

K-MEANS CLUSTERING OF NEUTRON SPECTRA FOR CROSS SECTION COLLAPSE

Steven Douglass, Nathan Gibson

Naval Nuclear Laboratory

P.O. Box 1072, Schenectady, NY 12301, USA

steven.douglass@unnpp.gov, nathan.gibson@unnpp.gov

ABSTRACT

The process of generating cross sections for whole-core analysis typically involves collapsing cross sections against an approximate spectrum generated by solving problems with reduced scope (e.g., 2D slices of a fuel assembly). Such spectra vary with the location of a material region and with other state parameters (e.g., burnup, temperature, soluble boron concentration), resulting in a burdensome and potentially time consuming process to store and load spectra. Commonly, this is resolved by manually determining material regions for which the cross sections can be collapsed with a single weighting flux, requiring a combination of domain knowledge, engineering judgment, and trial and error. Exploring new reactor concepts and solving increasingly complicated problems with deterministic transport methods will therefore benefit greatly from an automated approach to grouping spectra independent of problem geometry or reactor type. This paper leverages a data analytics technique known as k -means clustering to group regions with similar weighting spectra into individual clusters, within each of which an average weighting flux is applied. Despite the clustering algorithm being agnostic to the physics of the problem, the approach results in a nearly 98% decrease in number of spectra regions with minimal impact to the accuracy.

KEYWORDS: Multigroup Cross Sections, Cross Section Collapse, Bengal, Jaguar

1. INTRODUCTION

Deterministic transport methods require significant computational expense to convert pointwise nuclear cross section data into cross sections appropriate for solving the multigroup transport equation. As these types of transport methods are applied to increasingly large scale reactor physics problems, the challenge of managing the cross section data becomes significant. The typical process used for generating such data involves successive transport calculations of increasing complexity in space/angle with decreasing complexity in energy. This typically involves pin-cell or infinite medium calculations with high energy fidelity (pointwise or ultrafine multigroup), assembly-level calculations with medium energy fidelity (fine-group), and core-level calculations with low energy fidelity (coarse-group). Even in the case of whole-core transport without spatial homogenization, such considerations are necessary to reduce the energy complexity, as problems on the scale of full (or even quarter-core) nuclear reactors are large enough that using a high-fidelity energy group structure is impractical.

This paper focuses on one particular element of this process for reactor physics calculations: spectra generation and cross section collapse from the fine-group level to the coarse-group level. Reducing the cross sections from the dozens or hundreds of groups necessary for accurate resonance self-shielding calculations to the ten or fewer groups for which the core problem is tractable broadly takes the form of

$$\Sigma_{Gr} = \frac{\sum_{g \in G} \sum_i N_r^i \sigma_{gr}^i \phi_{gr}}{\sum_{g \in G} \phi_{gr}}, \quad (1)$$

where Σ_{Gr} is the macroscopic cross section in fine-group G and material region r , N_{ir} is the number density of isotope i in region r , σ_{gr}^i is the resonance self-shielded microscopic fine-group cross section for isotope i in region r , and ϕ_{gr} is the scalar flux in fine-group g and region r . Because ϕ_{gr} is the primary unknown of the core physics problem, it is not available *a priori* to collapse the cross sections.

The common resolution to this conundrum is to evaluate Eq. (1) with an approximate value of ϕ_{gr} , commonly referred to as the “weighting spectrum” or “collapsing spectrum”. To approximate the spatial dependence of ϕ_{gr} , a smaller representative problem is solved with fine-group cross sections, the solution of which provides collapsing spectra for each material region. The fidelity of the resulting coarse-group transport problem depends on how well the smaller problem approximates the solution that would be obtained if the full problem were solved with the fine-group structure. The accuracy could therefore generally be improved by increasing the size of the smaller problem to as large as can be reasonably solved by the available computational resources; however, as the size of the problem increases, the memory requirements of storing and loading the spectra for later collapsing calculations increases as well.

The repeated lattice arrangement inherent in most reactor designs suggests an approach for reducing this memory burden. As an example, consider the flux spectra for a single fuel pin. The flux spectra within the pin depends on the material composition and the local environment (e.g. proximity to gadolinia-loaded pins or guide tubes). Because reactors typically consist of a repeated lattice, there are likely many other pins with similar composition and local environment, leading to substantively similar flux solutions. The memory footprint of the fine-group solution can be reduced without impacting the downstream calculations by saving only one average spectra for all such similar pins instead of individual spectra for each. This naturally extends beyond fuel regions; most regions likely have spectra similar to that of other regions. Unfortunately, the size and complexity of reactor problems makes manually determining which regions should be combined into common “spectra groups” a thoroughly time-consuming process, made moreso because the appropriate grouping could change if the state parameters (e.g., burnup, temperature, soluble boron concentration) change and because the number of spectra groups necessary to avoid significant impact to the downstream coarse-group solution is unknown.

2. THEORY

It is towards the goal of automatically determining appropriate spectra groups that we turn our attention to data clustering algorithms. The most straightforward clustering algorithm is known as k -means clustering [1]. In this algorithm, a large dataset of “observations” are considered, which are arbitrarily long vectors, and the user specifies the number of clusters k into which the data

should be categorized. The goal of the algorithm is to determine k clusters of observations which minimize the intra-cluster distance. The algorithm is summarized as follows:

1. Start with the observations and randomly guess the location of the centers of k clusters.
2. Categorize each observation into the cluster whose center it is closest to.
3. Calculate the mean value of all observations in each cluster and update the cluster center to that value.
4. Recategorize each observation according to the new cluster centers.
5. Repeat until the cluster centers are stationary between iterations within a specified tolerance.

At completion of the algorithm, each observation will be mapped to a specific cluster and will be closer to that cluster's center than any other. In this paper, distances between observations and the cluster centers are given by the L_2 norm, though this not a limitation of the k -means algorithm. Exploring additional norms that better consider the application might have benefit but is beyond the scope of this paper.

The k -means algorithm is attractive because it is simple, fast, and already implemented in several off-the-shelf libraries such as SciPy [2]. However, the resulting clusters are sensitive to the initial conditions, and different executions may result in different clusters depending on the initial (guessed) cluster centers. This is exacerbated when the dataset does not possess obvious clusters (or not enough obvious clusters for the value of k specified). In order to reduce the impact of this sensitivity, the k -means algorithm is often run several times with different randomly generated initial guesses and the realization with the best clustering is returned. In this context, the best clustering is defined as the realization with the lowest average intra-cluster distance (the average distance between each observation in a cluster to the cluster's center) over all the clusters. In order to apply this algorithm to the problem of spectra reduction, we define each "observation" as the spectra in a single region. The k -means algorithm can then be applied to these spectra observations to group similar spectra regions together into a specified number of clusters. Each region is then assigned the average spectra of the cluster into which its spectrum was categorized.

3. RESULTS

3.1. Test Problem and Code System

The test problem used to gauge the accuracy of the k -means algorithm for spectra reduction is a 2D planar slice of the quarter-core Watts Bar Hot Zero Power benchmark problem from the VERA benchmark suite [3]. Fig. 1 presents the relative pin fission density (PFD) distribution and solution parameters for this problem. All calculations in this paper are performed using the Naval Nuclear Laboratory's Advanced Deterministic Transport (ADT) system, consisting of the Jaguar transport solver [4], Bengal multigroup cross section processing package, and the Lynx system for code execution.

3.2. Accuracy of Clustered Results

To explore the effectiveness of the k -means algorithm in spectra reduction, the k -means algorithm implemented in Scipy (specifically `scipy.cluster.vq.kmeans` and

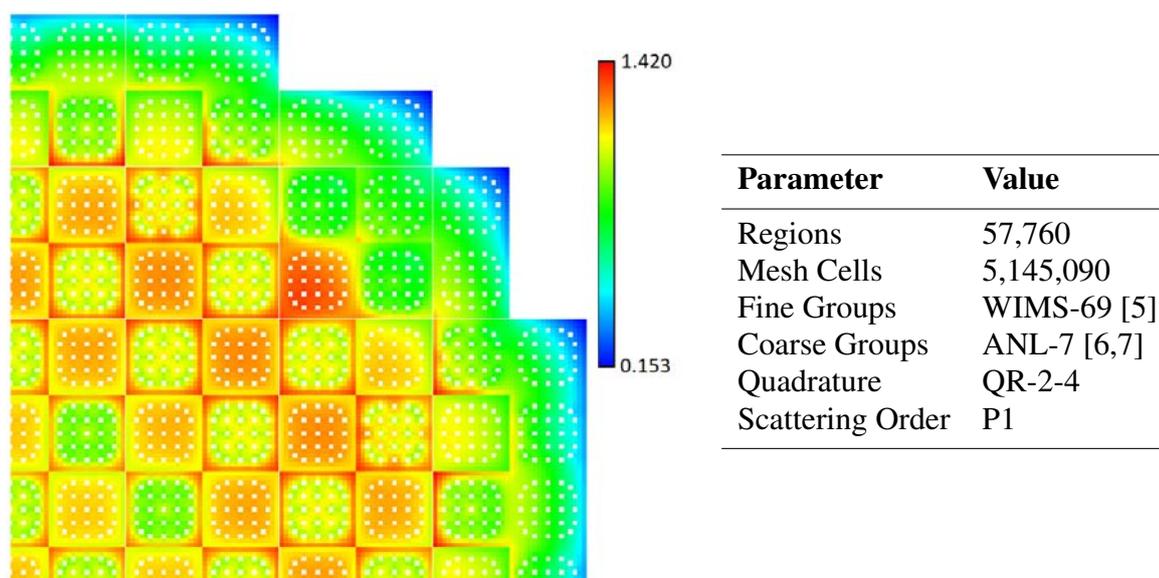


Figure 1: Relative pin fission density distribution and parameters for the test problem.

`scipy.cluster.vq.vq`) was used to cluster the fine-group flux from the test problem described in the last section. Cases were run from 50 to 5000 clusters using 30 realizations of the algorithm to reduce the impact of initial conditions on the result. In Fig. 2, the regions mapped to the same spectra group are presented for the 50- and 1,000-cluster cases. Despite the lack of any physics embedded in the algorithm, the 50-cluster layout looks consistent with what one would expect intuitively: within the active core, similar locations within assemblies possessing similar enrichment and poison configuration tend to be clustered, and at the core periphery the reflector effect creates rings of similar spectra. Fig. 2 also demonstrates that while using a small number of clusters results in identifiable features, increasing the number of clusters reduces the clarity of any physical interpretation.

Using the average spectra within each of the clusters as demonstrated in Fig. 2 to collapse the cross sections of every region in that cluster, coarse group calculations were performed, and the accuracy is examined through both eigenvalue in Table I and PFD in Fig. 3. In these comparisons, the reference solution is a coarse-group calculation wherein each material is collapsed using its own spectrum from the fine-group calculation. This is done to isolate the additional error of spectra clustering from the inherent error of collapsing the cross sections in general. Examining the results, it is clear that 50 clusters is insufficient to fully capture the physics, resulting in an eigenvalue error of approximately 50 pcm and error in the relative PFD of nearly 4% near the core periphery. However, using 1,000 clusters, the reference solution is much better preserved, with an eigenvalue error of less than 20 pcm and PFD error within 2% (99.4% of pins within 1%).

3.3. Uncertainty Quantification

The k -means algorithm is subject to a degree of stochastic uncertainty as a result of the randomness of the initial guess and the sensitivity of the algorithm to initial conditions. While selecting the best

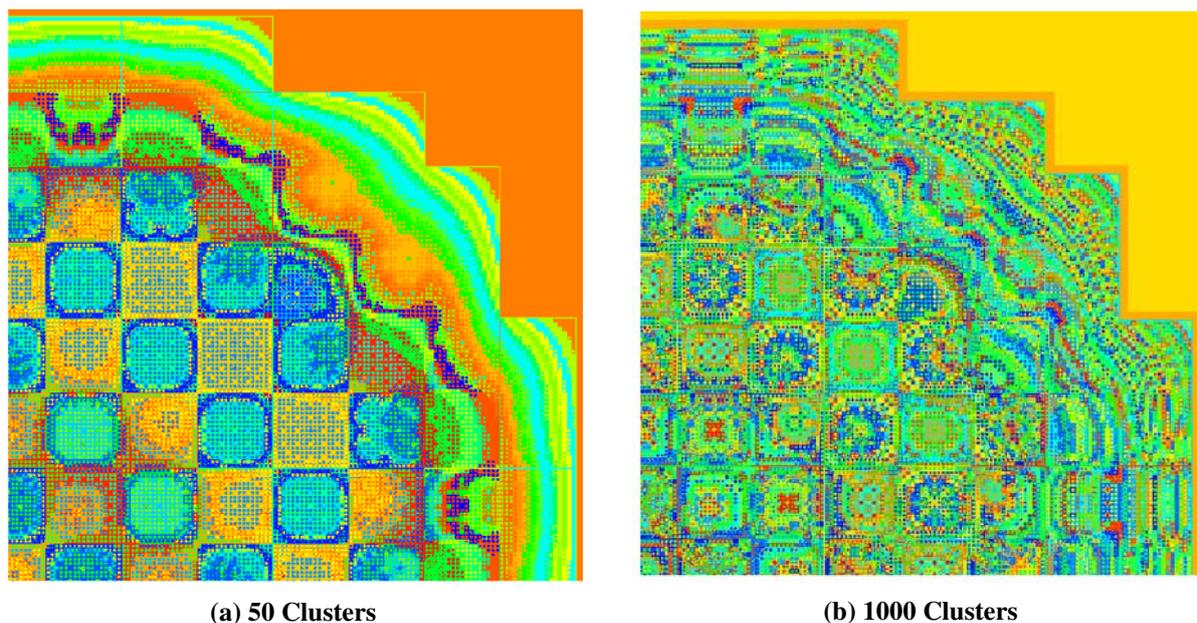


Figure 2: Example clustering of spectra regions with 30 realizations of the k -means algorithm.

Table I: Summary of Jaguar Calculations with Clustering

Case	Spectra Regions	Eigenvalue	Clustering Error [pcm]
Reference	57,760	0.99773	—
k -means	50	0.99826	53
k -means	100	0.99822	49
k -means	200	0.99811	38
k -means	300	0.99796	23
k -means	500	0.99795	22
k -means	1,000	0.99788	15
k -means	5,000	0.99781	9

clustering from several realizations of the k -means algorithm can partially mitigate this uncertainty, it does not wholly eliminate it. Additionally, the magnitude of the uncertainty in the result may vary depending on the number of clusters chosen. It is important, therefore, to quantify the uncertainty in the coarse-group results as a consequence of using the k -means algorithm.

To achieve some understanding of this uncertainty, the calculations in the previous section were each repeated 50 times. The eigenvalue distributions from these calculations are presented in Fig. 4. That there is no guarantee of normality in the distributions, and in fact, the 300-cluster

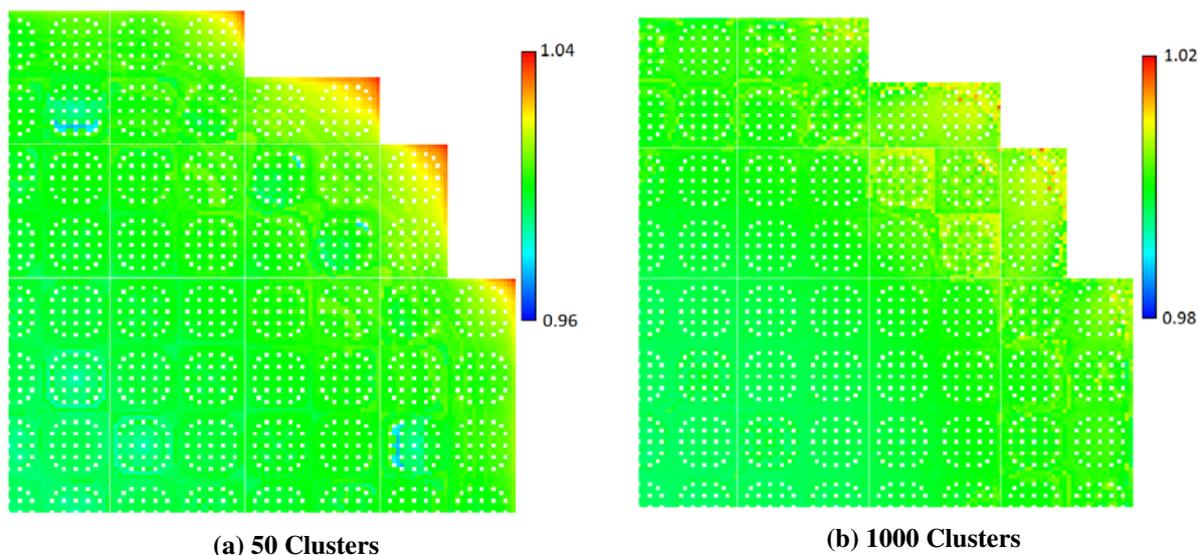


Figure 3: Ratio of PFD using 50 and 1,000 clusters to the reference calculation.

distribution appears bimodal, likely because the initial conditions randomly cause the clustering to either capture some important physical effect or not. Nonetheless, a general trend is observed that increasing the number of clusters reduces the error in the eigenvalue and the uncertainty of the result. In Table II, the eigenvalue results from the uncertainty analysis are summarized, and in Fig. 5, the standard deviation of the PFD is presented for the 50- and 1,000-cluster cases. This provides a measure of to what degree and in what core locations the clustering algorithm struggles to determine the appropriate cluster for the region.

Table II: Average Eigenvalue Error and Standard Deviation

Clusters	Error [pcm]	Std. Dev. [pcm]
50	59.3	7.1
100	46.5	3.8
200	37.3	3.5
300	29.0	4.6
500	20.5	1.8
1,000	14.5	1.4
5,000	8.2	1.8

4. CONCLUSIONS

This paper highlights the unique capability of data analytics to capture the underlying physics of a system with limited or no knowledge of the application. In this case, a clustering algorithm applied

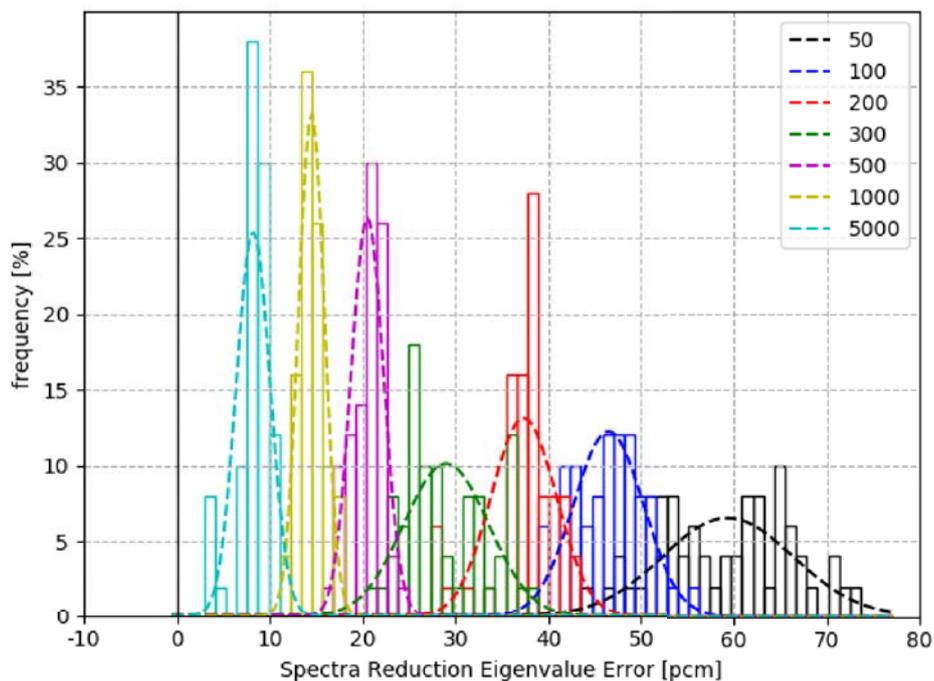


Figure 4: Histogram of eigenvalue error of the k -means algorithm with 50-5,000 clusters using 30 realizations per sample and 50 samples. Dashed lines are normal distributions fit to the data.

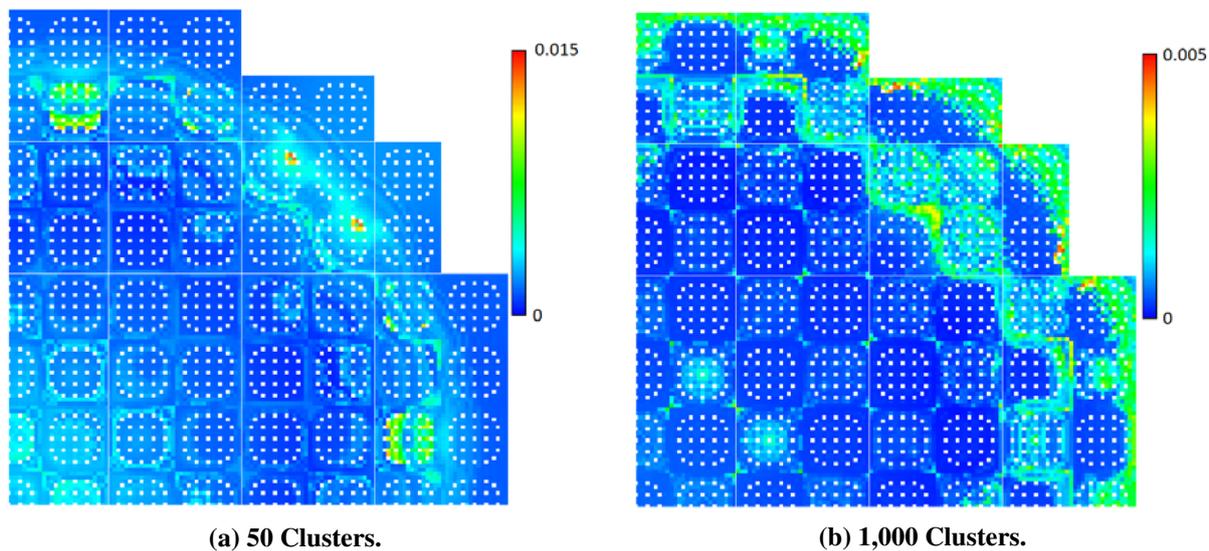


Figure 5: Relative standard deviation of the pin fission density using 50 and 1,000 clusters and a sample size of 50.

to weighting spectra can enable improved efficiency in a multigroup cross section generation workflow by removing redundant data without burdensome manual processing. Because this approach is agnostic to the physics of the system, it can be of particular value when analyzing novel reactor physics concepts for which the lack of broad experience and engineering judgment can make cross section processing even more challenging.

In the test problem, the k -means clustering algorithm demonstrated that a reduction of 98.3% in the number of spectra regions from 57,760 spectra regions to only 1,000 has minimal impact on the accuracy of the solution. The eigenvalue of the 1,000-cluster calculation matched the reference within 14.5 pcm with pin fission density error generally within 1%. That magnitude of reduction in number of spectra is of particular value given that weighting spectra vary with burnup, moderator density, and soluble boron concentration, magnifying the savings in data size and access time. While the k -means algorithm does carry with it an uncertainty due to the initial guess, the uncertainty is small, resulting in a standard deviation of only 1.4 pcm in eigenvalue and a maximum of 0.5% in pin fission density for the 1,000 cluster case.

ACKNOWLEDGEMENTS

This work has been authored by Fluor Marine Propulsion, LLC, under Contract No. DOE-89233018CNR000004 with the U.S. Department of Energy. The United States Government retains and the publisher, by accepting this article for publication, acknowledges that the United States Government retains a non-exclusive, paid-up, irrevocable, and world-wide license to publish, distribute, translate, duplicate, exhibit, and perform the published form of this manuscript, or allow others to do so, for United States Government purposes.

REFERENCES

- [1] H. Steinhaus. “Sur la division des corps materiels en parties.” *Bulletin de L’Academie Polonaise des Sciences, Cl III*, **volume IV** (1956).
- [2] E. Jones, T. Oliphant, P. Peterson, et al. “SciPy: Open source scientific tools for Python.” (2001–). URL ”<http://www.scipy.org/>”.
- [3] A. T. Godfrey. “VERA Core Physics Benchmark Progression Problem Specifications.” Technical Report CASL-U-2012-0131-004 (2014).
- [4] A. Watson, R. Grove, and M. Shearer. “Effective software design for a deterministic transport system.” In *International Conference on Advances in Mathematics, Computational Methods, and Reactor Physics* (2009).
- [5] J. R. Deen and W. L. Woodruff. “WIMS-D4M User Manual, Rev. 0.” Technical Report ANL/RERTR/TM-23 (1995).
- [6] C. Bergiers, B. Ivanov, and K. Ivanov. “Establishment of Consistent Benchmark Framework for Performing High-Fidelity Whole Core Transport/Diffusion Calculations.” In *PHYSOR-2006, ANS Topical Meeting on Reactor Physics* (2006).
- [7] M. DeHart et al. “Preliminary Results for the OECD/NEA Time Dependent Benchmark using Rattlesnake, Rattlesnake-IQS, and TDKENO.” Technical Report INL/EXT-16-39723 (2016).