EPJ Web of Conferences **302**, 04006 (2024) https://doi.org/10.1051/epjconf/202430204006
SNA + MC 2024

The effect of dominance ratio on the statistical convergence of sensitivity in Monte Carlo codes

Mohammad Rababah[1*], *Adrien* Bidaud[1]

[1]LPSC, Université Grenoble-Alpes, CNRS/IN2P3 53, rue des Martyrs, 38026 Grenoble, France

Abstract. Sensitivity computation in Monte Carlo-based codes is widely used and has been enhanced significantly. Although sensitivity converges in most reactor configurations, it encounters difficulties in large core designs. We suspect that this issue might be linked to the dominance ratio. To test this hypothesis, we develop a simple benchmark to validate whether statistical convergence of sensitivity depends on dominance ratio, and if it may be linked to other factors, such as neutron energy spectrum. The benchmark's simplicity enables us to calculate the dominance ratio analytically and use eigenmodes decomposition to compute sensitivity to total fission neutron yields, substantially lowering computational costs. This method results in a good match when compared to the direct method which serves as a reference. We clearly see regular sensitivity convergence speed behavior linking it with the number of latent generations and the dominance ratio. Therefore, we establish a formula to recommend the number of latent generations required for sensitivity to converge, thus significantly saving computational resources.

1 Introduction

The utilization of the Monte Carlo method in reactor physics has notably expanded with the advancements in computational resources and the rapid growth of computing power, and its use for sensitivity computation has evolved with the integration of perturbation theory in numerous Monte Carlo-based codes. Despite the enhancements in the sensitivity calculation, little research has focused on the statistical convergence of sensitivities.

The convergence of local variables such as local fluxes in Monte Carlo may be much slower than global ones, e.g., k-eff. The convergence of their sensitivity with the number of histories simulated is not well known. Their actual convergence may be doubtful as it depends on "new" parameters of the Monte Carlo method used for the sensitivity calculations such as the number of "latent" generations. Latent generation is the terminology used in SERPENT [1], defined by the number of propagation generations through which events (perturbations) are weighted to compute sensitivity. The same concept is called differently in other codes, such as propagation generations or propagation batches [2] and super-generations [3].

While the literature review shows that sensitivity converges with 10-20 latent generations in typical power reactors [3,4], recent research [5,6] shows that even with 100

* Corresponding author: mohammad.rababah@lpsc.in2p3.fr

latent generations, the sensitivity of local variables, such as power distribution, does not converge for large reactors such as UAM-UOX GEN III Benchmark [7].

We wonder whether the factor is not the size but rather the dominance ratio, which relates the size of the reactor with the neutron migration length. It is known that even the k-eff and power distribution convergence depends on the dominance ratio, and that the number of batches required for this convergence varies as a function of the dominance ratio [8], in any deterministic or stochastic code using the power iteration method. The method to calculate sensitivities implemented in Monte Carlo codes relies on the iterated fission probability and it requires weighting the neutron importance over the latent generations. Therefore, we want to examine whether the convergence speed of sensitivities with the number of latent generations has the same behavior as the convergence speed of k-eff or flux with the number of batches.

Furthermore, we know that the sensitivity can be decomposed on eigenmodes. The effect of perturbations on fundamental fission source distribution can be represented by [9,10]:

$$\frac{dS_0}{dP} = \sum_{i=1}^{\infty} S_i . \frac{S_i^{\dagger} \frac{dF}{dP} S_0}{k_0 \left(1 - \frac{k_i}{k_0} \right)} \tag{1}$$

Where $\frac{dS_0}{dP}$ represents the derivatives of the fundamental fission source distribution S_0 to the perturbation P. $\frac{dF}{dP}$ represents the effect to the perturbation in P on the fission kernel $F.S_i$, $S_i^{\dagger}$ and k_i form the i^{th} eigentriplet.

The dominator clearly shows the importance of the dominance ratio on the change in the source to the perturbation. This means that the higher the dominance ratio, the more modes may contribute to sensitivity, which may require a larger number of latent generations to be evaluated accurately.

In the first section of this paper, we provide more results about the sensitivity computation of the UAM-UOX benchmark using different options implemented in Serpent. Then we design a benchmark to understand the convergence issue, by having a geometry with two distinct dominance ratios and two neutron energy spectra: a thermal one matching the UAM-UOX benchmark, and a fast spectrum.

We provide an analytical computation of the dominance ratio using one group diffusion equation and compare it with the values obtained by the Fission Matrix obtained by Serpent.

Another section provides three different methods to compute sensitivities: direct sensitivity computation, sensitivities produced by Serpent, and sensitivities calculated by the use of eigenmodes.

In the last section, we provide the results of the sensitivity of k-eff and power distribution to the change in total fission neutron yields (ν) and to the total cross-section, and discuss the effect of the dominance ratio on the convergence speed.

2 Sensitivity convergence in UAM-UOX

Recent research [5, 6], shows that even with 100 latent generations, the sensitivity of local variables, such as power distribution, does not converge in large reactors, such as UAM-UOX. To examine the issue, we compute the sensitivity of the power distribution to the change in total cross-section for the UAM-UOX, using three different options that are available in Serpent, these options are:

- Default: this is the default way used by Serpent. It creates so-called event-objects during each accepted or rejected event of interest (e.g., collision, sampled fission neutron energy, or sampled scattering angle). Then, when the net number of accepted collisions is needed, this list of events is traversed and the net number is calculated and integrated into the sensitivity computation.

- Score matrix: this option updates the net number of a certain event on-the-fly, every time that event is sampled. Event objects are no longer stored. Instead, the net number of accepted collisions is either incremented (accepted collision) or decremented (rejected collision) directly.
- Batching interval: in Serpent, the statistics are divided in batches, and by default, each generation forms its own batch. This should not change the expected value but may change the statistical uncertainty.

We compute the sensitivity of power distribution to the change in total cross-section using these three options. The reference value is directly calculated by the direct method, i.e., by changing the material density, which is equivalent to changing the total cross-section and looking at the induced relative change in power distribution.

In theory, there should be no difference between the score matrix and default methods, however, there is about a 20-30% difference, Figure 1. This is not compatible with the statistical uncertainties unless those are underestimated, or “not accurate”.

In Figure 1 middle, we compare the default value with different batching intervals, 1 and 50. Sensitivity values are almost identical, yet the standard deviation is 4 times larger when using 50 batching intervals compared to 1. This makes the expected value closer to being statistically compatible with the reference one at 100 latent generations. When uncertainties are calculated over 50 batches instead of at each batch, they tend to be better estimated. We know from [5], that the variance increases with the number of latent generations as shown in Figure 1 middle, and we are surprised to see that the difference between 50 and 1 batching intervals grows that vast with the number of latent generations. We suspect that estimating the standard deviation of sensitivity requires as many numbers of batching intervals as the number of latent generations needed to correctly estimate sensitivity, whose values will be discussed later.

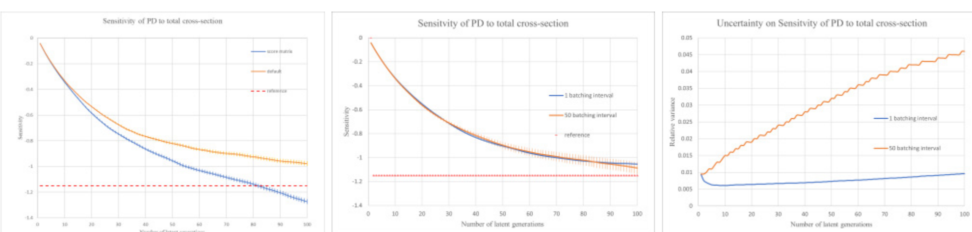

Figure 1: Sensitivity of power distribution to the change in total cross-section, UAM-UOX benchmark.

It is clear that whichever method we use, the sensitivity of power distribution does not converge even with 100 latent generations. Since the UAM-UOX reactor configuration is quite complex, we designed a simple representative benchmark (Homogenous Core Benchmark) to narrow down the factors that impact the convergence.

3 Methodology and benchmark description

3.1 Benchmark description

The proposed geometry is a parallelepiped, cut in half on the x-axis, filled with homogenous material composed of either uranium carbide or uranium oxide, to achieve fast and thermal spectra. By using reflective boundary conditions, we can choose the dominance ratio by changing the size without changing the physics, such as k-eff and neutron energy spectrum.

The simplicity of the benchmark allows us to calculate the dominance ratio analytically and to use eigenmodes decomposition to compute sensitivity to total fission neutron yields.

Table 1 shows the geometry characteristics. Figure 2 left shows the geometry plot XY plane, and the right figure shows the second eigenmode, obtained via Fission Matrix. Figure 3 shows the neutron energy spectrum for the UAM-UOX, thermal, and fast spectrum of the homogeneous core benchmark.

Table 1. Main characteristics of the Benchmark.

Spectrum		Dimensions X, Y, Z	Dominance ratio	Material composition
Fast	Large	300, 200, 100	0.992	Uranium Carbide
	Small	60, 40, 20	0.847	
Thermal	Large	160,100,40	0.992	Uranium Oxide
	Small	30,20,5	0.841	

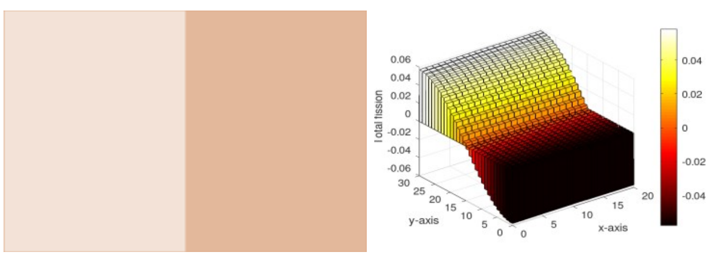

Figure 2. Left: Geometry plot of the parallelepiped, x-axis view, right: Second eigen mode obtained via Fission Matrix.

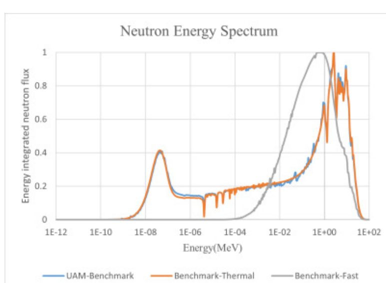

Figure 3. Neutron Energy Spectrum

We run each sensitivity simulation for 10000 batches, with 50000 and 100000 neutrons per batch, for small and large geometry respectively, with 100 latent generations, and 44g energy grid from the ENDF/B-VII.1 evaluated nuclear data library [11] that we use for uncertainty analysis. Notably, such computations demand substantial computational resources; a single simulation with 100 latent generations, 50,000 neutrons per batch, and 10,000 cycles required approximately 21 days of runtime on 72 CPUs allocated 78 GB of memory.

3.2 Eigenvalues and Eigenmodes

The Fission Matrix method implemented in Serpent is simple to use. By post-processing the Fission Matrix obtained via Serpent2.2., we can extract the eigenvalues and eigenmodes. But since we have a simple geometry, we can also calculate the eigenvalues using the one energy group diffusion equation. In this section, we provide a comparison as a way to validate the results of Serpent.

From the general transport equation, assuming a steady state system and a single energy group for neutrons, the equation can be simplified to the one-group diffusion equation

$$D\nabla^2\phi(\vec{r})+\Sigma_a\phi(\vec{r}) = \left(\frac{1}{k}\right)\nu\Sigma_f\,\phi(\vec{r}) \tag{2}$$

Where $\phi(\vec{r})$ is the neutron flux, D is the diffusion coefficient, Σ_a is the macroscopic absorption cross-section, $\nu\Sigma_f$ is the neutron production term, k is the effective multiplication factor, and values of D, Σ_a, and $\nu\Sigma_f$ are obtained by Serpent.

When reflective boundary conditions are applied to the parallelepiped surfaces, the derivative of each component of the flux is zero at the surface. The neutron flux can be expressed as a product of each flux component:

$$\varphi(x, y, z)= X(x)Y(y)Z(z) \tag{3}$$

This leads to a set of ordinary differential equations based on the diffusion equation and the applied boundary conditions, which when solved leads to an eigenvalue problem, where the equation for eigenvalues is:

$$B_x^2+ B_y^2+B_z^2-\frac{\Sigma_a}{D} = -\nu\frac{\Sigma_f}{kD} \tag{4}$$

where B_x , B_y , and B_z are constants determined by the boundary conditions, and k is the eigenvalue representing the effective multiplication factor.

All eigenvalues of the system can be presented as in equation (4),

$$k_{n,m,l} = \frac{\nu\Sigma_f}{\Sigma_a+D\left(\left(\frac{\pi}{2X}n\right)^2+\left(\frac{\pi}{2Y}m\right)^2+\left(\frac{\pi}{2Z}l\right)^2\right)} \tag{5}$$

The first eigenvalue is when n, m, and $l=0$ which leads to $k_{0,0,0}= k_{eff}$ and a flat flux. Table 2 shows a good match between values calculated using the diffusion equation and using the Fission Matrix method. K-eff obtained directly from Serpent is 1.0216, which is 10 pcm from k_0 calculated from the Fission Matrix for which we used 30,20, and 1 meshes on X, Y, and Z respectively.

Table 2. Eigenvalues for the 0.99 DR configuration, fast spectrum.

	k_0	k_1	k_2	k_3	k_4
Fission Matrix	1.0217	1.0138	1.0041	0.9965	0.9909
Diffusion equation	1.0180	1.0101	1.0004	0.9928	0.9871
Relative difference	0.36%	0.36%	0.37%	0.37%	0.38%

3.3 Sensitivity computation

We perturb ν and total cross-section in half the geometry, and we use the continuous energy Monte Carlo code Serpent2.2. to compute the sensitivity of both effective multiplication factor and asymmetry of power distribution, i.e., energy deposition in half the geometry normalized to the full geometry, to these changes. Only energy-integrated sensitivities of both the effective multiplication factor and the power distribution to the total cross-section are presented because of the similarity in the behavior of sensitivities to other cross-sections.

We present here three methods to compute sensitivities, direct method, sensitivities using GPT via Serpent [1,12], and using the Fission Matrix and eigenmodes [9,10].

3.3.1 Direct method

This method comprises two Monte Carlo simulations, one without perturbation and one with perturbing the parameter of interest. We use a 10% change in the total density to represent the perturbation to the total cross-section, and we change ν by 0.3% to obtain the sensitivity to the change in ν. This method, while simple, is quite sensitive to the amount of change to remain in the domain of small perturbations. That is why those changes were chosen to minimize statistical uncertainty and have correct values. If changes were large, they would not represent the first-order sensitivities, and if they were small, there would be no visible difference in the k-eff or the power distribution.

3.3.2 Sensitivity via Serpent

Serpent relies on the collision history based first order GPT equivalent implementation to calculate the sensitivity of various responses to various perturbations. In version 2.1.31, these capabilities were extended to be used for statistical uncertainty propagation of Serpent estimates.

3.3.3 Sensitivity via eigenmodes

One way to compute the sensitivity of the fundamental fission source is by decomposing power distribution perturbation on modes [9], [10], as per equation
(1).

In the case of sensitivity to ν, the $\frac{dF}{dP}\frac{P}{F}$ is 1 as an increase of 1% of ν will increase the number of neutrons produced at the next generation by 1% exactly. In this case, we can compute the direct and adjoint functions needed in equation (1), for instance using the eigenfunctions of the Fission Matrix and its transposed matrix, to obtain the sensitivity of power distribution to a change in ν.

4 Sensitivity computation results

In this section we present the sensitivity of the effective multiplication factor and the power distribution (asymmetry of the fission source) to the total cross-section and total ν obtained by Serpent2.2., the reference values calculated by the direct method, and the one given by the eigenmodes formula.

While we have performed the computation for both thermal and fast spectrum cases, we will show the figures only for the thermal case, since the convergence trend and speeds are similar. However, Table 3 shows the sensitivity of power distribution to the change in total ν for both spectra and for both DR obtained via three methods.

When comparing the reference value to other methods, we see that sensitivity in Serpent converges when the dominance ratio is 0.85, but it is very far from converging in the 0.99 DR case.

The sensitivity obtained by decomposing power distribution on modes, whether obtained by the Fission Matrix or analytically may predict the values regardless of the dominance ratio. The difference between the analytical and Fission Matrix solutions could be related to the difference between the first and second eigenvalues deduced from the Fission Matrix and the diffusion equation. Moreover, the computational cost, which took just a few hours, is significantly reduced compared to the sensitivity obtained via Serpent which took 21 days. More work will be done, probably to refine the Fission Matrix and to reduce the gap with the reference value.

Table 3. Sensitivity of power distribution to total ν.

DR	Spectrum	ref-sensitivity	Serpent -Sensitivity at 100 latent generation	Eigenmodes Fission Matrix	Eigenmodes Analytical
0.85	Thermal	2.01	2.16	2.60	2.69
0.99		63.02	31.02	56.04	63.81
0.85	Fast	2.40	2.32	2.73	2.53
0.99		54.33	29.76	52.31	52.45

4.1 Sensitivity up to 100 latent generation

We present sensitivity results up to 100 latent generations. Figure 4 represents the k-eff sensitivity to the total cross-section, in both DR configurations. The sensitivity in the 0.85 DR configuration is very small, but the sensitivity is statistically compatible with the reference value and with zero despite the important computational effort. By contrast, in the case of the high dominance ratio configuration, the sensitivity is not statistically compatible with the reference value. This raises the question of whether the statistical uncertainties on sensitivity are well estimated as discussed in part 2.

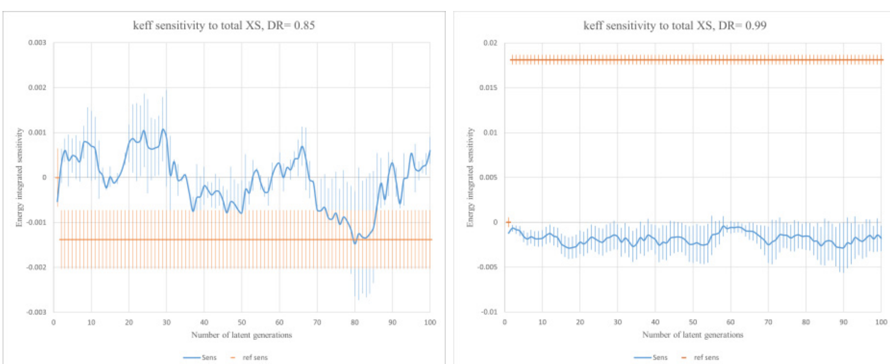

Figure 4. Sensitivity of k-eff to the total cross-section for small and large dominance ratio.

Figure 5 and Figure 6, show respectively the sensitivity of power distribution and k-eff to total ν for both dominance ratios. We see the impact of the dominance ratio on the convergence speed of power distribution sensitivity. For the 0.85 DR, the sensitivity converges within around 30 latent generations. While, for 0.99 DR, sensitivity does not converge with 100 latent generations, it is growing and may need more latent generations to converge.

The four figures show a very regular convergence behavior, that is compatible with $1\text{-}DR^{NLG}$. Where *DR* is the dominance ratio and *NLG* is the number of latent generations. We use a fit function $f=T\left(1\text{-}DR^{NLG}\right)$. Where T is the target value, ideally, the reference value is calculated by the direct method.

Figure 7 shows the sensitivity of k-eff to changes in total ν in half the geometry, which should match a theoretical change of 0.5%/% by the change in total ν. While the convergence speed is the same and seems directly achieved with one latent generation, the estimation of sensitivity of Serpent is well-matched with the reference one in 0.85 DR, unlike the case of sensitivity in 0.99 DR. Nevertheless, once again, part 2 shows that we may not rely on the statistical uncertainties of sensitivity given by Serpent. There is also a difference in the convergence speed with the number of latent generations, between total ν sensitivity and total cross-section sensitivity.

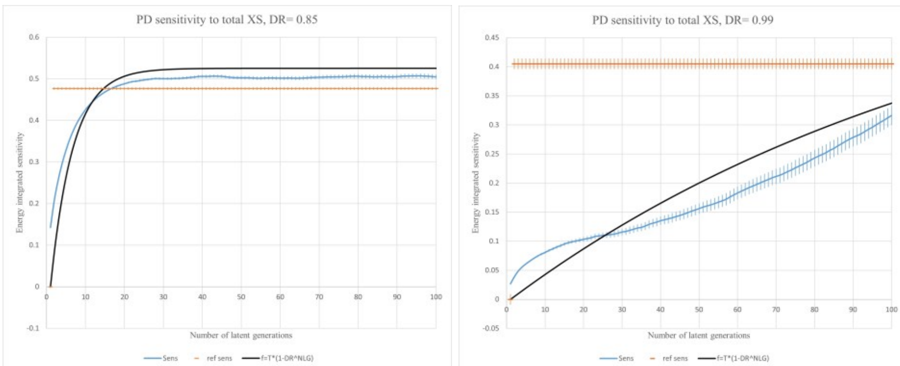

Figure 5.Sensitivity of power distribution to the total cross-section for small and large dominance ratio.

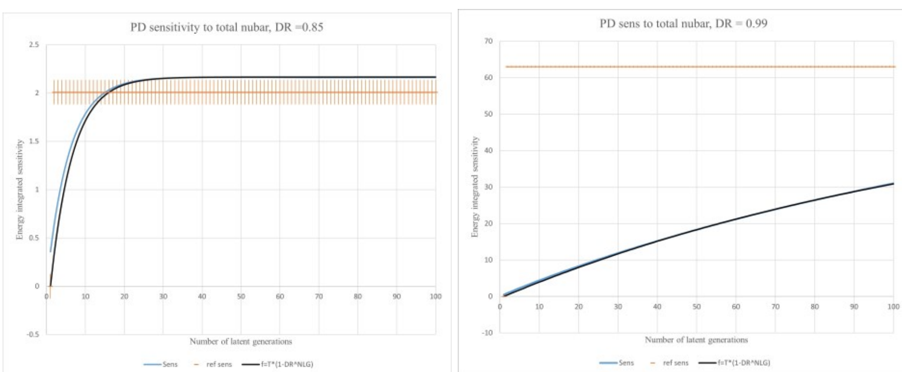

Figure 6. Sensitivity of power distribution to total ν for small and large dominance ratio.

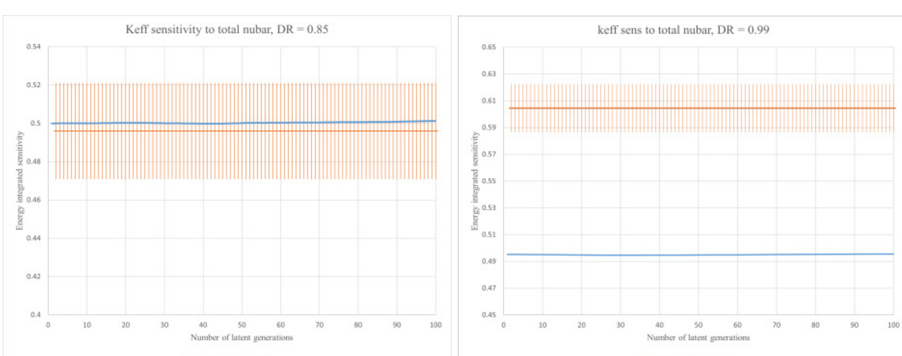

Figure 7. Sensitivity of k-eff to total ν for small and large dominance ratio.

4.2 Sensitivity beyond 100 latent generation

The more we increase the number of latent generations, the more computation time and memory are needed. Serpent uses 100 latent generations as the maximum, and the source code was changed to achieve up to 600 latent generations and obtain the results of Table 4 for the sensitivity of power distribution to the change in total ν.

From Figure 5, Figure 6, and Table 4, we see a very regular convergence behavior, as the evolution of the sensitivity is compatible with $T\left(1\text{-}DR^{NLG}\right)$.

It is demonstrated in [8] that we can assess the dominance ratio from the power iteration process used to compute the neutron flux, and it is shown that the dominance ratio can be deduced from an exponential fit of the expression of the flux expressed as a function of the cycle number. We suspect commonalities between the convergence of flux and the convergence of sensitivities, and we can perform some prediction of the latent generation number for which the sensitivity converges to some X percentage of the final value, for instance for the 0.99 DR case, using the following equation

$$\textit{number of recommended latent generations} = \frac{ln(1\text{-}X\%)}{ln(DR)} \quad (6)$$

573 is the number of latent generations at which the sensitivity is expected to be calculated with less than 1% difference to the reference value using Equation (6) for which Serpent gives a sensitivity that is less than 1% from it.

Table 4. Sensitivity of power distribution as a function of latent generation.

Latent generation	100	200	300	400	500	600	**573**
Sens-Serpent	31.02	45.66	52.68	61.04	61.84	62.98	**62.39** **Eq.** (6)
Sens-reference	63.02						

5 Conclusion

In this paper, we demonstrate that the convergence of sensitivity computations is significantly influenced by the dominance ratio. This is crucial, especially for large reactor configurations like UAM-UOX, where sensitivity computations are computationally very demanding.

The introduction of a simple, representative benchmark allows us to examine and analyse the effect of the dominance ratio and neutron energy spectrum on the convergence of sensitivity with the number of latent generations. The employment of eigenmodes decomposition provides a more efficient and less resource-intensive approach for such computations when sensitivity to total ν is searched.

Our results demonstrate a clear relationship between the dominance ratio and the number of latent generations required for achieving convergence in sensitivity analyses. This understanding allows us to establish a formula to recommend the number of latent generations required for sensitivity convergence, thus allowing researchers and users to minimize efforts and computational resources.

We also compare different options for sensitivity computations, highlighting the effect of batching interval on the accuracy of estimating statistical uncertainties. We suspect that the batching interval needed to estimate uncertainties could be set equal to the number of latent generations required for sensitivity to converge. This will be the focus of future work. We also want to generalize the use of eigenmodes to compute sensitivity in Serpent as done in [10], consequently, accelerating convergence and reducing computation costs.

Acknowledgments

We express our gratitude to the CNRS for funding this research through the joint project NEEDS, in which CNRS, CEA, and IRSN work together to study the propagation of uncertainty in fuel burn-up calculations. We also would like to thank our colleagues from LPSC for their support and Bastien Guillemet for proofreading the paper.

References

[1] M. Aufiero, A. Bidaud, and M. Hursin, “A collision history-based approach to sensitivity/perturbation calculations in the continuous energy Monte Carlo code SERPENT,” *Annals of Nuclear Energy*, 2015.

[2] Y. NAGAYA and T. MORI, “Impact of perturbed fission source on the effective multiplication factor in monte carlo perturbation calculations,” *Journal of Nuclear Science and Technology*, vol. 42, no. 5, pp. 428–441, 2005, doi: 10.1080/18811248.2005.9726411.

[3] B. C. Kiedrowski, “Review of Early 21st-Century Monte Carlo Perturbation and Sensitivity Techniques for k-Eigenvalue Radiation Transport Calculations,” *NUCLEAR SCIENCE AND ENGINEERING*, 2017.

[4] B. Kiedrowski and F. Brown, “AdjointWeighted Tallies for k-Eigenvalue Calculations with Continuous-Energy Monte Carlo,” *Nuclear Science and Engineering*, 2011.

[5] M. Rababah and A. Bidaud, “Computation of Sensitivity in Monte-Carlo codes: Investigation of Convergence for Reactor Cores with High Dominance Ratio,” presented at the International Conference on Physics of Reactors (PHYSOR 2024), San Francisco, CA, 2024, doi: 10.13182/PHYSOR24-43648.

[6] P. Lopez and A. Bidaud, *Sensitivity analysis and its convergence through Monte Carlo calculations for the UAM Gen-III benchmark: application to power distributions*. United States: ANS - American Nuclear Society, 2021. [Online]. Available: http://inis.iaea.org/search/search.aspx?orig_q=RN:54081746

[7] k Ivanova and M. Avramova, “Benchmarks for Uncertainty Analysis in Modelling (UAM) for the Design, Operation and Safety Analysis of LWRs - Volume I: Specification and Support Data for Neutronics Cases (Phase I),” Organisation for Economic Co-Operation and Development, 2013.

[8] E. Dumonteil and T. Courau, “Dominance Ratio Assessment and Monte Carlo Criticality Simulations: Dealing with High Dominance Ratio Systems,” *Nuclear Technology*, vol. 172, no. 2, pp. 120–131, 2010, doi: 10.13182/NT10-A10899.

[9] R. B. Nelson, “Simplified calculation of eigenvector derivatives,” *AIAA Journal*, vol. 14, no. 9, pp. 1201–1205, 1976, doi: 10.2514/3.7211.

[10] M. Aufiero and M. Fratoni, “A new approach to the stabilization and convergence acceleration in coupled Monte Carlo–CFD calculations: The Newton method via Monte Carlo perturbation theory,” 2017.

[11] M. B. Chadwick and et al., “ENDF/B-VII.1 Nuclear Data for Science and Technology: Cross Sections, Covariances, Fission Product Yields and Decay Data,” *Nuclear Data Sheets,* vol. 112, pp. 2887–2996, 2011.

[12] M. Aufiero, M. Martin, and M. Frat, “XGPT: Extending Monte Carlo Generalized Perturbation Theory capabilities to continuous-energy sensitivity functions,” *Annals of Nuclear Energy*, 2016.