Performance Evaluation of a New Transport Operator in SCONE Monte Carlo Code

Congjin Ding[1,*], *Mikolaj Adam* Kowalski[1], and *Eugene* Shwageraus[1]

[1]University of Cambridge, Department of Engineering, Trumpington Street, Cambridge CB2 1PZ, United Kingdom

Abstract. A novel transport operator, called "HELL", has been developed and integrated into the SCONE Monte Carlo code. This alternative approach combines two commonly used tracking methods: surface-tracking and delta-tracking. The unique aspect of this operator is its flexibility, allowing users to assign these tracking methods to different geometry universes according to their specific requirements. The methodology of HELL and its implementation in SCONE is presented and detailed. The performance and efficiency of the HELL have been evaluated through its application in two different cases. The first case involved a core model featuring involute surfaces, while the second case focused on a Pressurized Water Reactor (PWR) application. The results demonstrated that HELL effectively eliminates the limitations typically associated with surface-tracking in complex geometries and delta-tracking in materials with localized heavy absorbers. This makes it a valuable tool in nuclear physics simulations and computations. The HELL method can significantly reduce the calculation time required for such scenarios, and can be widely applied to Monte Carlo modelling in reactor physics.

1 Introduction

Monte Carlo simulations are a powerful tool for neutron transport modelling in reactor physics. They primarily function by explicit stochastic simulation of particle histories within a geometric model. This method is highly flexible and can be used to model complex geometries with high fidelity. Despite their effectiveness, these simulations can be computationally demanding, necessitating substantial computing resources to achieve precise results. One approach to mitigate this limitation involves accelerating the transport simulation through the optimization of particle tracking.

By particle tracking, in the context of Monte Carlo simulation, we mean a procedure that moves particles between consecutive collisions. For neutral particles two approaches are commonly used: surface-tracking (ST) and delta-tracking (DT), each offering its own unique advantages, albeit with certain constraints. ST performance is influenced by surface distance calculations and boundary crossings in complex geometries [1, 2]. Meanwhile, DT exhibits reduced performance when localized heavy absorbers, like control rods or burnable poisons, are present [3–5]. To overcome these constraints, we propose a novel method called "HELL". The term refers to particles tracking at different universe levels (circles). In several

*e-mail: cd825@cam.ac.uk

Monte Carlo codes, the geometry can be divided into separate regions known as "universes". Each universe constitutes an independent subdivision of the entire space into cells which can be filled with materials or other universes. HELL optimizes computational efficiency by combining the ST and DT in different universes. Unlike the HOLE geometry used in Monte Carlo code MONK [6, 7], HELL does not require separate geometry definitions for the DT regions. This flexibility allows to vary the decomposition into ST and DT regions without difficulty or potential for user errors during translation from one geometry input syntax to another. It also eliminates the need to superimpose a mesh over the geometry, as required by Multi-Regional delta-tracking [8].

SCONE (Stochastic Calculator Of Neutron Transport Equation) is an object-oriented Monte Carlo particle transport code designed for reactor physics[1]. The code has been compared against Serpent and has shown sufficient accuracy to be used for teaching and proof-of-concept applications [9].

In this paper, we present an implementation of the HELL algorithm within SCONE. The performance of this method is subsequently evaluated through its application to two representative case studies: a core with involute surfaces and a PWR core model inspired by the Hoogenboom-Martin benchmark [10]. The paper is structured as follows: we first introduce the methodology of the transport operator and present the two primary tracking methods in Section 2; we detail the algorithm and its implementation within SCONE in Section 3; we construct the core models and perform calculations in two distinct cases to evaluate the performance of HELL in Section 4; then, we conclude with presenting our findings in Section 5.

2 Methodology of transport operator

In this section, we introduce the two primary methods of particle tracking in Monte Carlo codes: surface-tracking (ST) and delta-tracking (DT).

ST is a widely used conventional method in Monte Carlo simulations to track particles through a geometry. It follows a random walk from one interaction to the next. The distance that a particle travels in a material can be expressed as a sum of distances traveled between interactions [1, 11], while the distance between subsequent collisions can be sampled with:

$$\mathrm{dist} = -\frac{\log(\xi)}{\Sigma_{\mathrm{tot,m}}(\mathrm{E})}\ , \tag{1}$$

where ξ is a uniformly distributed random variable on the unit interval; $\Sigma_{\mathrm{tot,m}}(\mathrm{E})$ is the total macroscopic cross section for a given energy E of the particle in the material m, which describes the interaction probability per path length travelled by the neutron [11].

However, the distance in Eq. (1) is not valid when the particle crosses a material boundary. The surface-tracking method calculates the nearest distance to the current cell boundary surface (potential material composition interface) along the direction of the particle's flight and compares it to the sampled path distance to select whether the particle can move up to its next collision or whether it needs to cross into the next cell with the distance to the collision being resampled. Such a procedure is permitted due to the Markov property of the exponential distribution. Therefore, the performance of ST depends on the cost of the calculating surface distances and the number of boundary crossings between collisions [1].

DT is another particle transport method proposed by Woodcock [3] and verified by Coleman [12]. It was popularised by the Monte Carlo code Serpent [4]. Unlike surface-tracking, DT uses the concept of virtual collisions to perform distance sampling. This method calculates the majorant macroscopic cross section $\Sigma_{\mathrm{maj}}(\mathrm{E})$, which is the maximum material total

[1] https://github.com/CambridgeNuclear/SCONE

cross section in the system at each energy point. This ensures the distance will be valid in all materials and avoids distance calculation to cell interfaces. However, a rejection sampling step is needed to compensate for sampling the distance with a cross section larger than the physical one. A random number ξ is sampled and a collision is rejected if ξ is larger than $\frac{\Sigma_{\mathrm{tot,m}}(\mathrm{E})}{\Sigma_{\mathrm{maj}}(\mathrm{E})}$ (virtual) and accepted if ξ is smaller than $\frac{\Sigma_{\mathrm{tot,m}}(\mathrm{E})}{\Sigma_{\mathrm{maj}}(\mathrm{E})}$ (real). Following a virtual collision, the direction and energy of the particle remain unchanged. Real collisions are processed normally.

The rejection sampling algorithm may be inefficient in materials where the local cross section is significantly lower than the majorant cross section, since the number of virtual collisions will be large in this case. This is particularly true in the presence of localised heavy absorbers such as control rods and burnable absorbers. In such problems, DT may exhibit particularly poor performance [4].

Hybrid-tracking (HT) is an option that can be used to overcome the efficiency problems. It is a mixture of ST and DT, and can switch between DT and ST by comparing the ratio between the local material cross section and the majorant cross section to a preset cut-off constant. If, at the particle energy, the ratio is larger than the cut-off, DT is used; otherwise ST is used [1]. HT is generally applied to the entire geometry and does not allow for the separation of high cross-section materials from the majorant. As a result, it prevents the degradation of performance due to low acceptance thresholds, but does not preserve the benefits of DT if optically thick media are present in the problem. Additionally, distance calculation procedures must be implemented for every surface type in the problem.

Thus, we propose “HELL”, an alternative approach devised to surmount the performance constraints inherent in both ST and DT. This method enables either tracking method to be alternatively applied in different universes defining the geometry, thereby allowing the code user to tune the particle tracking performance. It provides a flexible and efficient solution in two key scenarios: (1) In complex geometries where ST is either unavailable or slow, and (2) In geometries that contain materials with large cross sections, such as localized heavy poisons, where the efficiency of DT is compromised.

3 Implementation of HELL in SCONE

The concept of “universe nesting” (or nested universes) is applied in SCONE[2]. Instead of a single-level approach, the geometry consists of multiple levels. This means that a particle can be present in multiple universes simultaneously, where each universe is a complete description of the geometry, that is, a decomposition of the entire space into disjoint cells which can be filled with a material or serve as a portal to another universe. This makes the geometry definition more user-friendly and computationally efficient, especially since a reactor physics geometry can be composed of tens of thousands of individual fuel pins.

In the multi-level geometry, the starting universe is called the root universe (Level 1). The transport operator is responsible for moving the particles from one collision location to another, performing tracking methods such as ST, DT, etc. It moves the particle through the geometry as follows: it pushes the particle forward by a distance; if a cell boundary is closer, the particle stops at the boundary; if a domain boundary is hit, boundary conditions are applied, and the particle is stopped. Note that since the cell boundaries do not need to align with the boundaries of the ‘portal’, the closest distance to the cell boundary needs to be calculated in each level the particle is in. Furthermore, since the co-ordinates of the particle in each universe may be different (due to translation and rotation when crossing a portal), all of them need to be updated when a particle is pushed forward.

[2] https://scone.readthedocs.io/en/latest/Geometry.html

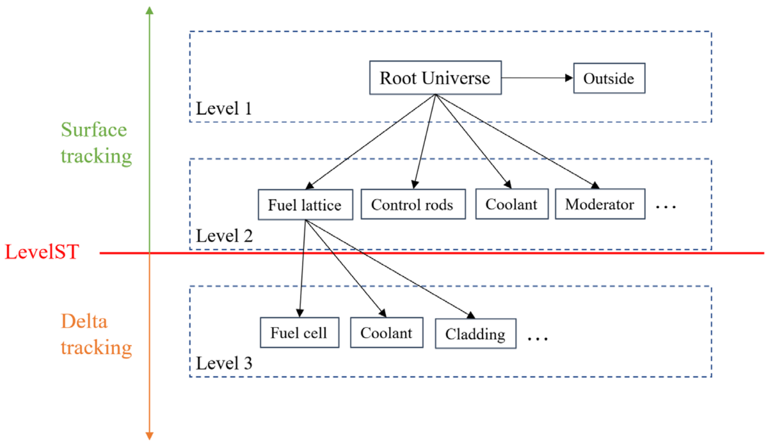

Figure 1: The scheme of HELL.

The principle of HELL is to apply ST and DT at different universe levels. An example scheme of HELL is illustrated in Figure 1. The user can nominate a "LevelST", which means that ST is applied between Level 1 to LevelST, and DT is applied to the universes above LevelST.

Based on the logic of this scheme, we created a new transport operator to facilitate the particle tracking loop (PTLoop). We named a logical term "pureST", which compares the nesting level and LevelST to (1) make a selection of ST using $\Sigma_{tot,m}(E)$ or DT using $\Sigma_{maj}(E)$, and (2) apply rejection sampling or not. Note that in the case when DT transition takes place, the distance to the closest boundary still needs to be calculated at the levels from 1 to LevelST. In fact, the normal ST procedure applies for these levels. The cell interfaces are ignored only deeper in the geometry. The HELL algorithm is outlined in Algorithm 1.

4 Application Calculations

In this section, we describe two simplified models to illustrate the performance of HELL. We performed the calculation using SCONE, with the JEFF-3.1.1 nuclear data library. The calculations presented in this section have been performed with 10^5 neutron histories, 50 inactive and 250 active cycles.

4.1 Case 1 - A core with involute surfaces

We selected a core with involute surfaces as Case 1 because calculating the surface distance in the involute geometry is computationally expensive. The model is based on the involute core at the Forschungs-Neutronenquelle Heinz Maier-Leibnitz (FRM II) high flux reactor in Germany [13]. The model refers to the geometry dimensions and material composition of the FRM II, which were studied previously [14–17].

A simplified model of the involute core is presented in Figure 2 and 3. The model consists of a single fuel element that comprises 113 involute-shaped fuel plates made of uranium silicide (U_3Si_2) with low-enriched uranium (LEU) at 4.8 g(U)/cc [14]. The active core region, as shown in region 6 of Figure 3, is 70 cm in height and has a cylinder radius between 6.5 and 11.45 cm [15]. To focus the tracking on the involute core region, a smaller diameter of heavy water reflector region is chosen in this model, with a diameter of 80 cm and a height of 120 cm. A single control rod follower made of beryllium is located in the center of the

Algorithm 1: HELL transfer simulation algorithm. ξ is used to denote a uniformly distributed random number on the interval [0, 1).

```
while True do
    pureST ← particle coordinates nesting <= LevelST
    if pureST then
        /* Obtain the local cross section                          */
        get Σ_tot,m(E)
        dist ← −log ξ/Σ_tot,m(E)
        /* Move particle stopping at surfaces in all levels, set
           'event' flag                                            */
        call geometry.move(dist)
        if event == collision then
            /* Exit if particle stopped at collision               */
            break
        end
    else
        /* Use majorant cross section                              */
        dist ← −log ξ/Σ_maj(E)
        /* Move particle stopping at surfaces in levels
           [1,LevelST], set 'event' flag                           */
        call geometry.move(dist, LevelST)
        /* Cycle the loop in case of surface hit                   */
        if event == surface crossing then
            continue
        end
        /* Obtain the local cross section                          */
        get Σ_tot,m(E)
        if ξ < Σ_tot,m(E)/Σ_maj(E) then
            /* Accept collision                                    */
            break
        end
    end
end
```

cylindrical fuel element. Additionally, a small piece of control rod (CR) made of a hafnium absorber is placed above it to evaluate the performance of different tracking methods without significantly changing the k_{eff}.

The performance of different tracking methods, namely ST, DT, HT, and HELL, has been evaluated in the involute core model. These assessments were conducted under two conditions: with and without the control rod. The default cut-off value (0.9) was used for hybrid-tracking (HT). The LevelST was set to 3, signifying a universe level that includes the control rod. HELL (LevelST=3) applies DT only to the involute lattice and ST to the remaining geometry regions. The results of these evaluations are presented in Table 1.

The values of k_{eff} are consistent across different tracking methods, indicating that HELL functions properly. It can be observed that the running time for DT with the CR is 1.49 times longer than that without the CR. This underscores the limitations of DT in the presence of a localized heavy absorber. HT outperforms DT when used with the CR and ST without the CR.

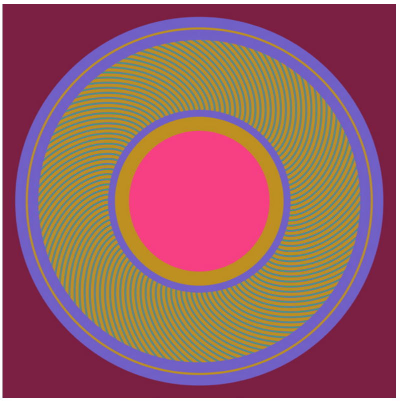

Figure 2: The radial cross section of the involute fuel.

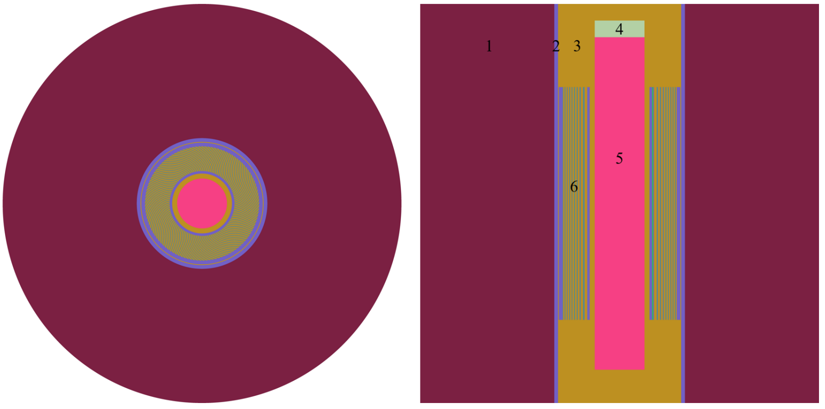

Figure 3: The horizontal (left) and vertical (right) cross section views of the involute core. (1 - heavy water, 2 - core tube, 3 - water, 4 - control rod, 5 - control rod follower, 6 - involute fuel)

However, it underperforms compared to HELL. The results indicate that HELL (LevelST=3) delivers the best performance. The execution time of HELL is 54.8% of that of DT when the CR is present, and 82.3% of that of ST when the CR is absent. This demonstrates that HELL enhances the tracking performance by addressing the constraints of DT with the CR and ST within the geometrically complex fuel region.

In situations where the CR is absent, the ST demonstrates a comparatively slower performance than the DT. However, we expect to see larger differences between the ST and DT. This is because the ST in the involute geometry requires expensive distance calculations with Newton iterations, as explained in detail in another accompanying paper [18]. A potential reason for this could be that while ST's efficiency diminishes in the involute fuel region, it operates effectively in areas with heavy water, which constitutes a large fraction of volume of this model.

To specifically assess the performance within the involute fuel region, we devised an extra calculation scenario. This scenario involves expanding the outer radius of the involute fuel to 22.9 cm and decreasing the diameter of the heavy water to 60 cm. The outcomes of this scenario can be found in Table 2. It can be observed that the running time for the ST is twice

Table 1: The running time for HELL compared to delta-tracking and surface-tracking (Case 1).

Scenario	Transport operator	k_{eff} (std)	Running time (s)	Ratio
With CR	ST	0.99215 (1.6e-04)	530	1
	DT	0.99207 (1.8e-04)	788	1.49
	HT	0.99219 (1.7e-04)	698	1.32
	HELL (LevelST=3)	0.99213 (1.7e-04)	432	0.82
No CR	ST	0.99277 (1.7e-04)	543	1.02
	DT	0.99220 (1.7e-04)	527	0.99
	HT	0.99241 (1.8e-04)	532	1.00
	HELL (LevelST=3)	0.99254 (1.9e-04)	447	0.84

Table 2: The running time for HELL compared to delta-tracking and surface-tracking (Case 1: Scenario with expanded fuel area and reduced coolant region).

Scenario	Transport operator	k_{eff} (std)	Running time (s)	Ratio
No CR	ST	0.99453 (1.8e-04)	573	1
	DT	0.99441 (1.8e-04)	283	0.49
	HT	0.99417 (1.8e-04)	264	0.46
	HELL (LevelST=3)	0.99448 (1.8e-04)	254	0.44

that of the DT, indicating that the performance of ST is suboptimal in the involute region. HELL demonstrates a considerable performance improvement, resulting in a running time that is 44.3% of ST's.

4.2 Case 2 - A PWR core

We selected a PWR model for the Case 2 because it has a large number of surfaces that require distance calculations. The core configuration and dimensions are derived from the original core model of the Hoogenboom-Martin benchmark, as described in [10]. We use a simplified model for illustrative purposes. The core is composed of 241 identical fuel assemblies, each measuring 21.42 × 21.42 cm. Each fuel assembly consists of 17 × 17 unit cells with dimensions of 1.26 × 1.26 cm.

The layout of the core model is displayed in Figure 4 and 5. For the purpose of particle tracking evaluation, the model includes only the fuel assemblies and the coolant. The model uses fresh UO_2 fuel at 4.95 wt% ^{235}U. The water coolant in the model has two different densities: 0.74 g/cm^3 below the core midplane to represent cold water, and 0.66 g/cm^3 above the core midplane to represent hot water. Boron is added to the coolant, with its concentration adjusted such that the reactor is near critical [10]. To evaluate the performance of HELL with DT, a control rod made of boron carbide is placed in the middle of the core.

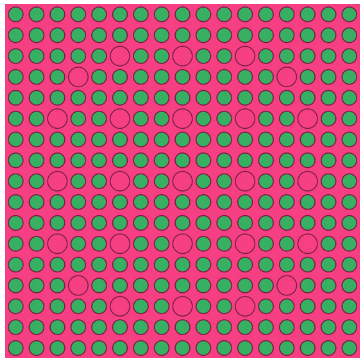

Figure 4: The horizontal cross section of the PWR fuel assembly.

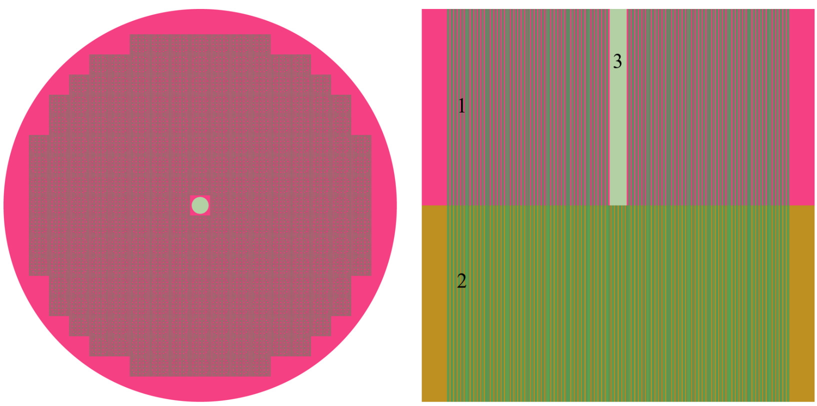

Figure 5: The horizontal (left) and vertical (right) cross section views of the PWR core. (1 - fuel assemblies with hot water, 2 - fuel assemblies with cold water, 3 - control rod)

We assessed the running time of the PWR core using ST, DT, HT, and HELL, for scenarios both with and without the CR. The results of these evaluations are presented in Table 3. DT with CR takes 2.83 times longer to run than DT without CR, indicating a significant difference. This result is consistent with the findings in Case 1.

The universe that includes the CR is present at Level 4. When HELL is configured to LevelST=4, it attains the minimum running time for CR scenarios, which is 37% of that of DT. HT also performs well, faster than DT and ST, but slower than HELL. For comparison purposes, HELL was also tested at LevelST=3. As expected, its running time was considerably longer than when HELL was set to LevelST=4. This indicates that by choosing an appropriate LevelST in HELL, we can significantly enhance the efficiency of the calculations.

In scenarios without control rods, DT and HT exhibit superior performance, attaining 85% of ST's running time. This underscores the DT's advantage over ST in reactor environments requiring crossing multiple surfaces. Given that a typical reactor involves multiple surfaces and control rods, HELL emerges as an effective choice to facilitate the integration of DT and ST for reactor modelling.

Table 3: The running time for HELL compared to delta-tracking and surface-tracking (Case 2).

Scenario	Transport operator	k_{eff} (std)	Running time (s)	Ratio
With CR	ST	1.07304 (1.3e-04)	295	1
	DT	1.07269 (1.3e-04)	690	2.34
	HT	1.07284 (1.3e-04)	278	0.94
	HELL (LevelST=4)	1.07263 (1.3e-04)	258	0.87
	HELL (LevelST=3)	1.07271 (1.4e-04)	660	2.24
No CR	ST	1.07286 (1.3e-04)	286	0.97
	DT	1.07268 (1.3e-04)	244	0.83
	HT	1.07305 (1.3e-04)	243	0.82
	HELL (LevelST=4)	1.07283 (1.3e-04)	279	0.95
	HELL (LevelST=3)	1.07302 (1.3e-04)	248	0.84

5 Conclusions

This paper presents our investigation into particle tracking within reactor physics simulations, specifically addressing the transport operator challenges. We examined two prevalent tracking methods: delta-tracking and surface-tracking. Delta-tracking, while effective for reactor modelling applications, struggles with efficiency when dealing with localized heavy absorbers. Surface-tracking, on the other hand, faces issues of inefficiency and impracticality due to high computational costs associated with computing distance to exotic surfaces.

To overcome these limitations, we introduced a novel method called HELL and implemented it into the SCONE Monte Carlo code. This method strikes a balance between the strengths and weaknesses of both delta- and surface-tracking methods, by splitting the geometry into ST and DT tracking regions. These regions are delimited by the nesting level in the geometry graph, allowing the user to separate the regions with localised heavy absorbers from DT, or switching to DT only in zones exhibiting large geometrical complexity.

We assessed the performance of HELL in two distinct reactor physics applications. The findings indicate that HELL optimizes the particle tracking process, leading to a substantial reduction in running time. Demonstrating robust performance, HELL emerges as a practical and adaptable solution that holds promise for a broad range of particle tracking simulation applications.

Acknowledgements

The research presented in this paper has been partially supported by the EPSRC under the project “Mathematical Theory of Radiation Transport: Nuclear Technology Frontiers (MaTh-Rad)”, grant agreement number EP/W026899/1.

References

[1] J. Leppänen, Annals of Nuclear Energy **37**, 715 (2010)
[2] L. Morgan, D. Kotlyar, Annals of Nuclear Energy **85**, 1184 (2015)
[3] E. Woodcock, T. Murphy, P. Hemmings, S. Longworth, *Techniques used in the GEM code for Monte Carlo neutronics calculations in reactors and other systems of complex geometry*, in *Proc. Conf. Applications of Computing Methods to Reactor Problems* (Argonne National Laboratory, 1965), Vol. 557
[4] J. Leppänen, Annals of Nuclear Energy **105**, 161 (2017)

[5] J. Rehak, L. Kerby, M. DeHart, R. Slaybaugh, Nuclear Engineering and Design **342**, 231 (2019)
[6] S.D. Richards, C.M. Baker, A.J. Bird, P. Cowan, N. Davies, G.P. Dobson, T.C. Fry, A. Kyrieleis, P.N. Smith, *MONK and MCBEND: Current status and recent developments*, in *Proc. SNA+MC 2013* (EDP Sciences, 2014), p. 06013
[7] D. Long, S. Richards, P.N. Smith, C. Baker, A. Bird, N. Davies, G. Dobson, T. Fry, D. Hanlon, R. Perry et al., Tech. rep., American Nuclear Society-ANS (2015)
[8] Q. Guo, Z. Chen, Sustainability **10** (2018)
[9] M.A. Kowalski, P. Cosgrove, J. Broman, E. Shwageraus, Journal of Nuclear Engineering **2**, 57 (2021)
[10] J.E. Hoogenboom, W.R. Martin, B. Petrovic et al., *The Monte Carlo performance benchmark test-aims, specifications and first results*, in *Proc. M&C 2011* (2011), Vol. 2, p. 15
[11] J. Leppänen et al., Ph.D. thesis, VTT Technical Research Centre of Finland (2007)
[12] W. Coleman, Nuclear science and engineering **32**, 76 (1968)
[13] H. Breitkreutz, Ph.D. thesis, Technische Universität München (2011)
[14] A. Glaser, Ph.D. thesis, Technische Universität München (2005)
[15] C. Reiter, H. Breitkreutz, A. Röhrmoser, A. Seubert, W. Petry, Annals of Nuclear Energy **131**, 1 (2019)
[16] A. Röhrmoser, Nuclear engineering and design **240**, 1417 (2010)
[17] C. Reiter, A. Bergeron, D. Bonete-Wiese, M. Kirst, J. Mercz, R. Schönecker, K. Shehu, B. Ozar, F. Puig, J. Licht et al., Annals of Nuclear Energy **183**, 109599 (2023)
[18] M.A. Kowalski, C. Ding, E. Shwageraus, *Particle Tracking in Involute-based Reactor Geometries in SCONE*, in *Proc. PHYSOR 2024* (2024)