Parametric Neutronics Study and Machine Learning-Based Tritium Breeding Ratio Prediction for the ARC Tokamak

***Austin Carter**
Ethan Peterson
Alex Creely
James Logan
Matthew Vernacchia
Christoph Hasse
Katie Saltzgiver
Ryanne Kennedy

***acarter@cfs.energy**

Commonwealth Fusion Systems

Outline

- Introduction: What is ARC and how does it breed tritium?
- Objectives
- OpenMC Methodology: Building the TBR library
- Machine Learning Approach
- Results & Discussion
  - Is the surrogate model accurate for TBR?
  - Which parameters have the strongest impact on TBR?
- Conclusions
- Future Work

Introduction: What is ARC and How does ARC breed tritium?

- CFS is currently constructing SPARC [1]: a compact, high-field tokamak using high-temperature superconductors (HTS)
  - Goal: produce fusion gain Q > 2. No contribution to electrical grid
  - Major radius: 1.85 m
- ARC [2] is a similar tokamak to SPARC, but a demonstration fusion pilot power plant
  - ~ 1,000 MW of fusion power
  - ARC uses an all-liquid blanket of low pressure, slowly flowing FLiBe molten salt
  - Tritium breeding ratio $\left[\frac{N_T^{produced}}{N_T^{consumed}}\right] > 1.1$
  - Major radius: ~4.5 m

SPARC

A. J. Creely and others

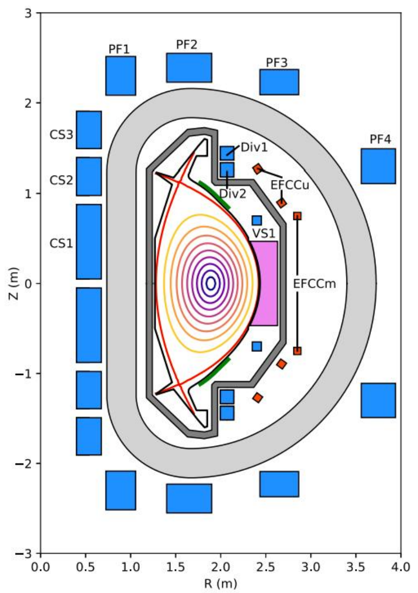


FIGURE 2. SPARC V2 poloidal cross-section. The toroidal field coil is light grey. The central solenoid and poloidal field coils are blue. Error-field correction coils are orange-red. The vacuum vessel is dark grey. The ICRH antenna is pink. The divertor and first limiting surfaces are black. Vertical stability plates are green. The plasma separatrix is red.

ARC

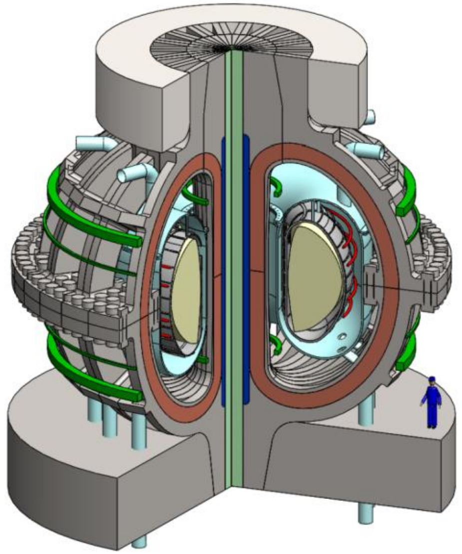


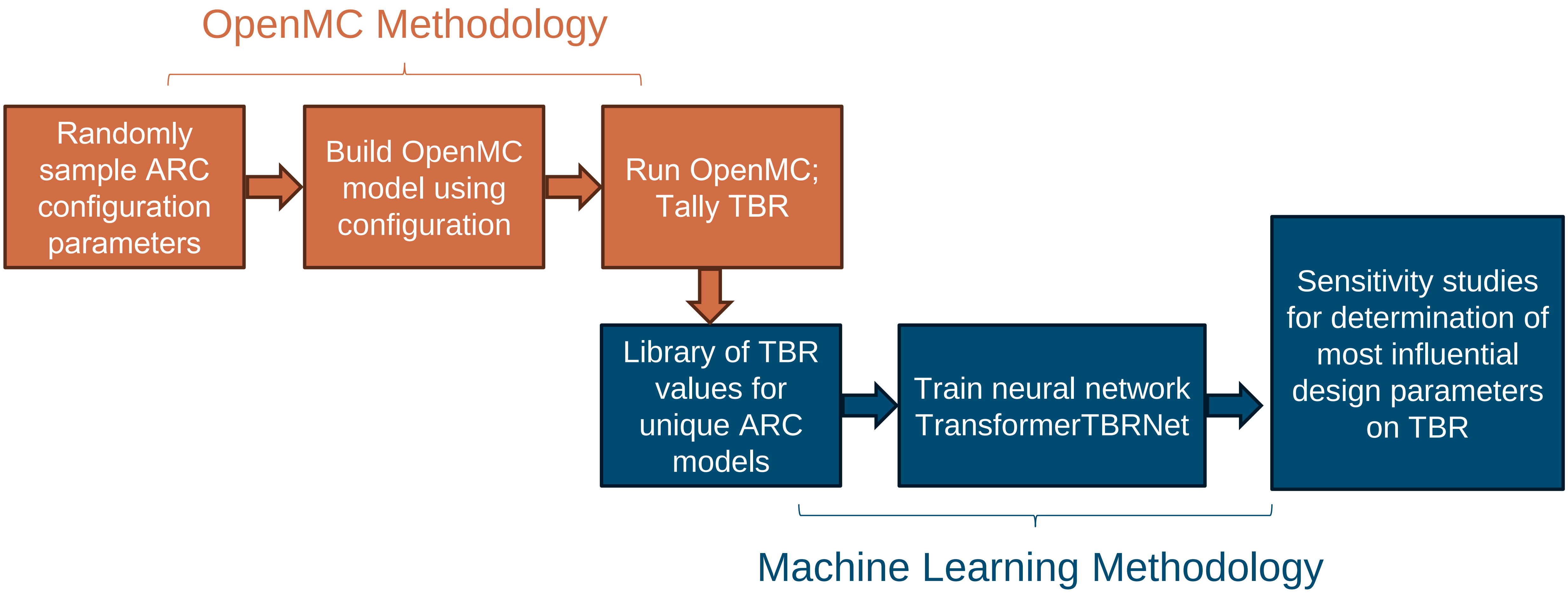
Figure 1: The ARC reactor, shown with the plasma in yellow and the TF superconducting tape in brown. Note the neutron shield is omitted for viewing clarity. Also note that although the ARC design is based on a diverted plasma, the physical divertor design was left for later study and a simplified representation of the vacuum vessel is shown here.

[1]Creely AJ, Greenwald MJ, Ballinger SB, et al. Overview of the SPARC tokamak. *Journal of Plasma Physics*. 2020;86(5):865860502. doi:10.1017/S0022377820001257
[2]Sorbom, B. N., et al. ARC: A compact, high-field, fusion nuclear science facility and demonstration power plant with demountable magnets. United States: N. p., 2015. Web. doi:10.1016/j.fusengdes.2015.07.008.

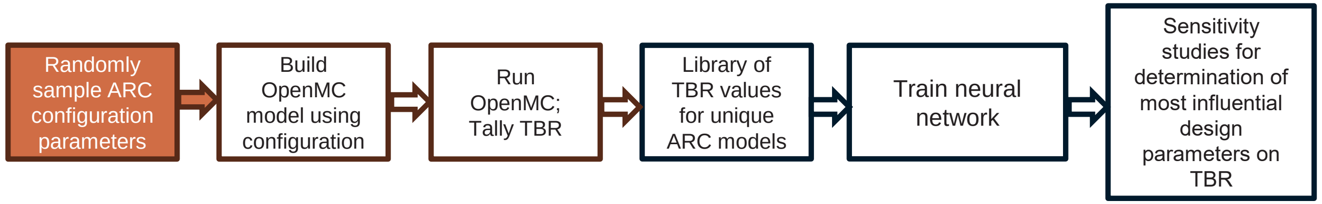
Objectives

- Assess the sensitivity of the tritium breeding ratio (TBR) to key ARC tokamak design parameters
- Maintain design flexibility during early-phase ARC development by identifying parameters with the greatest impact on TBR
- Use simplified OpenMC simulations to efficiently generate a wide range of ARC design perturbations
- Build a comprehensive TBR dataset that spans the relevant multidimensional design space
- Train a machine learning surrogate model (TransformerTBRNet) to predict TBR for arbitrary, unseen design points
- Ensure the surrogate model operates in a well-populated interpolation regime, avoiding extrapolation

Methodology



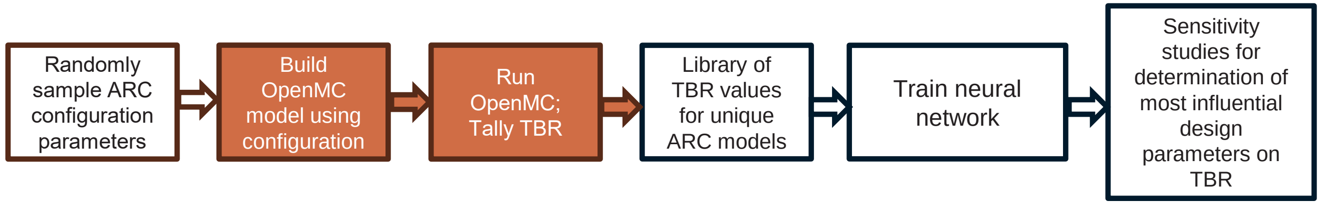
OpenMC Methodology



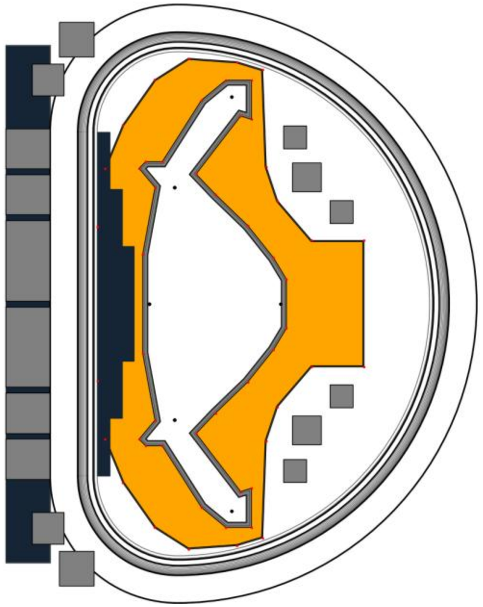
- Each ARC configuration is defined by a random draw from the sample space of design parameters shown for the definition of a unique tokamak configuration
- Sampling was performed uniformly and independently across each parameter’s allowed range
- Each configuration is passed to the automated model builder, which generates a fully resolved OpenMC geometry
- This approach enables broad coverage of the design space, supporting both sensitivity analysis and surrogate model training

Perturbed Parameter
Lithium-6 enrichment
Multiplier
Structural Material
Neutron Shield Material
First Wall Thickness
VV Inner Thickness
VVCC Thickness
VV Outer Thickness
Multiplier Thickness (if present)
Port Axial Extent
Port Toroidal Extent
Fraction of FLiBe displaced by structural material
FLiBe Impurity Fraction

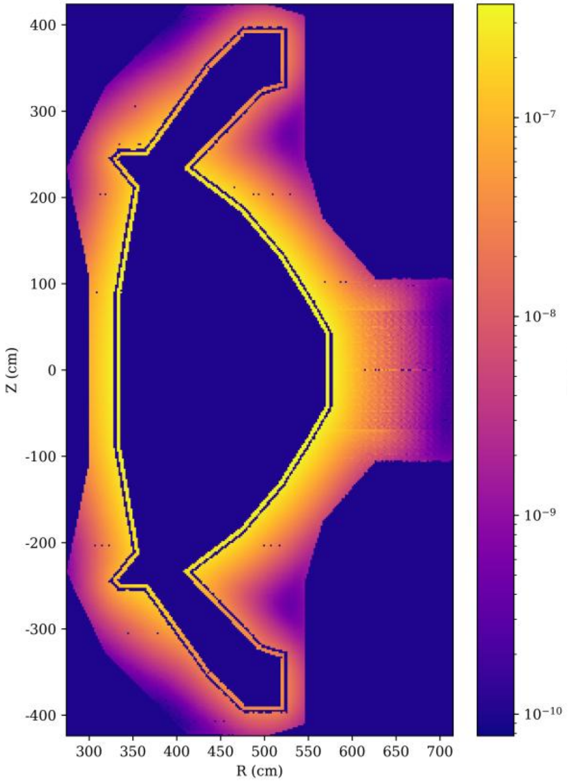
OpenMC Methodology



- Leveraged an automated ARC model builder to translate configuration files into fully defined OpenMC [1] models, with lightweight level of detail
- Geometry constructed using constructive solid geometry (CSG) with extensive use of `openmc.model.Polygon` for complex tokamak components
- Employed ENDF/B-VIII.0 [2] nuclear data for all neutron interaction cross sections
- Tallied tritium production using OpenMC's built-in "H3-production" score in regions containing FLiBe
- Each simulation run used 5 batches of 20,000 histories, balancing accuracy and speed
- Achieved an average runtime of ~0.4 CPU-minutes per simulation
- The mean 1σ standard deviation on TBR predictions was 0.00328, ensuring high-fidelity data for ML training



OpenMC rendering of an example ARC iteration. Yellow-orange indicates molten salt blanket.

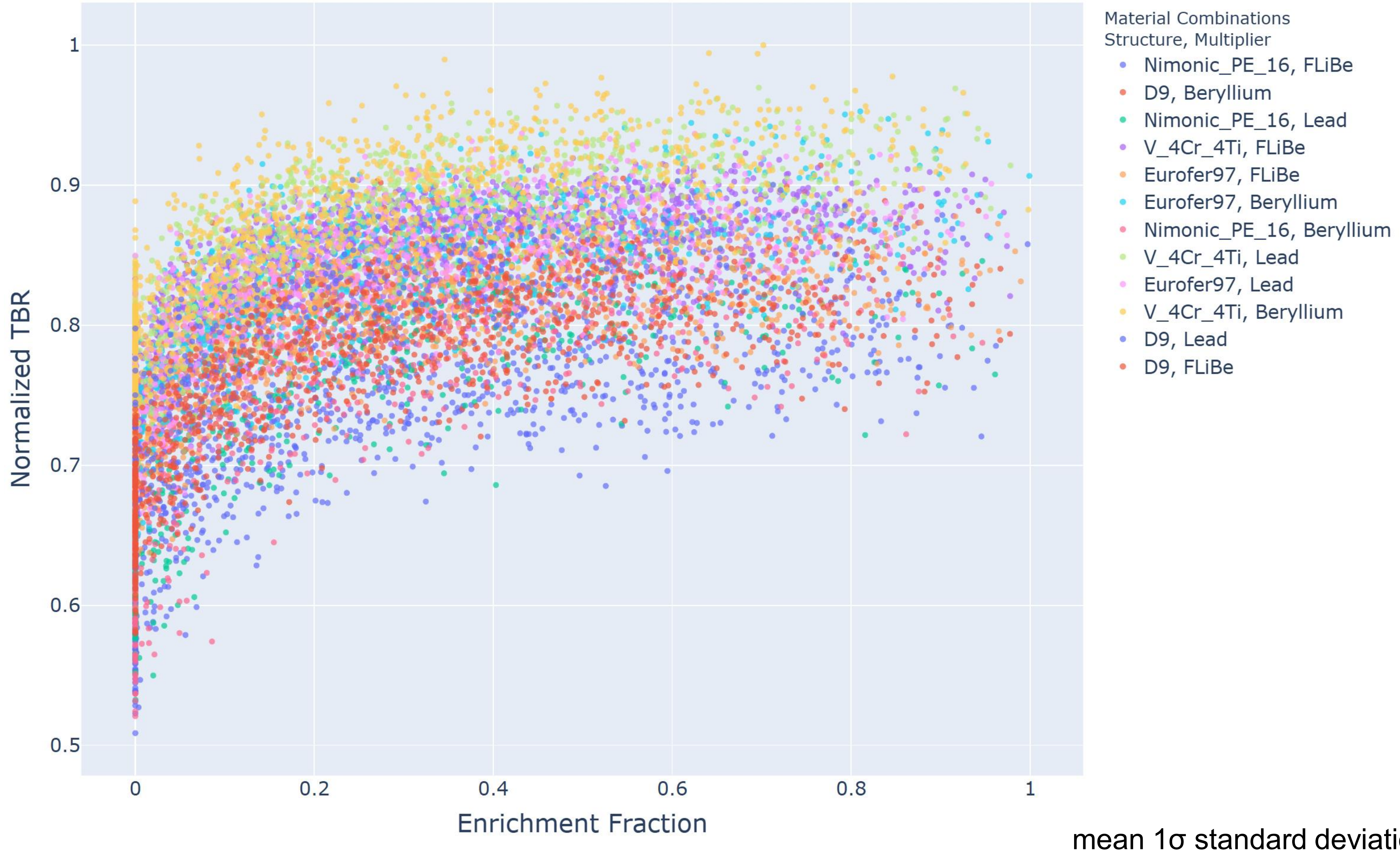


R-Z mesh tally for relative distribution of "H3-production"

[1] Paul K. Romano, Nicholas E. Horelik, Bryan R. Herman, Adam G. Nelson, Benoit Forget, and Kord Smith, "OpenMC: A State-of-the-Art Monte Carlo Code for Research and Development," *Ann. Nucl. Energy*, **82**, 90–97 (2015).
[2] D.A. Brown, M.B. Chadwick, R. Capote, et al., "ENDF/B-VIII.0: The 8th Major Release of the Nuclear Reaction Data Library with CIELO-project Cross Sections, New Standards and Thermal Scattering Data", Nuclear Data Sheets, 148: pp. 1-142 (2018).

Library of TBR values

Randomly sample ARC configuration parameters → Build OpenMC model using configuration → Run OpenMC; Tally TBR → **Library of TBR values for unique ARC models** → Train neural network → Sensitivity studies for determination of most influential design parameters on TBR



- Normalize TBR to the maximum in the library

$$Normalized\ TBR = \frac{TBR}{\max[TBR]}$$

- Plot shows the entirety of the TBR library categorized by structural material and multiplier material
- The enrichment fraction is relative to natural abundance

mean 1σ standard deviation on TBR predictions = 0.00328

TransformerTBRNet

Randomly sample ARC configuration parameters → Build OpenMC model using configuration → Run OpenMC; Tally TBR → Library of TBR values for unique ARC models → Train neural network → Sensitivity studies for determination of most influential design parameters on TBR

- The TransformerTBRNet architecture was implemented using the **PyTorch** [1] deep learning framework, enabling efficient model definition, GPU-accelerated training, and flexible experimentation with transformer-based architectures.
- Input features include both numerical parameters and one-hot encoded material selections
- Data was split into three sets:
  - 70% Training set: Used to fit model weights
  - 15% Validation set: Used to monitor generalization performance during training
  - 15% Test set: Held out entirely until final evaluation
- Splitting performed using scikit-learn's `train_test_split` with a fixed random seed (random_state=42) to ensure reproducibility
- Validation and test sets were drawn separately from the same 30% pool, ensuring the model never sees the test data during training
- All sets were normalized using the same scaler, fitted only on the training data, to prevent data leakage

Input Features

Size = 21

Parameters of ARC design

→

Linear Embedding Layer

Projects raw input vector (design parameters) into higher-dimensional space (128)

→

Add Positional Encoding

Injects positional information (even for non-sequential / tabular inputs)

→

Transformer Encoder Blocks

- 2 layers
- Multi-head Self-Attention
- Feedforward Layer (dim=256)
- LayerNorm & Dropout (0.1)

→

Average Pooling Across Sequence Dimension

Reduces (batch_size x seq_len x embed_dim) → (batch_size x embed_dim)

→

Feedforward MLP Head

- Linear (128→64)
- ReLU
- Linear (64→1)

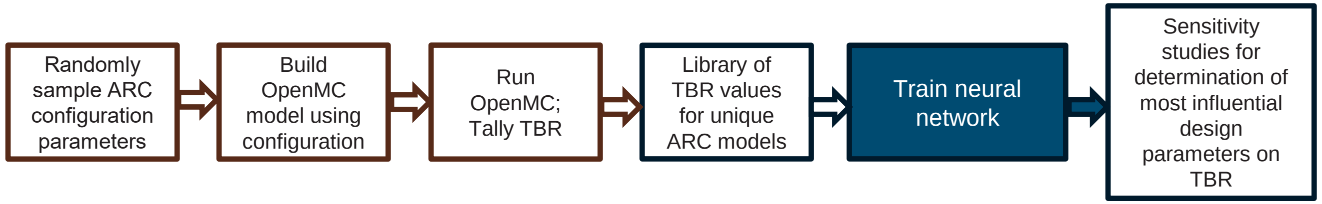
→

Output

TBR Prediction

[1] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., ... & Chintala, S. (2019). **PyTorch: An imperative style, high-performance deep learning library**. *Advances in Neural Information Processing Systems*, *32*, 8024–8035.

TransformerTBRNet



Loss Criterion: L1 Loss = Mean Absolute Error (MAE)

$$\mathcal{L}_{L1} = \frac{1}{N}\sum_{i=1}^{N} |y_i - \hat{y}_i|$$

y_i : true target TBR (OpenMC)

$\hat{y}_i$: model prediction (TransformerTBRNet)

N : Number of samples in batch (512)

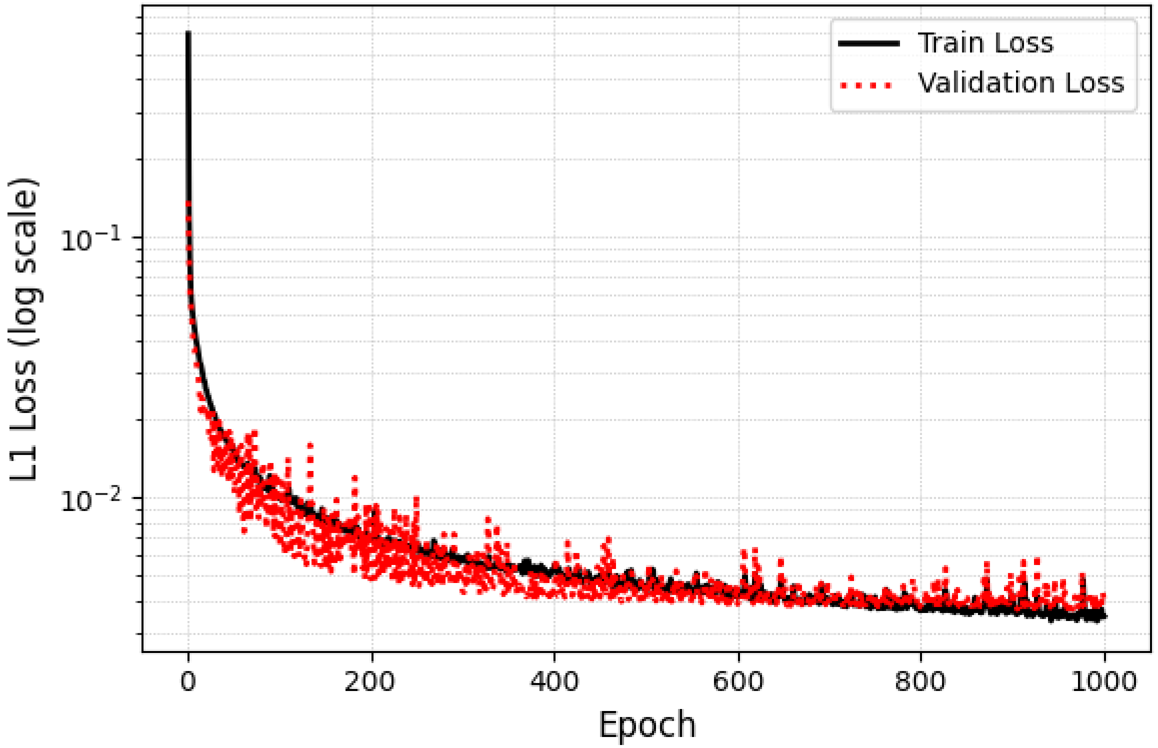
Optimizer: Adam Optimizer

$$\Theta_{t+1} = \Theta_t - \alpha \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon}$$

Θ_t : each parameter at time step t

α : learning rate = 1E-4

$\hat{m}_t, \hat{v}_t$: bias correction




```
Epoch 998/1000 - Train Loss: 0.0035, Val Loss: 0.0044
Epoch 999/1000 - Train Loss: 0.0035, Val Loss: 0.0042
Epoch 1000/1000 - Train Loss: 0.0035, Val Loss: 0.0040
```

Ultimately the surrogate model produces L1 loss comparable to OpenMC mean 1σ standard deviation on TBR predictions = 0.00328

Compute Time:

Model trained on a single NVIDIA Tesla T4 GPU

Total training time: 15 minutes (~0.25 GPU-hours)

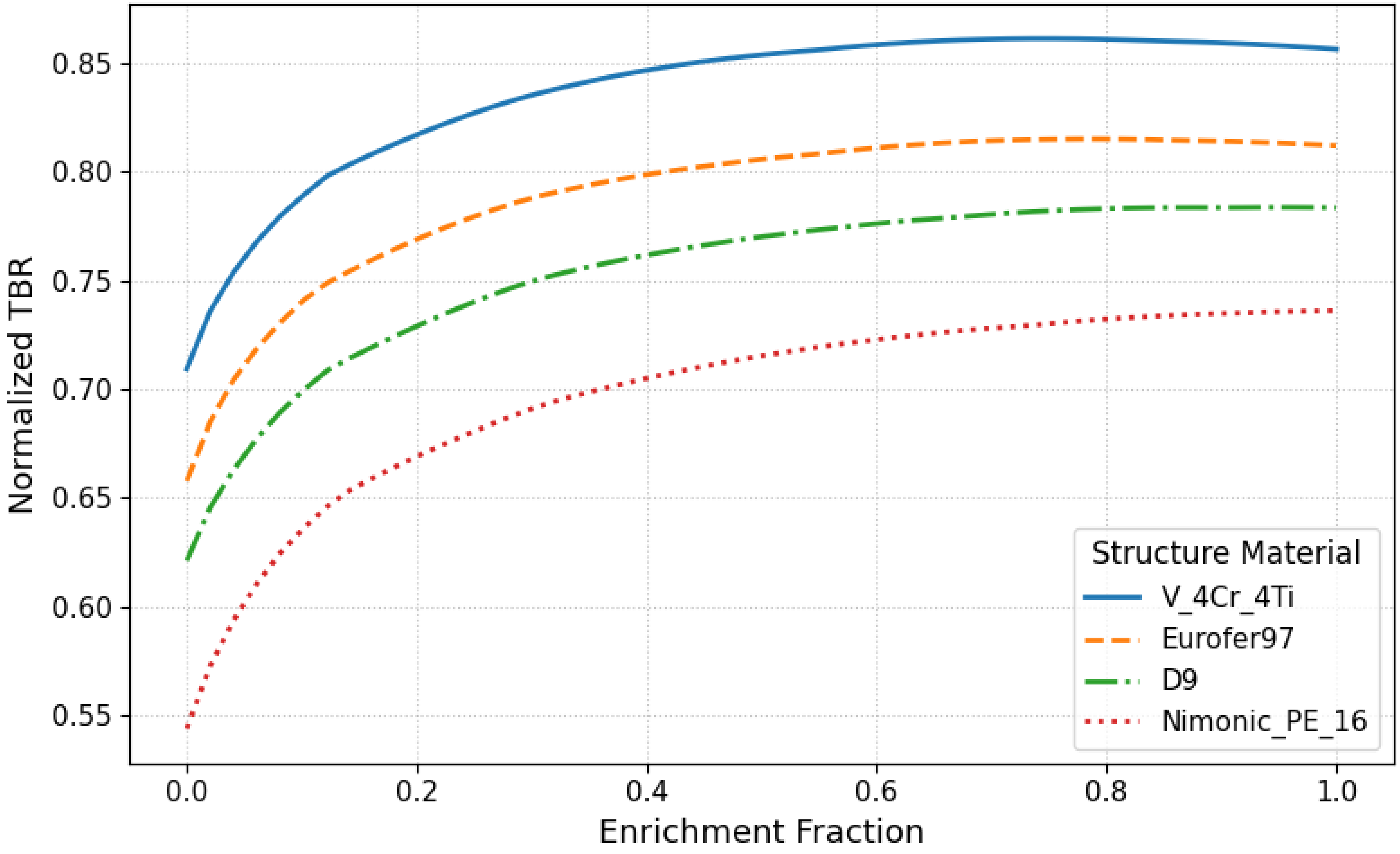
Estimated compute: ~16.25 TFLOP-hours (mixed precision)

Approx. energy consumption: ~17.5 Wh

Training completed efficiently due to model simplicity and dataset size

TransformerTBRNet Predictions & Results

Randomly sample ARC configuration parameters ⇒ Build OpenMC model using configuration ⇒ Run OpenMC; Tally TBR ⇒ Library of TBR values for unique ARC models ⇒ Train neural network ⇒ Sensitivity studies for determination of most influential design parameters on TBR



$$Normalized\ TBR = \frac{TBR}{\max[TBR]}$$

Varying two parameters that provide significant impact to the TBR

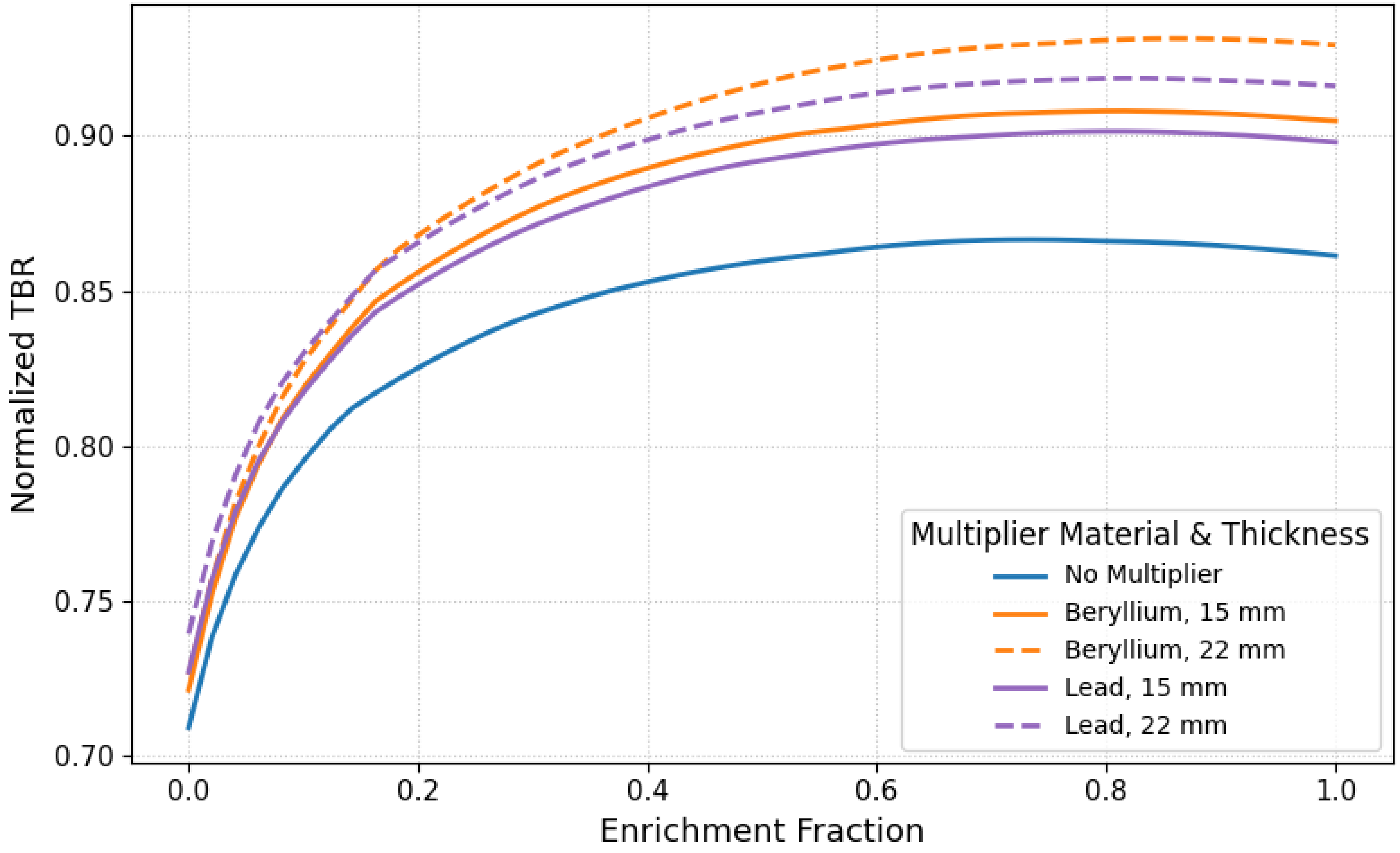
- Li-6 enrichment
- Structural material choice

Keeping other variables constant

- Shielding material = boron carbide
- No multiplier
- Nominal port size
- Nominal VV dimensions

TransformerTBRNet Predictions & Results

Randomly sample ARC configuration parameters ⇒ Build OpenMC model using configuration ⇒ Run OpenMC; Tally TBR ⇒ Library of TBR values for unique ARC models ⇒ Train neural network ⇒ Sensitivity studies for determination of most influential design parameters on TBR



$$Normalized\ TBR = \frac{TBR}{\max[TBR]}$$

Varying two parameters that provide significant impact to the TBR

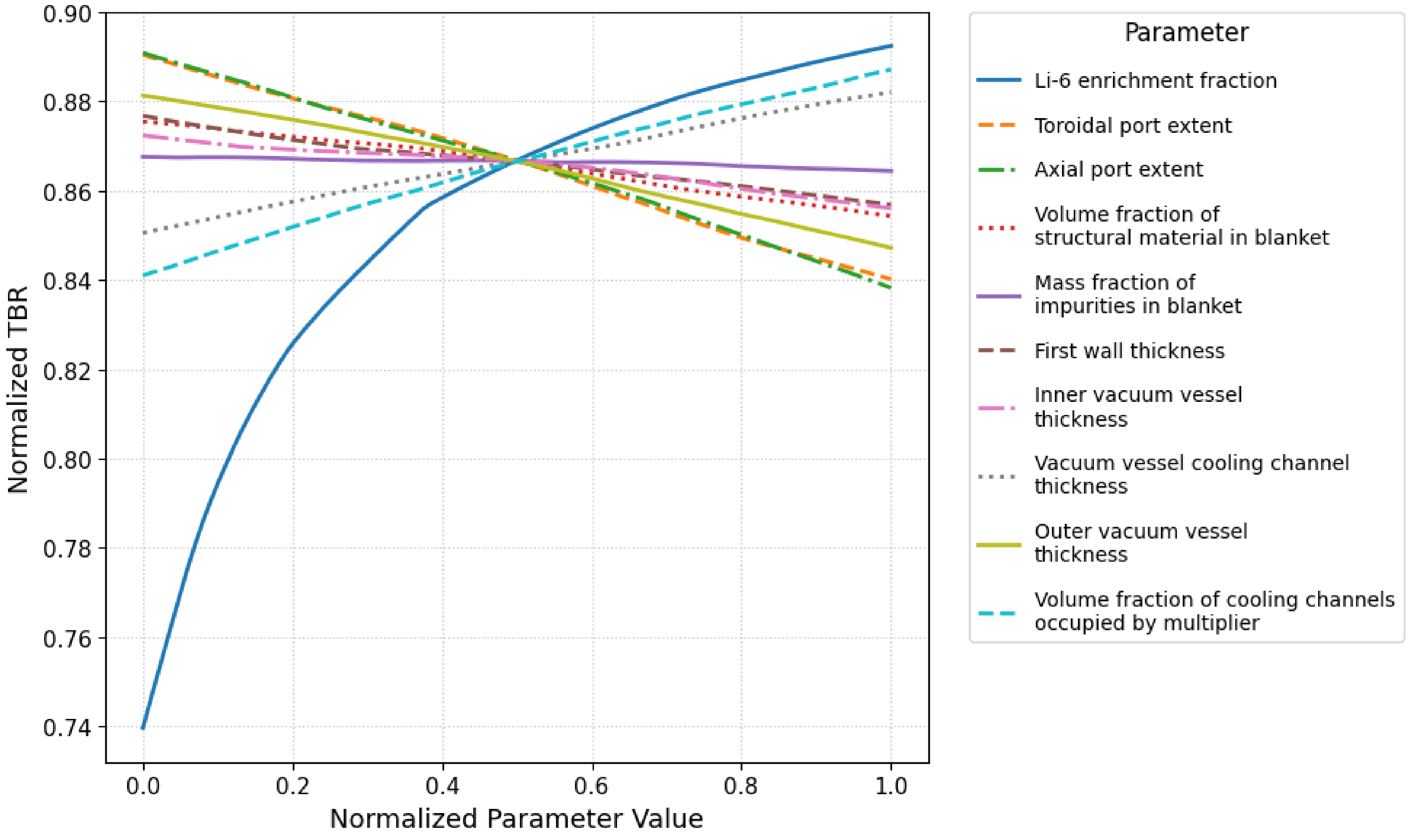
- Li-6 enrichment
- Multiplier layer embedded in the Vacuum vessel cooling channels
  - Hence the cooling channels' volume is reduced with more multiplier

Keeping other variables constant

- Shielding material = boron carbide
- Vanadium alloy structural material
- Nominal port size
- Nominal VV dimensions, slightly enlarged cooling channel volume

TransformerTBRNet Predictions & Results

Randomly sample ARC configuration parameters → Build OpenMC model using configuration → Run OpenMC; Tally TBR → Library of TBR values for unique ARC models → Train neural network → Sensitivity studies for determination of most influential design parameters on TBR



$$Normalized\ TBR = \frac{TBR}{\max[TBR]}$$

$$Normalized\ Parameter = \frac{x - \min[x]}{\max[x] - \min[x]}$$

Varying parameters individually, normalize from max to min of each

Keeping other variables constant

Use a beryllium multiplier in the cooling channel

Sensitivity coefficients on next slide

It is clear how much more impactful Li-6 enrichment is relative to other design parameters

TransformerTBRNet Predictions & Results

Randomly sample ARC configuration parameters ⇒ Build OpenMC model using configuration ⇒ Run OpenMC; Tally TBR ⇒ Library of TBR values for unique ARC models ⇒ Train neural network ⇒ Sensitivity studies for determination of most influential design parameters on TBR

Parameter sensitivity ranking (by absolute slope):	
Li-6 enrichment fraction	0.119
Axial port extent	-0.052
Toroidal port extent	-0.051
Volume fraction of cooling channels occupied by multiplier	0.046
Outer vacuum vessel thickness	-0.035
Vacuum vessel cooling channel thickness	0.031
Volume fraction of structural material in blanket	-0.022
First wall thickness	-0.018
Inner vacuum vessel thickness	-0.015
Mass fraction of impurities in blanket	-0.003

$$Sensitivity\ Factor = \frac{\partial\left[\frac{TBR}{\max[TBR]}\right]}{\partial\left[\frac{x-\min[x]}{\max[x]-\min[x]}\right]}$$

Conclusions

- ✅ Successfully generated a large, high-quality dataset of TBR values using a streamlined OpenMC modeling framework with automated ARC geometry perturbations
- ✅ Developed and trained a neural network surrogate model (TransformerTBRNet) capable of predicting TBR values across a multidimensional design space with high accuracy
- 🔍 Used the surrogate to systematically evaluate TBR sensitivity coefficients, quantifying the relative impact of various design parameters
- 🧱 Demonstrated a structured approach to assess the influence of material selections and geometric features on TBR response
- ➕ Enabled the efficient exploration of design tradeoffs by varying model parameters and observing predicted responses in TBR
- 🧪 Confirmed the capability of using automated neutronics pipelines to handle statistically meaningful variation across complex tokamak components
- 🧭 Verified that the surrogate model operates within a well-interpolated region of the design space, supporting reliable predictions without extrapolation risk

Future Work

- 🌡️ Incorporate realistic temperature profiles in the FLiBe blanket model to capture temperature-dependent neutron cross sections and fluid behavior
- 📐 Study the impact of geometry of both the vacuum vessel and blanket, including shaping and segmentation, on neutron streaming and breeding efficiency
- 🧱 Evaluate alternative blanket and shield geometries, particularly reshaping or re-optimizing the inboard neutron shield to balance tritium breeding and magnet protection
- 🧪 Include more detailed isotopic and chemical effects, including usable tritium molecular formation and transport behavior
- 🔁 Model time-dependent effects on TBR, such as burnup, irradiation damage, and material evolution over ARC lifetime
- 🧠 Improve surrogate model interpretability with tools like SHAP and uncertainty quantification
- 🚀 Use the surrogate model in automated design optimization to identify TBR-compliant ARC configurations with minimal cost or material burden

Questions?

*acarter@cfs.energy

Commonwealth Fusion Systems