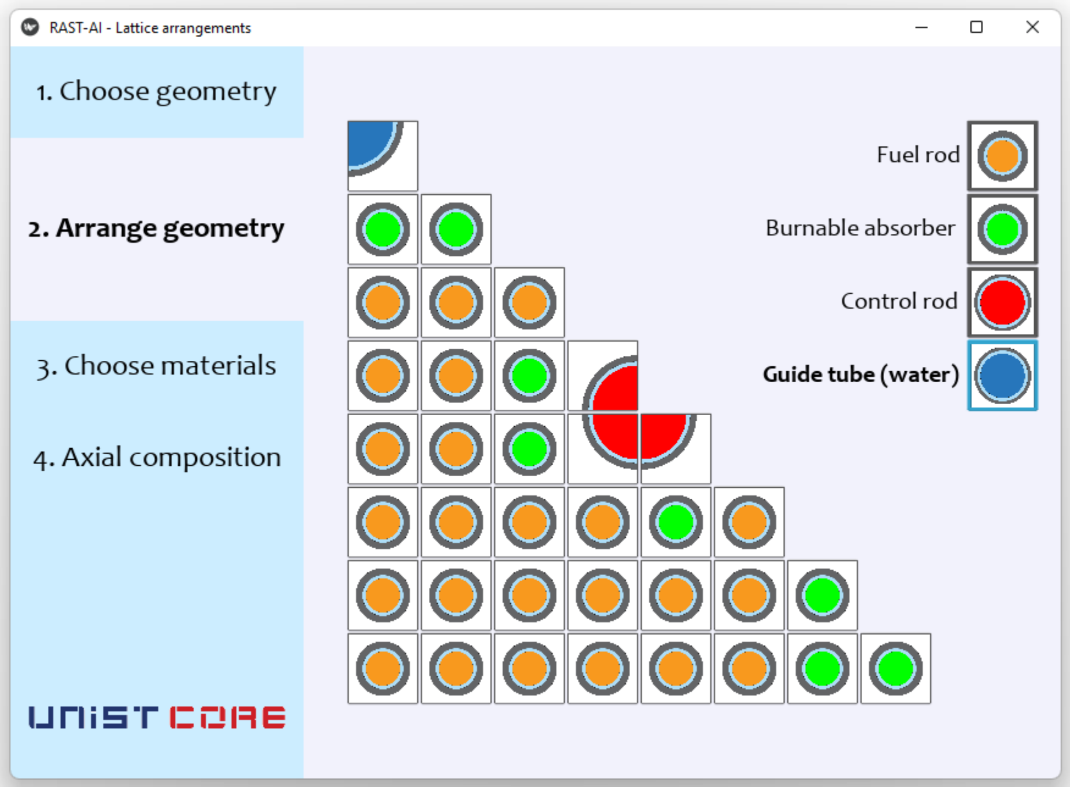
- In a conventional 2-step approximation code system, the main time consumer is the lattice physics code that is used to homogenize the materials and geometry for a nodal diffusion code.
- By replacing the lattice physics calculation, the overall performance improves in range of dozens percent to hundreds of times, depending on the problem type.
  - For a routine task of LP optimization, the speedup is lower compared to the task of designing a new reactor core.
    - This is because in LP optimization, fuel assembly parameters are fixed, whereas in the new reactor core design, they are to vary to achieve desired performance.
  - When designing a new reactor core, or simply modeling a new reactor core, homogenized fuel assembly parameters are not available and must be generated, which leads to a significant performance improvement of XSNET/RAST-K against STREAM/RAST-K.
  - RAST-AI, as well as XSNET/RAST-K are also great teaching tools because they allow getting a detailed reactor simulation solution very quickly and therefore allow for an easy tinkering with the core design inputs to see and learn the impact of changes.

4. Future of AI in reactor simulation

What to expect next

- With the rapid development of mobile devices and their compute capabilities, we expect that initiatives such as RAST-AI will gain further popularity due to them being more understandable to general audience.





Future is in our hands

- **We as developers should always remember that we design our products not only for ourselves but also for future generations of students, researchers and employees who take for granted the convenience of modern portable devices and the speed of AI-enhanced software.**

Contacts

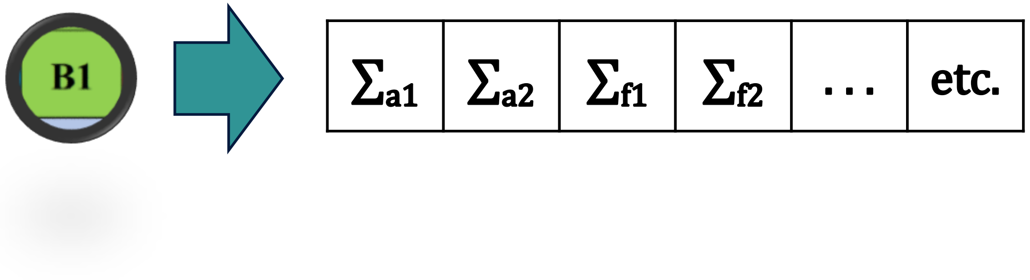
- **For further inquiries, please contact us!**

Website:	https://reactorcore.unist.ac.kr/
Email:	
Prof. Deokjung Lee	deokjung@unist.ac.kr
Dr. Siarhei Dzianisau	s.dzianisau@unist.ac.kr

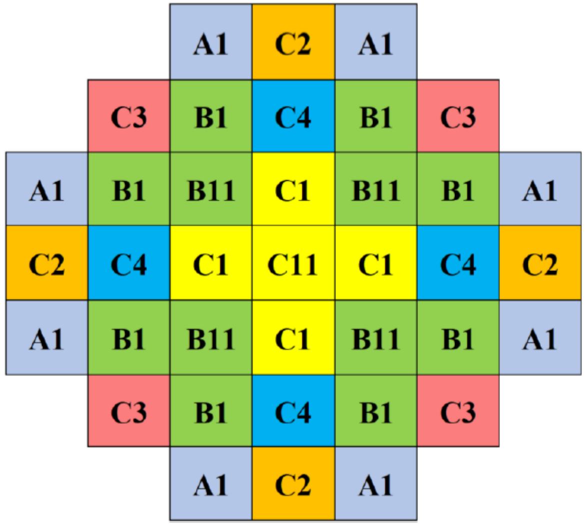
UNIST CORE

Problem statement

- **More questions to be answered:**
  - **digital noise in photos vs noise in LP?**
  - **importance of one single pixel in photos vs one single "pixel" in LP?**
  - **horizontal convolution vs vertical convolution – same or different?**

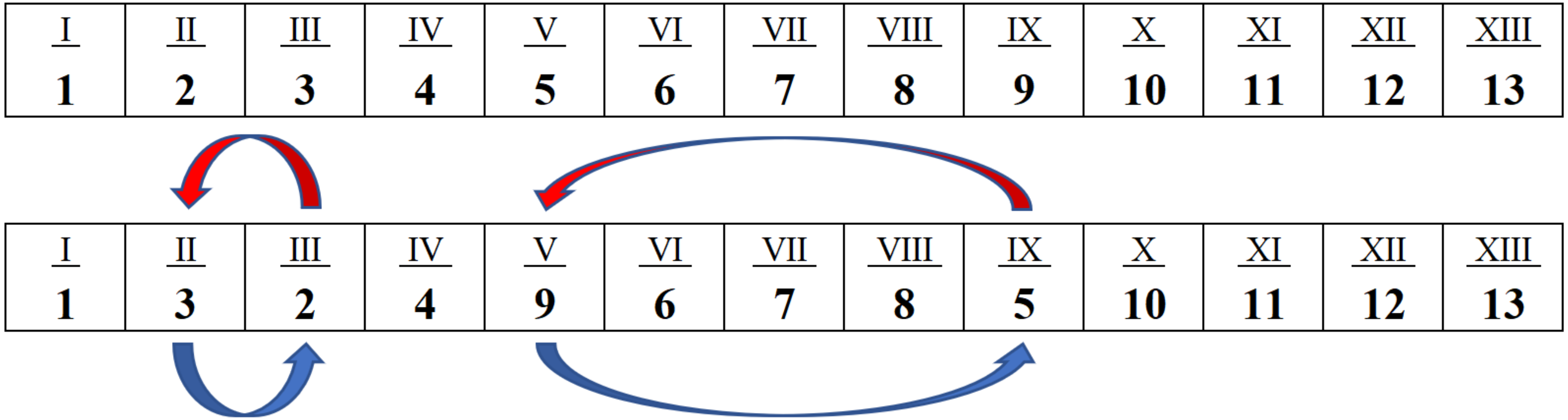


Σ_{a1}	Σ_{a2}	Σ_{f1}	Σ_{f2}	. . .	etc.



		A1	C2	A1		
	C3	B1	C4	B1	C3	
A1	B1	B11	C1	B11	B1	A1
C2	C4	C1	C11	C1	C4	C2
A1	B1	B11	C1	B11	B1	A1
	C3	B1	C4	B1	C3	
		A1	C2	A1		

Shuffling of data in Barcode Model



I	II	III	IV	V	VI	VII	VIII	IX	X	XI	XII	XIII
1	2	3	4	5	6	7	8	9	10	11	12	13

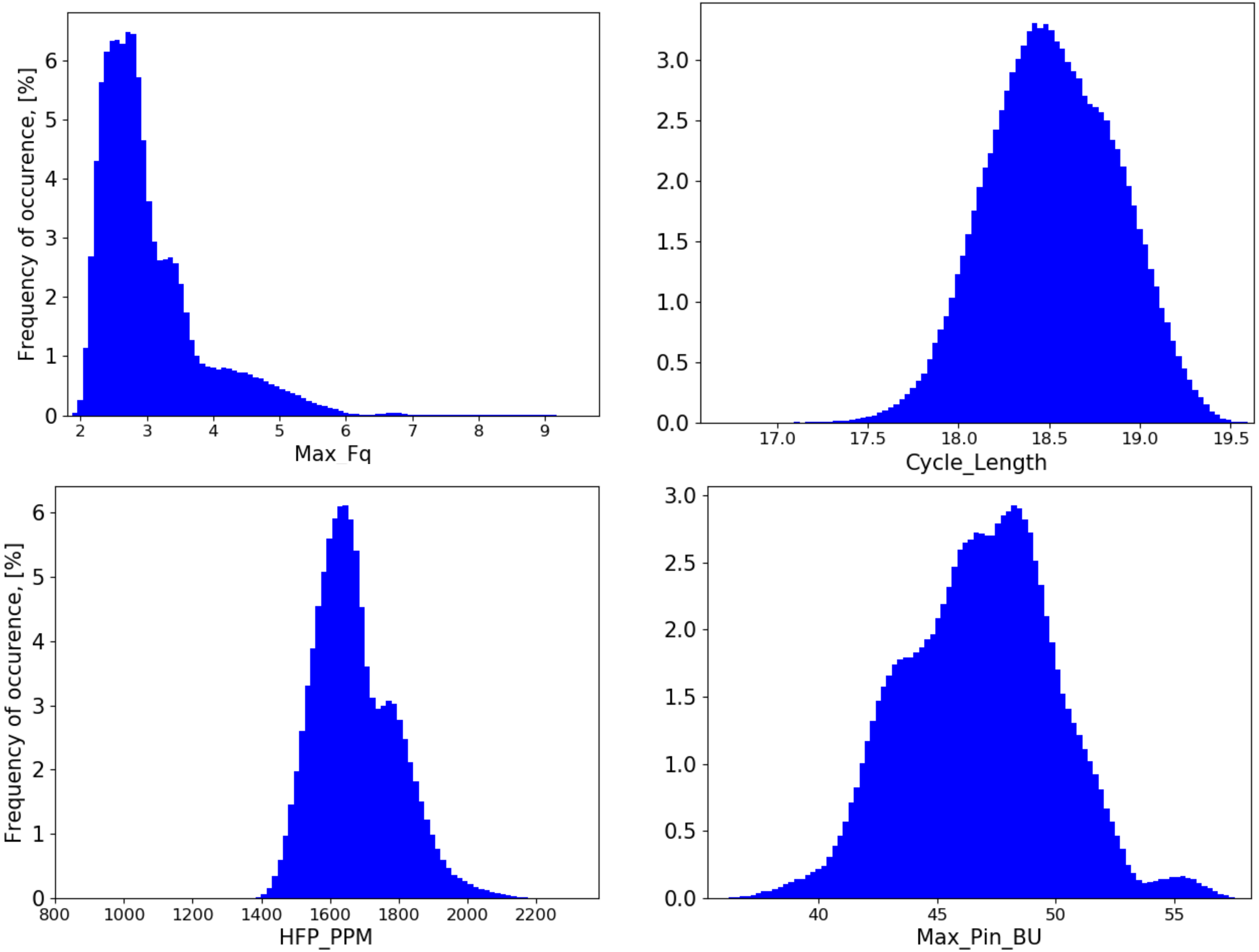
I	II	III	IV	V	VI	VII	VIII	IX	X	XI	XII	XIII
1	3	2	4	9	6	7	8	5	10	11	12	13

3 4 4 B A 8 7 6 Y 8 9

- **Real barcodes often come with a barcode index**
- **1-3-2-4-9-6-7-8-5-10-11-12-13 is a Barcode index for a chosen LP**
  - **it symbolizes the chosen unique LP layout**

Dataset 1 – 4 parameters to predict

- **800,000 training samples**
  - **Histogram of target parameters**



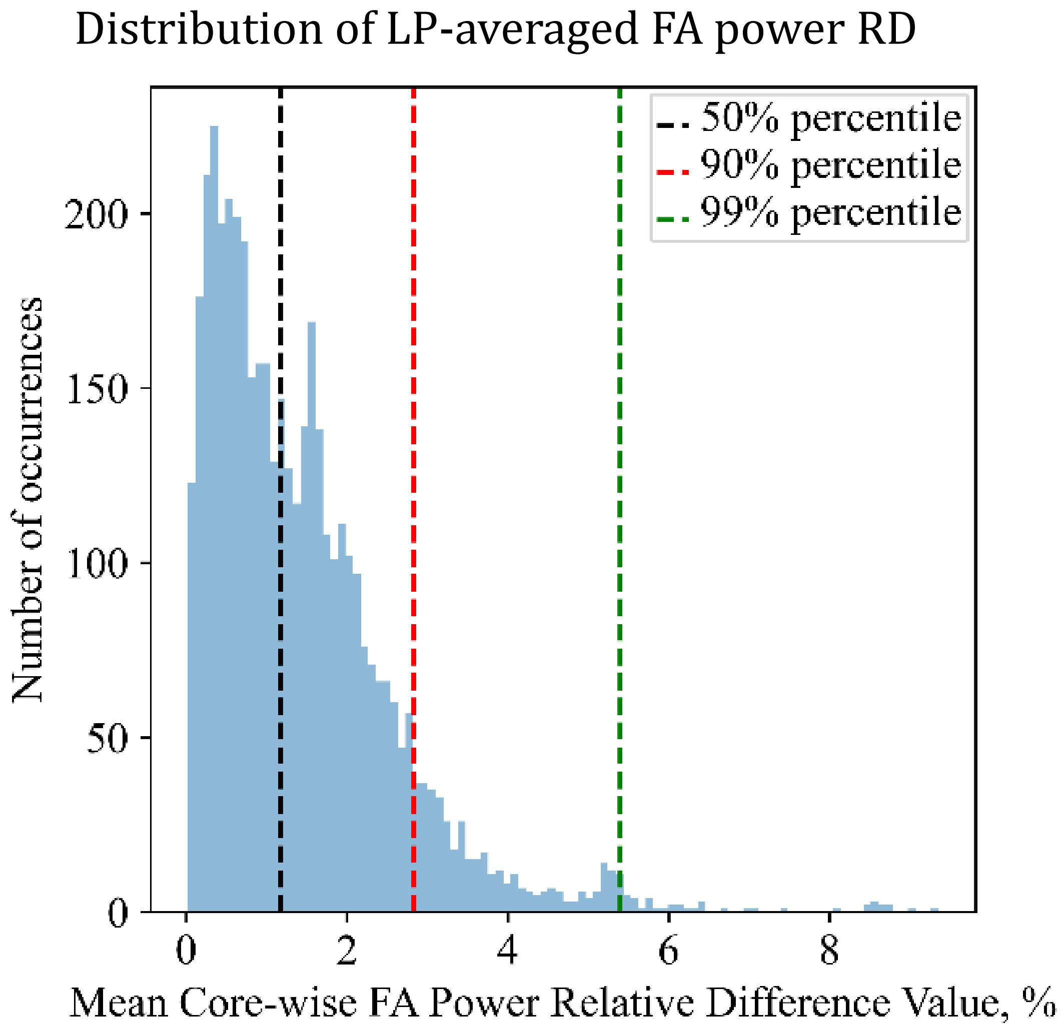
Neural Network Model

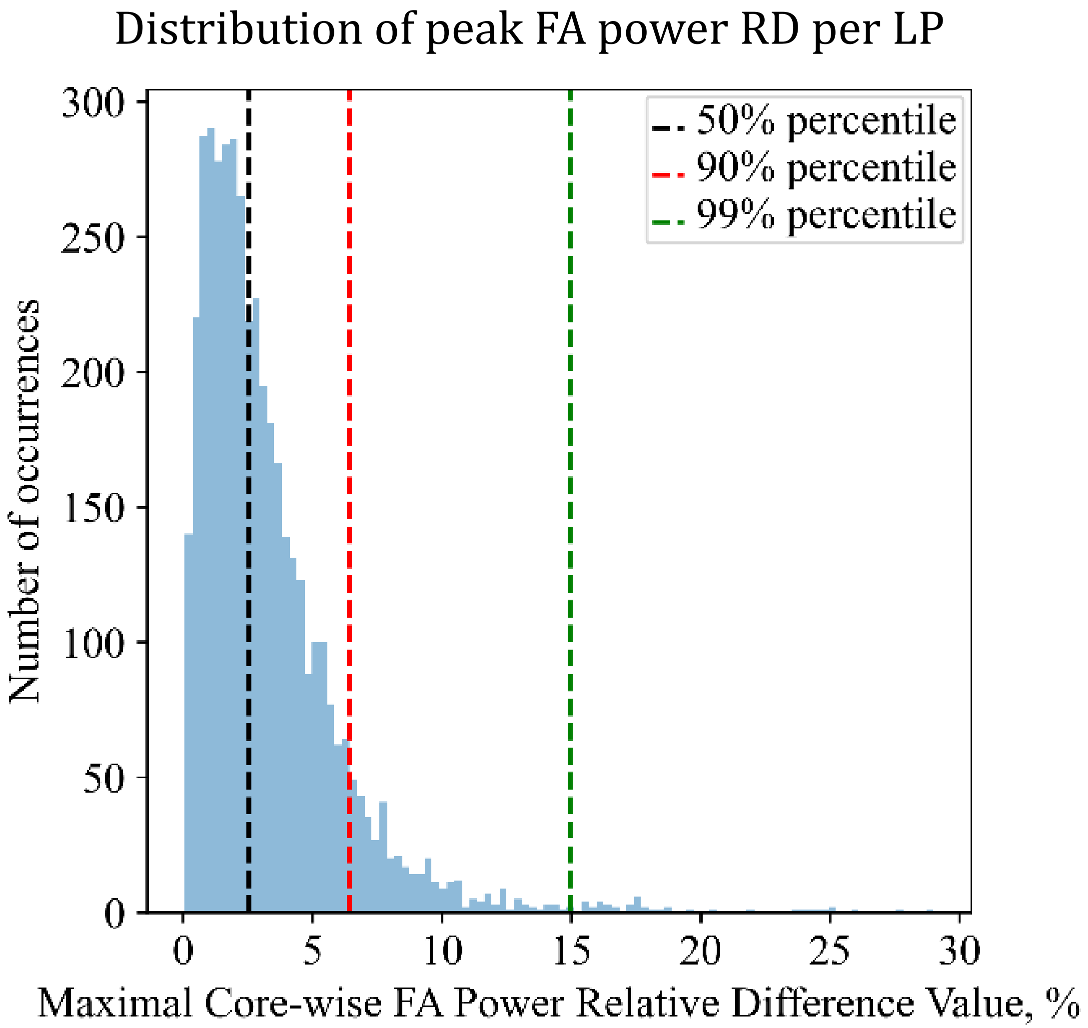
- **Stats of the CNN model:**

Parameter	**Value**	**Remarks**
Number of epochs	**2,000**	Early Stop, Save Best Model
Optimizer	**Adam**	Learning rate - **0.00002**
Loss Function	**MAE**	
Mini-batch size	**128**	-
Number of training samples	**7,931,022**	-
Number of validation samples	**252,292**	-

Core-wise stats – assembly power (axially-averaged)

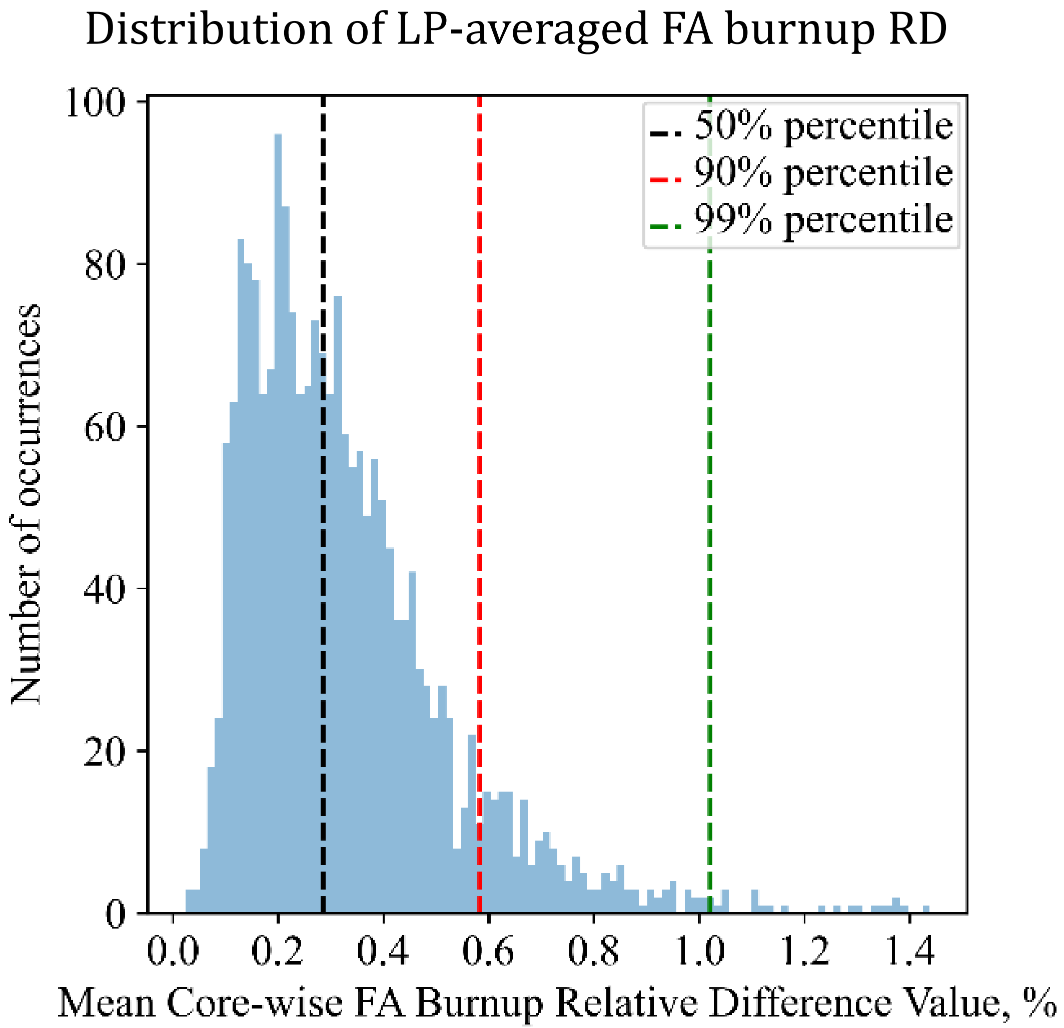
- **Statistics for FA power at BOC (0.0 GWd/MTU).**

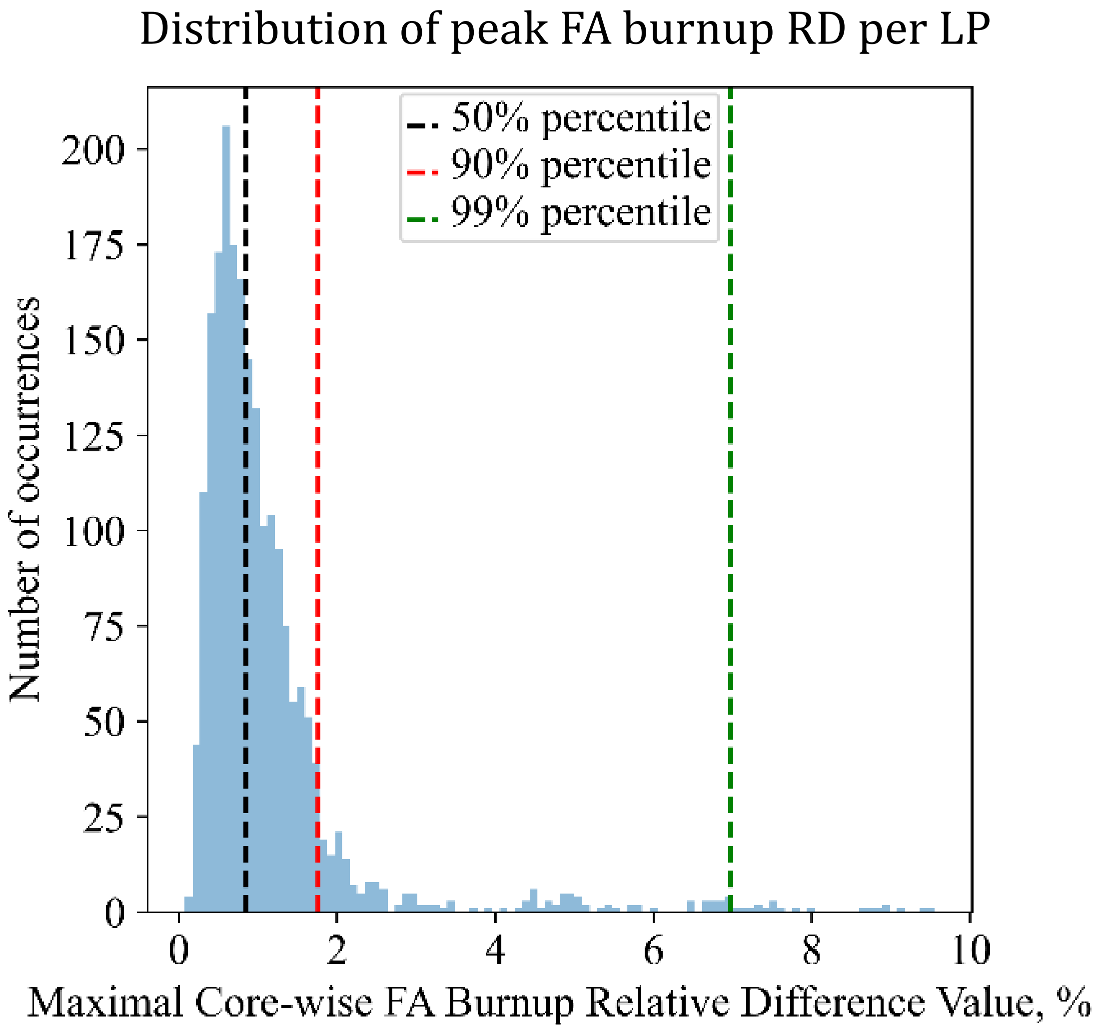




Core-wise stats – assembly burnup (axially-averaged)

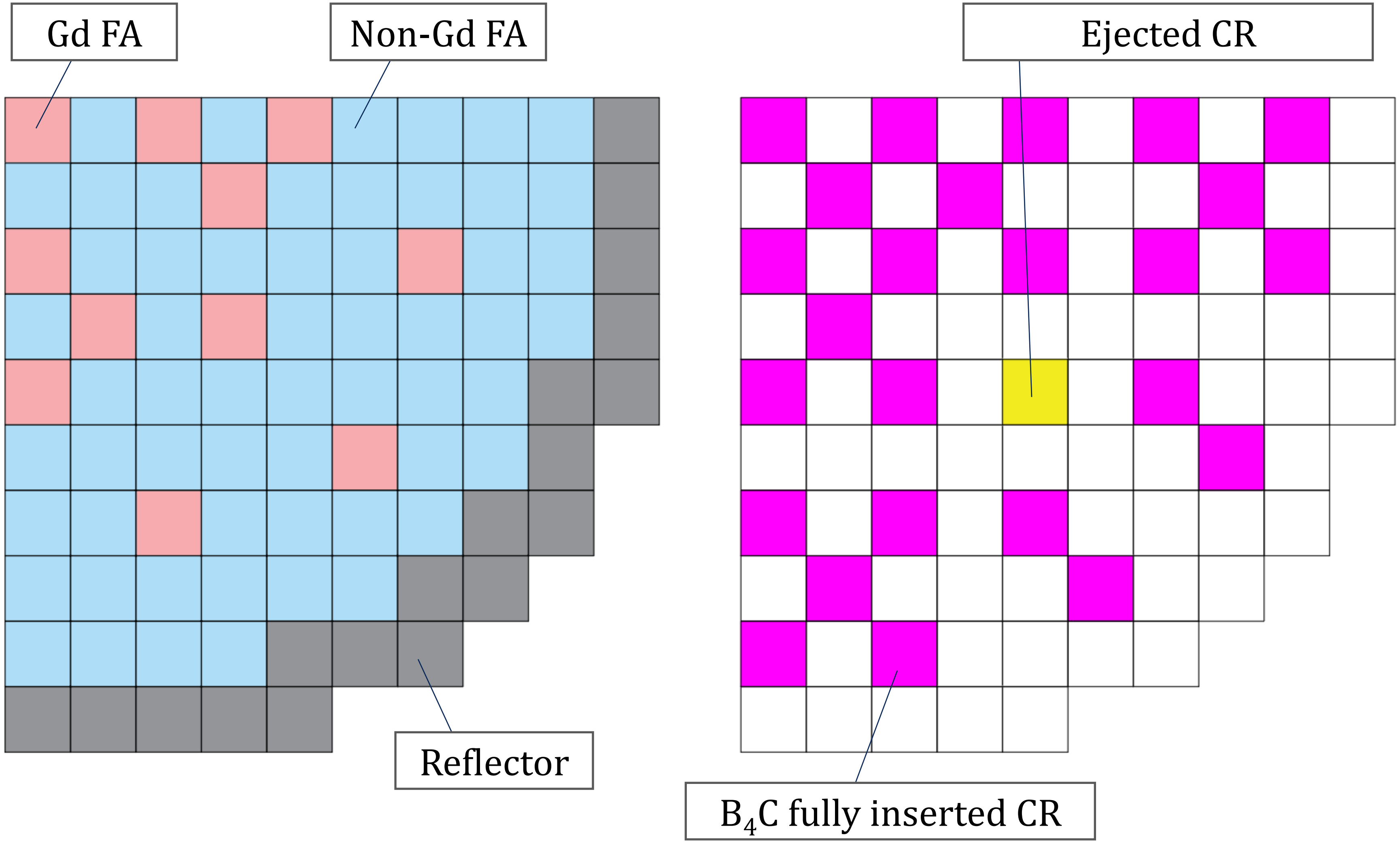
- **Statistics for FA power at EOC (16.0 GWd/MTU).**


Distribution of LP-averaged FA burnup RD


Distribution of peak FA burnup RD per LP

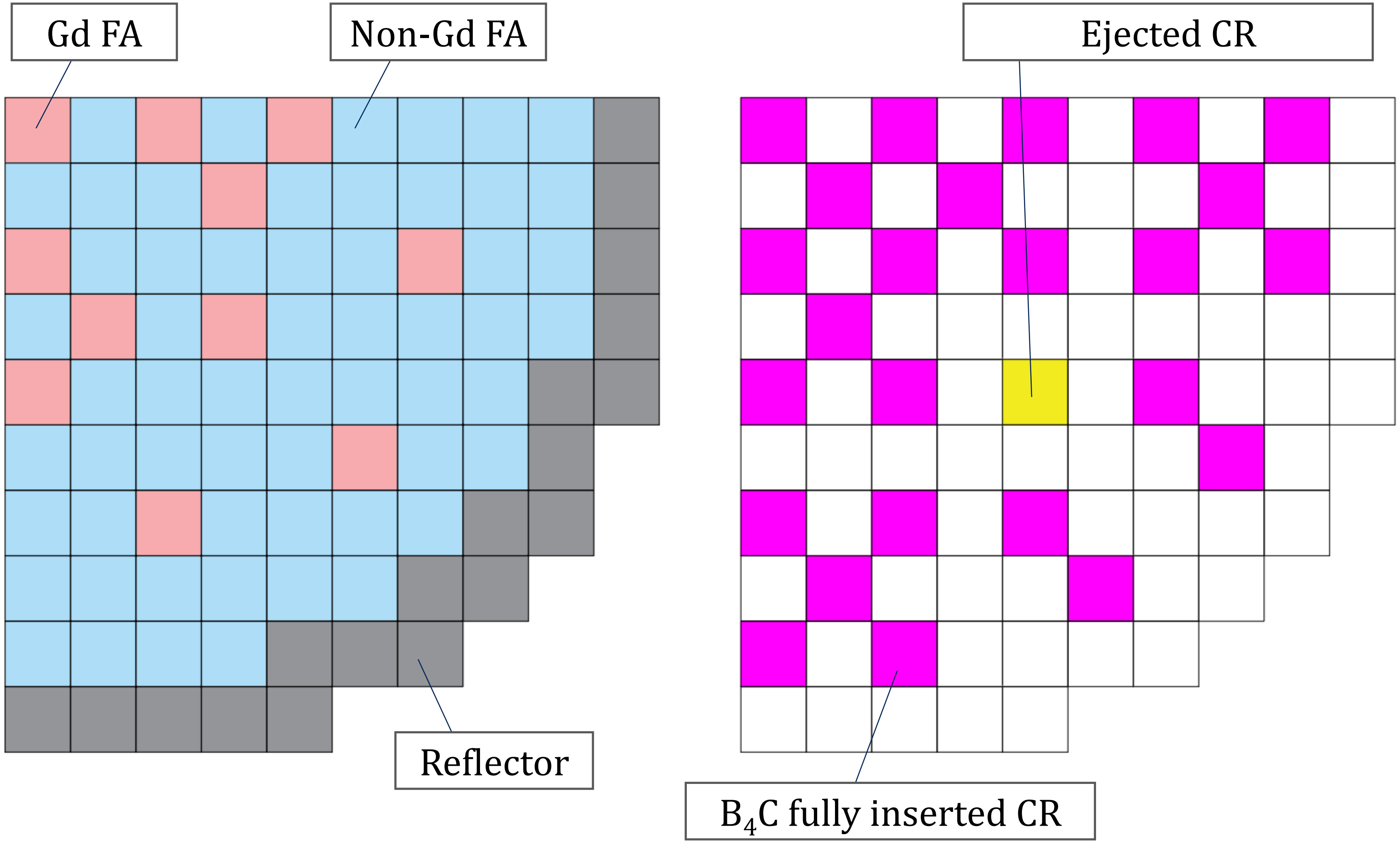
Scenario 1: HZP Rod Ejection

- All CR are inserted, and then, the “yellow” rod is rapidly ejected out of the core. The reactor power is around 0.

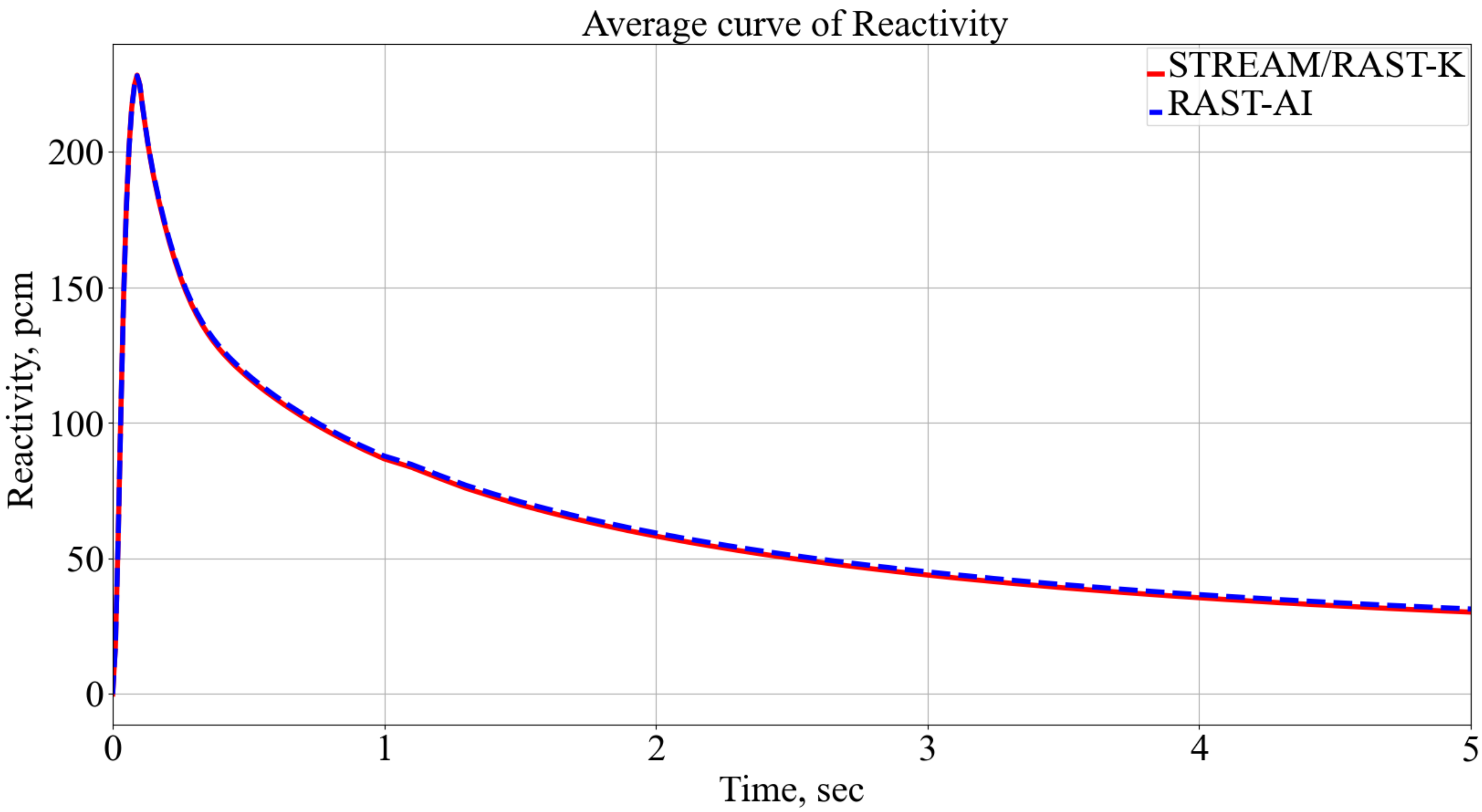


Scenario 2: HFP Rod Ejection

- **All CR are inserted, and then, the “yellow” rod is rapidly ejected out of the core. The reactor is at full power.**



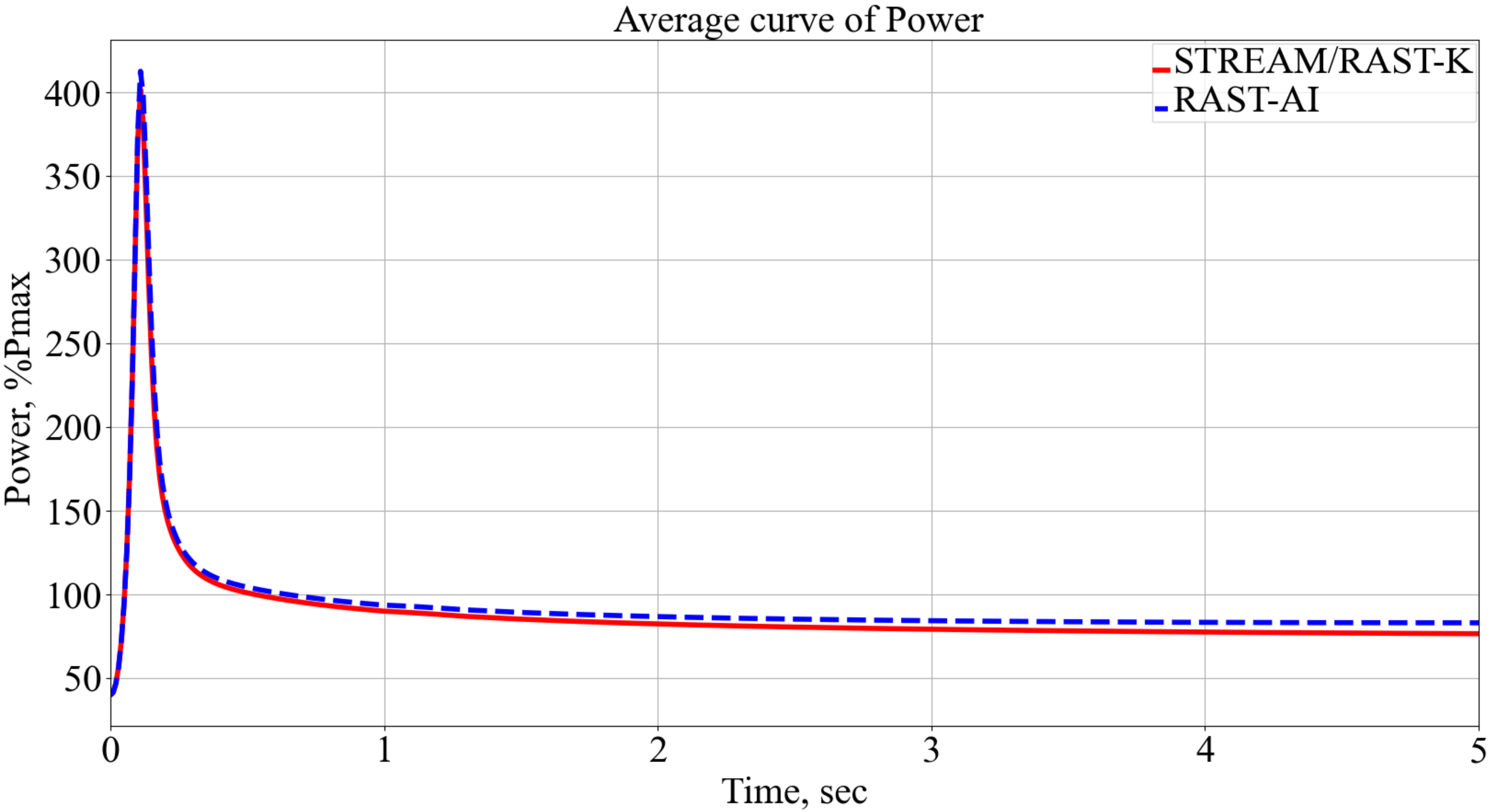
Scenario 2 – Averaged output curves



*20 full-core solutions

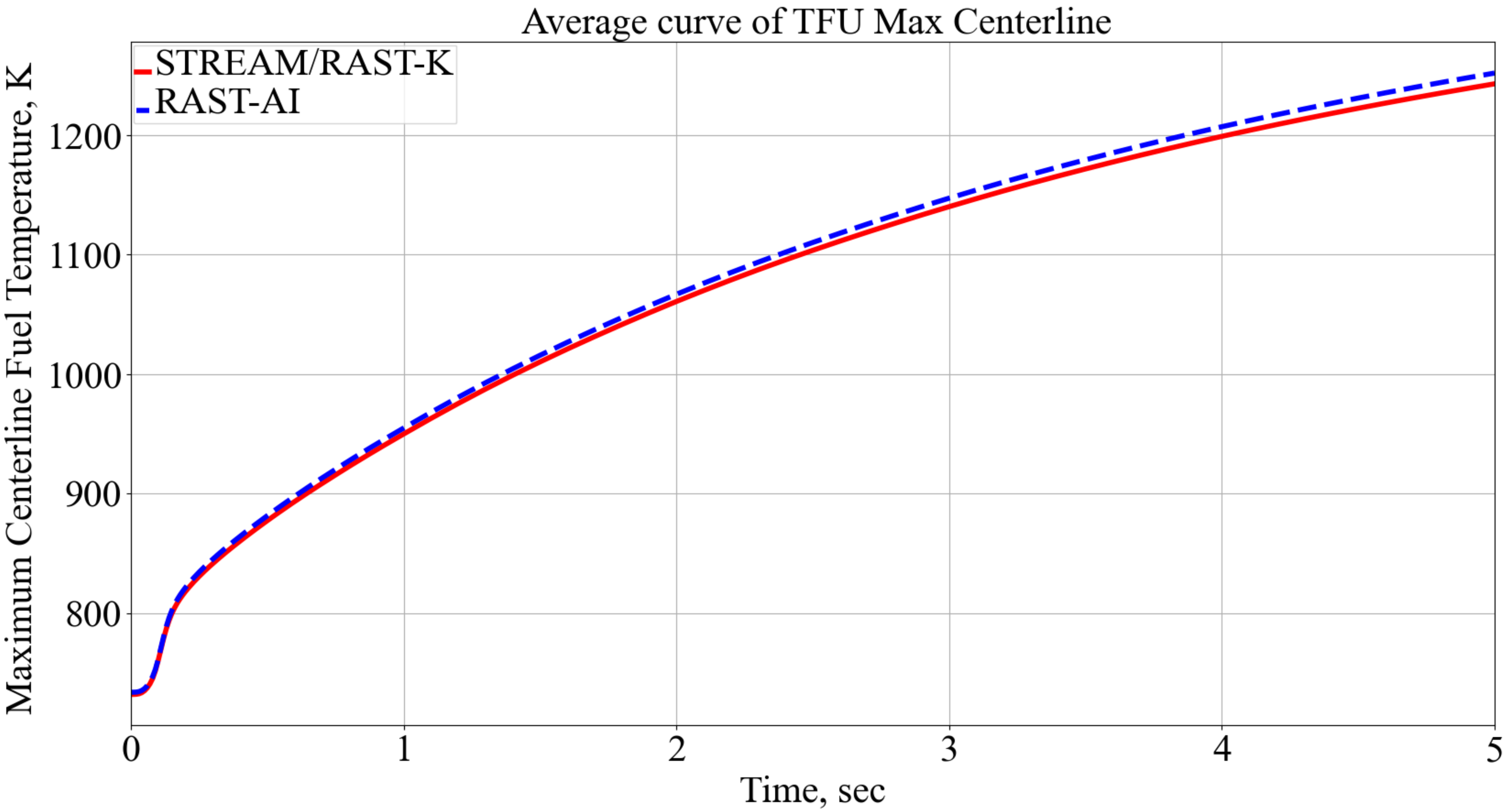
UNIST CORE

Scenario 2 – Averaged output curves



*20 full-core solutions

Scenario 2 – Averaged output curves



*20 full-core solutions

Scenario 3: HFP All Rods Insertion

- **All CR are ejected, the reactor is at full power. Then, all control rods are being inserted into the core.**