Fast Emulation of the Neutron Diffusion Equation using the Reduced Basis Method (RBM)

Patrick A. Myers
Senior
Connor C. Craig, Kyle Beyer, Brian C. Kiedrowski
University of Michigan

Introduction and Motivation

Solutions to large linear systems typically involve computationally expensive numerical solvers

- Costly for many solutions at different configurations (uncertainty quantification (UQ), near-critical analysis, core optimization)
- Common physics-based perturbative methods applied require intuition and expertise to apply properly

Mission Relevance

Computational neutron transport is important to detection and monitoring systems design

- Used in modeling of SNM-containing systems in nonproliferation and treaty verification
- Optimization of next-generation energy, fuel systems, and criticality experiments
- Applicable to UQ and optimization in the validation of simulation and nuclear data

Theory

Parametric One-speed Neutron Diffusion

$$\boldsymbol{M}(\mu)|\psi(\mu)\rangle = \lambda(\mu)\boldsymbol{F}(\mu)|\psi(\mu)\rangle$$

- Where μ is a vector of parameters $(D, \Sigma_a, \Sigma_f$, or $\bar{\nu})$
- Solved through numerical methods
- This work utilizes a mesh-centered finite difference solver
- $\boldsymbol{M}$ and $\boldsymbol{F}$ are $N \times N$ and flux, $|\psi\rangle$, is N
- N is the number of mesh nodes

The Reduced Basis Method (RBM)

Reduction of order from N to R where $R \ll N$

- Calculate T training forward and adjoint fluxes with a high-fidelity solver
- Reduce training subspace with Proper Orthogonal Decomposition (POD) to R fluxes
- Define $\boldsymbol{M}^t(\mu^t)$ and $\boldsymbol{F}^t(\mu^t)$ for each target point such that $\boldsymbol{M}^t_{i,j} = \langle\psi_i|\boldsymbol{M}(\mu^t)|\psi_j\rangle$ and $\boldsymbol{F}^t_{i,j} = \langle\psi_i|\boldsymbol{F}(\mu^t)|\psi_j\rangle$
- Solve $R \times R$ eigenvalue problem for $\tilde{\boldsymbol{\lambda}} \approx \boldsymbol{\lambda}$, c
  $$\boldsymbol{M}^t(\mu^t)\boldsymbol{c} = \tilde{\lambda}\boldsymbol{F}^t(\mu^t)\boldsymbol{c}$$
- Reconstruct flux with a Galerkin Projection

Quarter Core Problem

Emulator Training and Target Calculations

Offline Stage: Training Space Construction

- 45×45 Mesh-centered finite difference (2025 mesh elements)
- Mesh grid of 4 perturbed parameters (81 training points)
- Case 1: $\Sigma_a^0 = 0.130$ and $\Sigma_a^1, \bar{\nu}\Sigma_f^1, \Sigma_a^2 = 0.10$ (Perturbation Theory (PT))
- Case 2 – 5: $\Sigma_a^0 \in \{0.125, 0.130, 0.135\}$ and $\Sigma_a^1, \bar{\nu}\Sigma_f^1, \Sigma_a^2 \in \{0.095, 0.100, 0.105\}$ (RBM with 1 – 4 POD modes)

Online Stage: Target Calculations

- 500 randomly sampled points from a 4-dimensional box in parameter space
- 494 sampled outside the training space

Mesh Size Scaling Analysis

- Finite difference and affine and nonaffine RBM with 4 POD modes
- 5 mesh sizes: 324, 1296, 2916, 5184, 8100

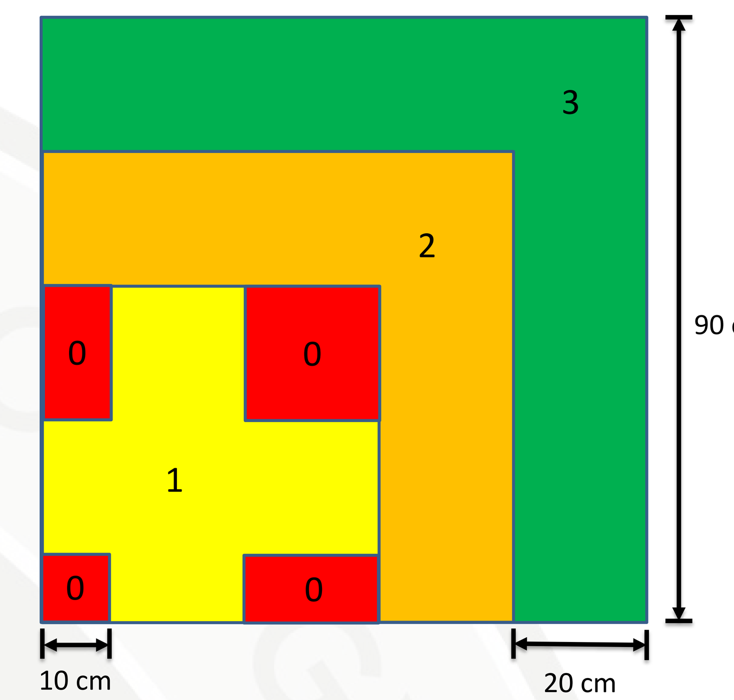

Figure 1. A reactor quadrant consisting of 4 regions with reflective boundaries at the bottom and left faces and vacuum boundaries at the top and right faces

Table 1. Material composition for each region in Figure 1

ID	Name	D (cm)	Σ_a (cm^{-1})	$\nu\Sigma_f$ (cm^{-1})
0	Fuel 1 + Rod	0.4	Perturb	Perturb
1	Fuel 1	0.5	Perturb	0.10
2	Fuel 2	0.4	Perturb	0.10
3	Reflector	0.3	0.01	~0

Conclusion

RBMs offer substantial increases in accuracy compared to perturbation theory with comparable speedup

- The target space was well approximated with two dimensions but L2 loss converged to zero with each additional POD mode
- Affine speedup remained constant with each additional POD mode, while nonaffine decreased
- Affine RBMs computationally outperformed nonaffine RBMs with an increasing mesh size
- Perturbation of Σ_a^0 and Σ_a^2 had negligible impact on k_{eff}
- Observed effect of perturbation of Σ_a^1 and Σ_f^1 on k_{eff}

Next Steps

Multi-group diffusion and empirical interpolation method for nonaffine perturbations

- Greedy training subspace construction
- C5G7 benchmark with density perturbation
- Parallelization and results generation on an HPC
- Implementation in the discrete ordinates solver in Hammer
- Exploration of applications

Results

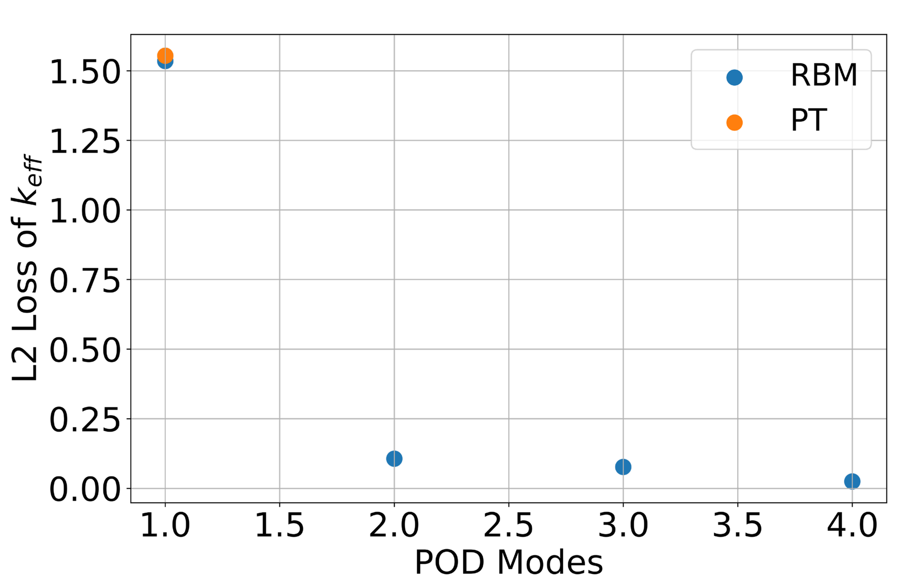

Figure 2. L2 norm of $k_{eff}^{RBM} - k_{eff}^{finite\ difference}$ versus number of POD modes in the RBM.

- Increase in accuracy with each additional POD mode
- Almost the entire target distribution can be reconstructed with 2 dimensions.

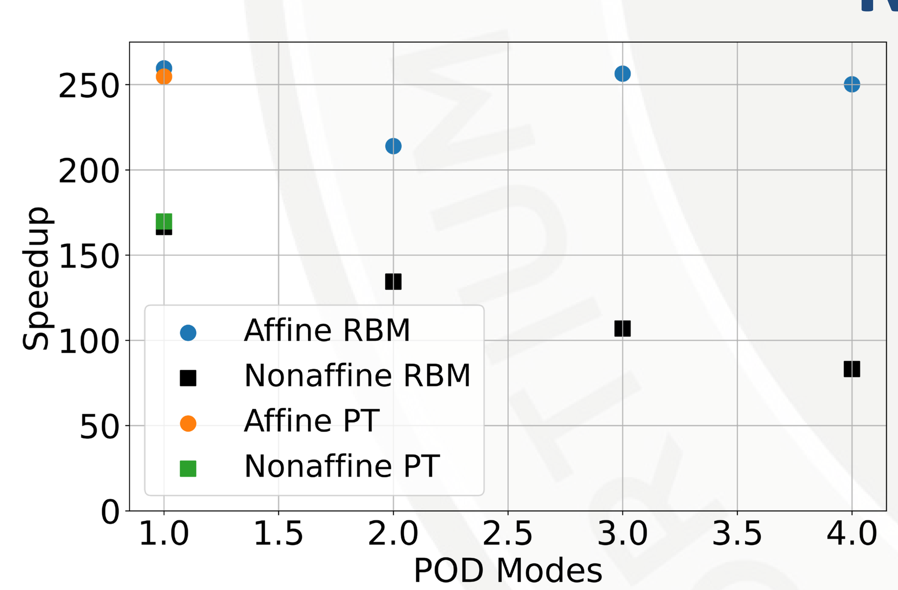

Figure 3. Speedup of affine and nonaffine RBM emulators versus emulator POD mode

- Distinct speedup decrease for each additional nonaffine emulator
- Affine emulator speedup remains constant
- Almost all emulators exceed 100x speedup over the finite difference solver.

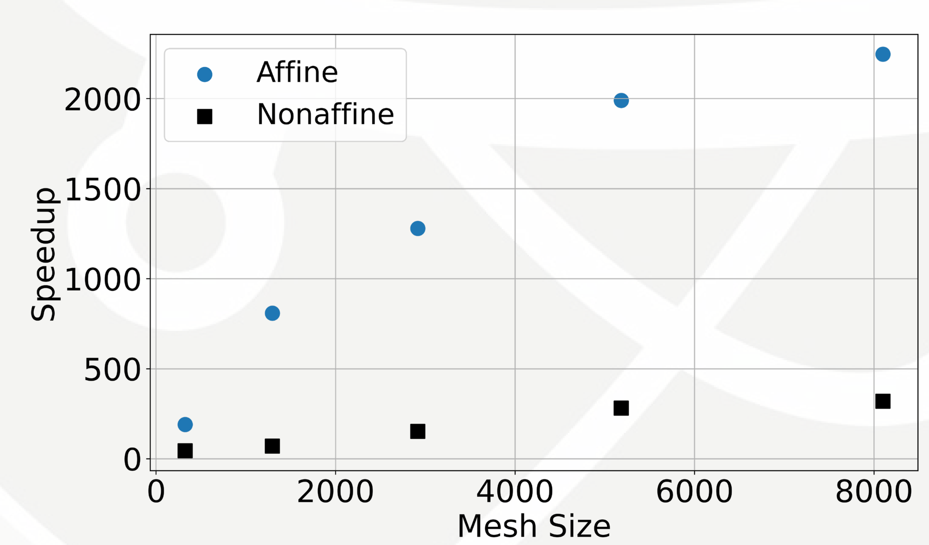

Figure 4. Speedup of affine and nonaffine RBM emulators with 4 POD modes versus mesh size

- Affine RBMs speedup scales more favorably with mesh size than nonaffine RBMs.
- Negligible impact of perturbing Σ_a^0 and Σ_a^2
- Positive relationship between Σ_f^1 and k_{eff}
- Negative relationship between Σ_a^1 and k_{eff}

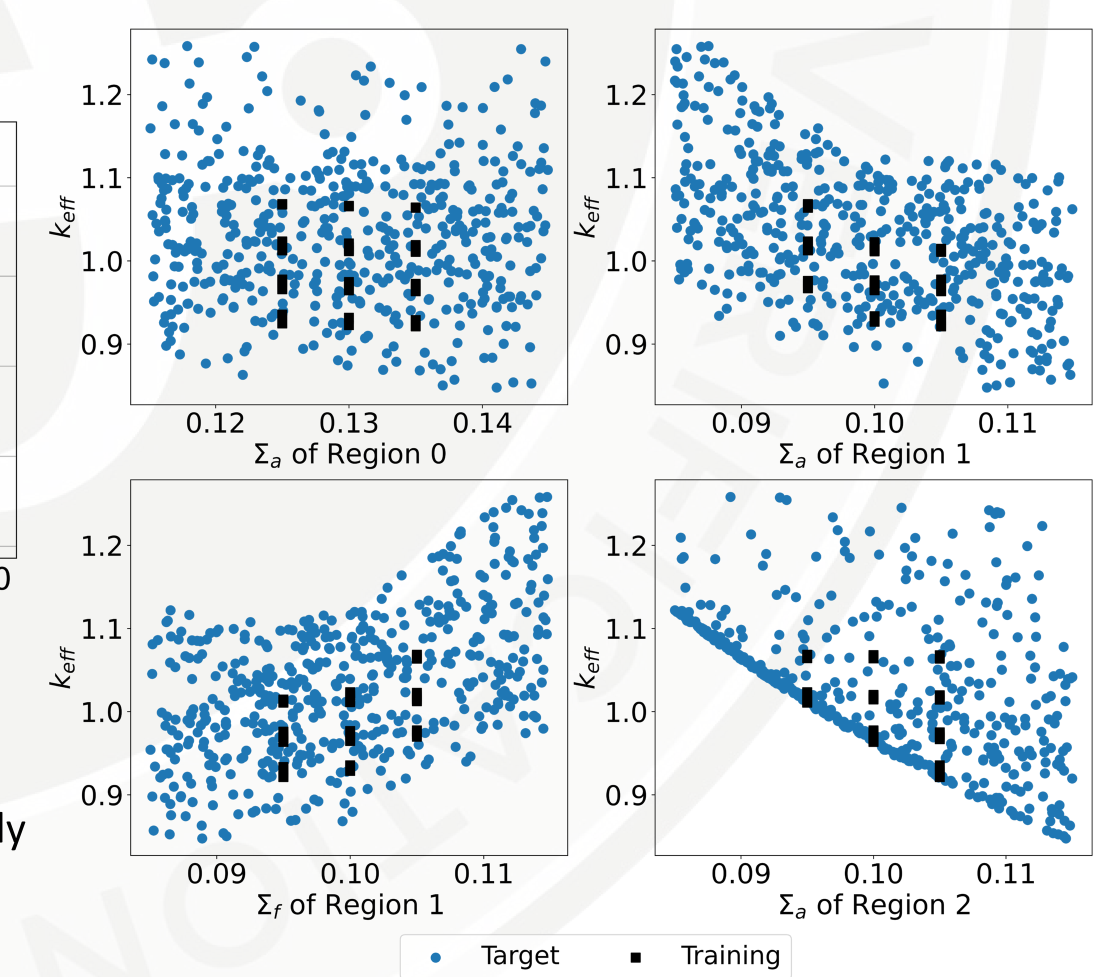

Figure 5. Effect of each parameter on k_{eff}

Expected Impact

Rapid solution of k-eigenvalue problems for multi-configuration applications

- Applicable to UQ, near critical analysis, optimization, etc.
- Reduction in computational time and cost with reliable approximations
- Reduction in the number of high-fidelity calculations required

MTV Impact

Developed passion for computational physics through research experiences

- Learned the basics of software development
- Explored computational methods for neutron transport
- Improved writing and public speaking skills
- Advisors provided critical expertise and support

Contact Email: myerspat@umich.edu

This work was partially funded by the Consortium for Monitoring, Technology, and Verification under DOE/NNSA award number DE-NA0003920

National Nuclear Security Administration