7th International Conference on Modelling and Simulation in Nuclear Science and Engineering (7ICMSNSE)
Ottawa Marriott Hotel, Ottawa, Ontario, Canada, October 18-21, 2015

7ICMSNSE-049

Static Benchmarks for G4-STORK, a Time-Evolution Monte Carlo Code for the Simulation of Multiplying Media.

A. Buijs, W. Ford and A. Tan
McMaster University, Hamilton, Ontario, Canada
buijsa@McMaster.ca

Abstract

The GEANT-4 toolkit was used to develop G4-STORK, which allows one to study the time-development of a neutron population in a multiplying media such as a nuclear reactor core. The code has a number of additional attractive features such as on-the-fly Doppler broadening, modification of material properties during a run, "real-time" treatment of delayed neutrons, 3-D representation of the geometry including particle tracks, and the ability to track particles in a magnetic field. The GEANT-4 toolkit has parallel processing options and the capability of importing and exporting geometries to CAD programs. This paper gives the current status of the G4-STORK project, including static applications to a SLOWPOKE and the SCWR conceptual design.

Keywords: Reactor core physics, Neutronics, Monte Carlo, Code Validation.

1. Introduction

1.1 Time-dependent Monte Carlo Codes

A conventional nuclear reactor periodically makes minor adjustments to the control devices to maintain a stable power level. For most analyses simulating this reactor behaviour, these time-dependent effects are small and can be ignored (i.e. static analysis). However, in situations where the state of the reactor is changing rapidly, such as power manoeuvres or loss-of-control accidents, the time-dependent response of the reactor is large and important. For these situations, time-dependent reactor codes are used to analyse and predict the behaviour of the reactor.

Traditionally, these time-dependent responses, also known as reactor kinetics, have been studied using deterministic approximations to the time-dependent neutron transport equation, such as the point kinetics approximation. Further research in this area has led to a quasi-static correction for spatial dependencies that can be applied to the point kinetics approximation, as well as various space-energy dynamics approaches. These other dynamics approaches involve decomposing the reactor response using a modal, nodal or finite difference approach[1]. Being deterministic approaches, they are limited by the same factors as static deterministic codes: the discretizations of space, energy and angle that are necessary to solve the neutron transport equation directly. Additionally, the time-dependent behaviour adds significant complexity compared to static solutions, so these approaches generally require longer computation times.The limitations of the discretizations may be avoided by employing a stochastic (Monte Carlo) approach. Rather than solving a simplified version of the transport equation, the Monte Carlo method guides the neutrons through the simulated geometry according to the relative probabilities and outcomes of various neutron-nucleus interactions. Since each neutron is tracked in both space and time continuously, the

Monte Carlo approach is intrinsically time dependent. Thus, the characteristics and time-dependent responses of a reactor can be calculated by averaging the behaviour of sufficiently many neutrons over the duration of the simulation.

Some current Monte Carlo reactor physics codes, such as MCNP5, eschew true time-dependent simulations for the calculation of reactor characteristics like reactivity. Instead, they simulate successive neutron fission generations, where each step consists of tracking the neutrons produced from the fissions at the end of the previous generation until they are lost. This is a time-independent calculation because each neutron has a unique lifetime within the generation, which may be greater or less than the average duration of the generation. Consequently, these calculations do not allow for accurate simulations of the reactor response to transient reactor conditions (e.g. material temperature changes, etc.), nor do they accurately simulate inherently time-dependent systems (e.g. sub- or supercritical systems). These types of calculations may still be used for most reactor analysis that is concerned with static, critical reactors, but they are insufficient for reactor kinetics.

2. The G4-STORK Code

To overcome these limitations and aided by advancements in computing power, new reactor physics codes are being developedthat use time-dependent Monte Carlo simulations ([2][3] [4]). In particular, the paper by Sjenitzer and Hoogenboom[2] presentsa Monte Carlo reactor physics approach that includes delayedneutrons, making it a viable code for reactor dynamics simulations. Rather than developing anentirely new reactor physics code, our approach utilizes the Geant4Monte Carlo toolkit to provide the background physics models,particle tracking and basic simulation control processes. Geant4 was released by the Geant4Consortium in 1998[5], whose members include CERN (EuropeanOrganization for Nuclear Research) and SLAC (Stanford LinearAccelerator), and has been used in high-energy particle physics,medical physics, space engineering and accelerator design. Thus, our code benefits from expertiseand experience that has gone into Geant4.Using Geant4 as a framework, we developed G4-STORK (Geant4STochastic Reactor Kinetics), a stochastic, time-dependent reactorphysics code. While Geant4 is normally used to simulate discreteevents, such as a particle impinging on a detector, the toolkitwas designed to be adaptable and extensible. By adding our owncode to the base toolkit to create G4-STORK, we can accurately model time-dependent neutron populations in a nuclear reactor,

Even in very sub- or supercritical systems, G4-STORK can model the response of the neutron population to dynamic reactor conditions. A description of the G4-STORK code is given in Ref. 6.

2.1 Geant4

The Geant4 toolkit is an open-source collection of object-orientedC++ code that acts as a framework for Monte Carlo particlephysics simulations. The toolkit itself isnot a program and cannot directly be used to simulate a physicsproblem. Instead, the toolkit provides libraries of functions andclasses that the end-user can compile alongside their own codeto create a unique simulation program. This architecture requiresmore work from the user than traditional nuclear physics codes,as well as an understanding of modern C++ conventions, but it alsoprovides the user with greater flexibility in terms of implementation.

The toolkit provides the basic particle tracking and interactions,including physics models for hadronic collisions,electromagnetic interactions (including photons), as well as the basic simulation management (e.g. the background processes thatcontrol the whole simulation and facilitate

communicationbetween its various appendages). However, the end-user isrequired to add the necessary code to apply the basic functionalityof the toolkit to a specific application.

The end-user is required to add C++ code defining the following:

1. Simulation geometry and material composition.
2. State of initial primary particles.
3. List of applicable physics models.

Most of this code is written using the classes and functions fromthe toolkit, and once it is compiled, it defines a unique simulation.

Besides these mandatory additions, most classes in the toolkit maybe modified and/or overridden using user-added code (polymorphism). This includes the physics models, particle definitions and simulation management processes. Moreover, any part of the Geant4 toolkit can be modifiedby changing the source code of the toolkit itself; however, this should be a last resort since any changes to the source code coulddamage the interoperability of the user's simulation on othercomputers.

The user is also required to implement data collection from the simulation using accessor classes in the toolkit. Unlike most othercodes where the available data are limited by the developers,Geant4 allows end-users to retrieve any data parameter in the simulation, although some are more accessible than others. For example,end-users may sample the exact characteristics (location, time,energy, etc.) of any neutron interaction, the history of the neutronsince birth, or the relative likelihood of any interaction type (scattering, fission, etc.) occurring for a given neutron. This freedom allows end-users to adapt Geant4 to purposes beyond the goals/needs of the original developers.

2.2 The G4-STORK Model

Starting from an initial guess of the equilibrium neutron position and energy distributions, the G4-STORK code works by tracking individual neutrons through a geometry provided by the user in steps of time [2]. At each time step, the important parameters such as k_{eff} and Shannon entropy (see Ref. 6) are determined and the neutron population is renormalized to the initial number of neutrons. These important parameters are not recorded into the final results until the spatial distribution of the neutrons has converged. The neutron distribution is said to be converged when the Shannon entropies of each of the last 25 time steps do not deviate from the mean Shannon entropy (taken from the last same 25 time steps) beyond a set limit. Thus, the closer the initial guess is to the actual final distribution, the faster the neutron population will converge. Since it is a stochastic simulation, the processes that the neutrons undergo as they move throughout the geometry are randomly selected from a list of potential processes that are weighted based on the probability of their occurrence. This is dependent on the isotopic composition of the material that the neutron is currently traversing and the kinetic energy of the neutron relative to the nuclei in its path. While the composition of the material is defined in the geometry by the user, the relative kinetic energy of the neutron to the nuclei is determined by the on-the-fly Doppler broadening algorithm described in the next section.

2.3 On-the-Fly Doppler Broadening

One of the features of G4-STORK is its ability to modify its geometry as the simulation proceeds. For example, the temperature of materials may change based on the energy released in the fission process.

The reaction cross sections in turn depend on the temperature, in particular for resonances which become broader as the temperature increases. A visualization of what the cross-section data looks like after being broadened can be seen below in Fig. 1 for the Pu-239 fission cross section.

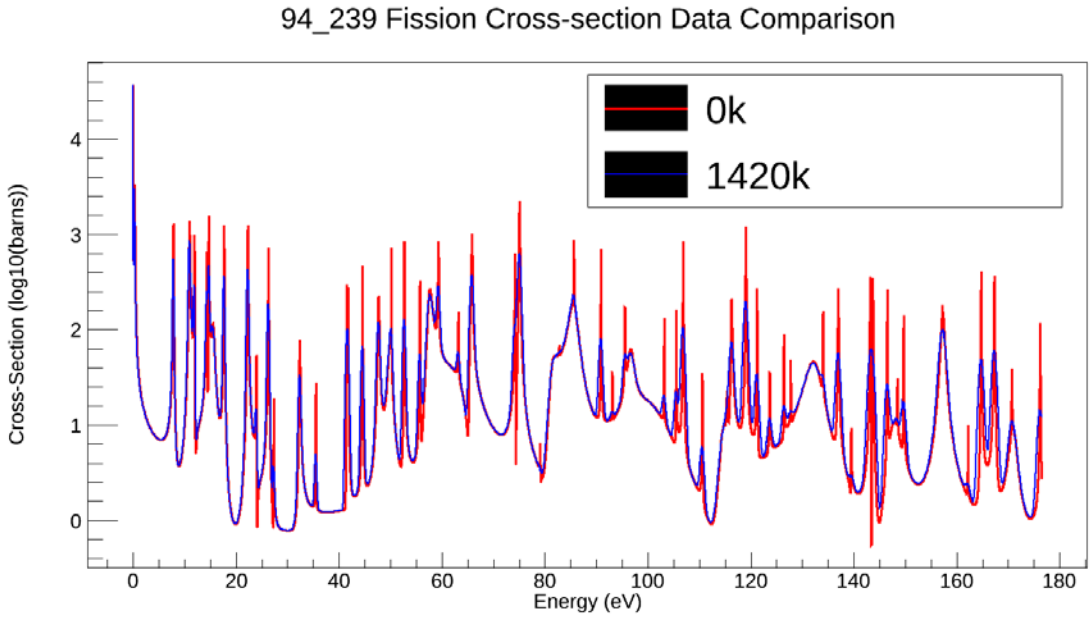


Figure 1: Cross-section data files of Pu-239 at 0K and at 1420K for the fission process.

In this graph the broadened data appears flatter and smoother than the unbroadened data, as expected. In order to avoid having to provide nuclear data at a multitude of temperatures, a method called on-the-fly Doppler Broadening is applied. The on-the-fly Doppler broadening algorithm basically generates a velocity for a nucleus in the material and calculates the energy of the neutron in the rest-frame of the nucleus. An iterative procedure is needed to account for different isotopes in the material. This is a very time-consuming process which is exacerbated if the temperatures used in the user defined reactor geometry are high (since more samples will have to be taken to cover the wider Gaussian distribution).

2.4 Delayed Neutrons

For the proper treatment of delayed neutrons, G4-STORK tracks the size of six precursor groups[1]. The removal and addition of precursors occurs between runs and is dictated by the following equations:

$$\frac{dC_i}{dt} = -\lambda_i C_i + \gamma_i R(x,t),$$

where C_i is our precursor population of group i, λ_i the decay constants, γ_i fission yield and $R(x,t)$ is the fission rate of U-235 in units of fissions/s/cm^3. To generate an initial population we assume a steady-state condition, then we integrate over time in space. This means using the total number of fissions and the timeduration of a previous simulation.

[1]Six groups were chosen as this has traditionally described the CANDU reactor well.

$$C_i = \frac{\gamma_i}{\lambda_i T} n_f$$

where$T$ is the time duration and n_f is the number of fissions.Duringasimulationthenumberofprecursorsaddedissimplytheproductofthe number of fissions in the run with the fission yield:

$$C_i += \gamma_i n_f^{run}$$

Russian roulette is used to remove precursors:

$$T_{decay} = \frac{-\ln r}{\lambda}$$

Where$r$isarandomnumberintheinterval[0,1]. Therouletteisrunforeveryprecursor and if the decay time T_{decay} is less than the upcoming run duration a delayed neutron is added to the survivor list and a precursor is removed.

In order to maintain the correct normalization between runs, both the non-decayed precursors and the delayed neutrons need to be subjected to the same combing algorithm as the prompt neutrons.

2.5 Additional Features

As the GEANT4 code is very versatile, it allows for the tracking of any other particles one might want to consider as well.For our purposes, these would be electrons and photons. The photons are of importance for the delayed photo neutrons and contribute to the heat deposition outside of the fuel. Also, electrons and photons are useful for understanding reactor physics instrumentation.

Lastly, GEANT4 allows for the tracking of charged particles in magnetic fields of any strength and shape.This opens up the possibility to use the G4-STORK code described here for particle transport in plasmas, which is of interest for magnetic confinement fusion, especially where the motion of charged particles in the magnetic field affects the magnetic field itself.

3. Benchmarking the G4-STORK Code

In the following, we will introduce two benchmarking exercises with the G4-STORK code: i) the simulation of a lattice cell of the Super-Critical Water Reactor (SCWR)[7], which is a conceptual design of CNL (Canadian Nuclear Laboratories) in collaboration with, among others, McMaster University; and ii) a full-core simulation of a SLOWPOKE reactor.Both simulations are still in the phase of static calculations, and the results are preliminary.

3.1 A Benchmark Simulation of an SCWR Lattice Cell.

The G4-STORK code allows for the usual boundary conditions[6]. Alattice cell of the SCWR core is simulated using periodic boundary conditions. Since the lattice cell is axially symmetric, only a quarter of the lattice was simulated in G4-STORK. Visualizations of the quarter lattice cell geometry used in

G4-STORK can be seen in Fig. 2, where the sides of the drawing are 12.5 cm each (25 cm lattice spacing).

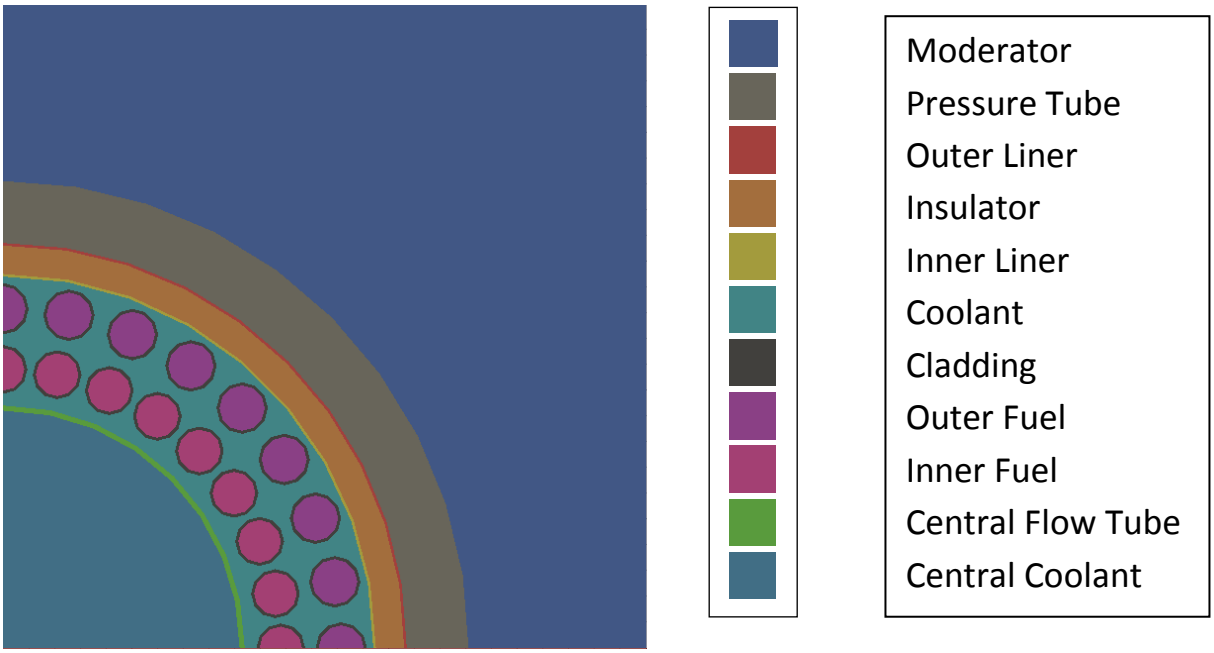
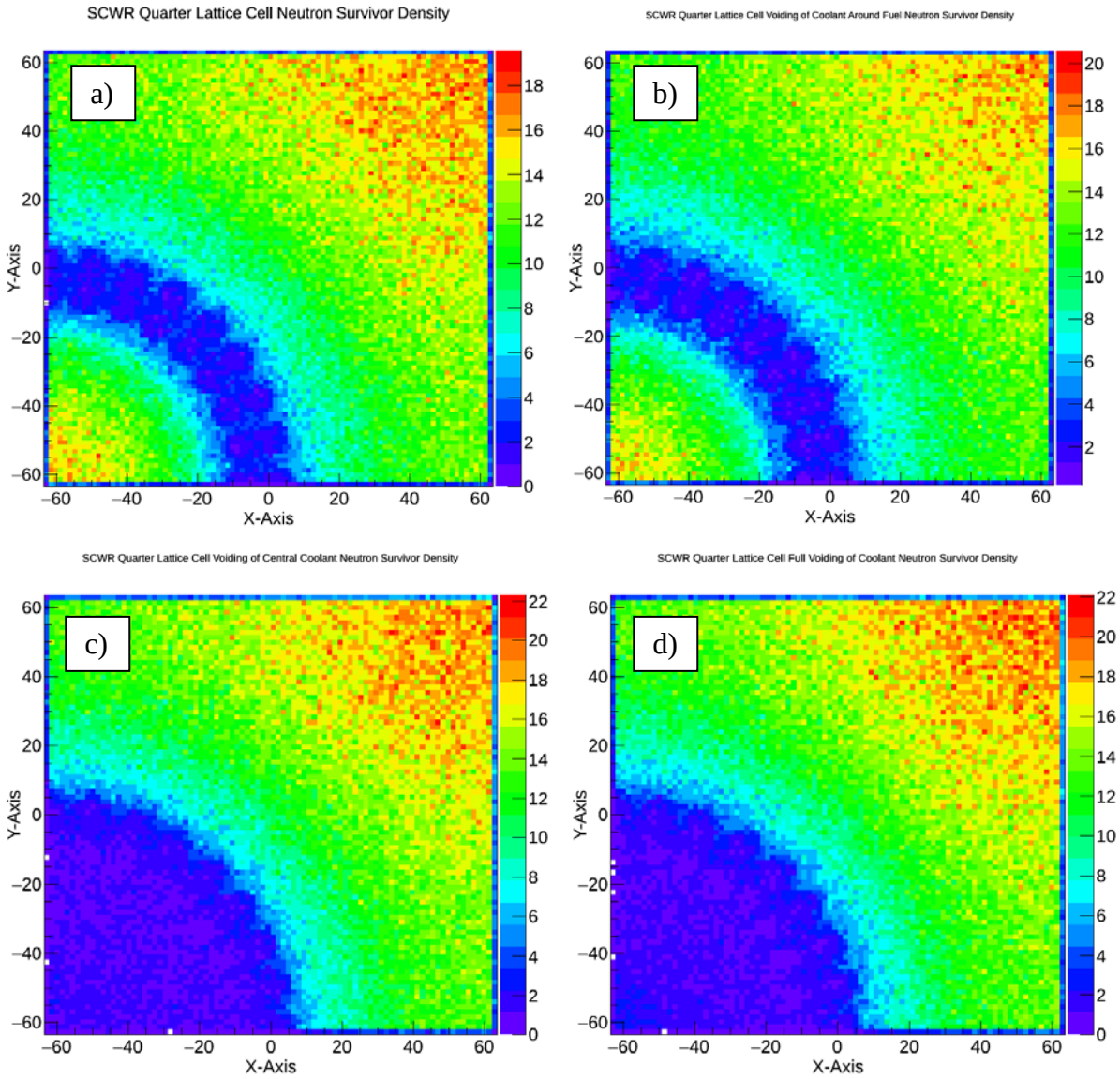


Figure 2: A color-coded version of the geometry used in the quarter lattice cell simulation.

The distribution of neutrons in the SCWR quarter lattice cell for the reference, fuel coolant voiding, central coolant voiding and full coolant voiding cases, as predicted by G4-STORK, can be seen in Fig. 3 (a-d). As expected, the neutron density is lowest near the fuel because of the high absorption cross-section of the isotopes present, and highest in the moderator regions furthest from the fuel for similar reasons.Little difference in the neutron distribution can be seen by comparing Figs.3a) and b), or by comparing Figs. 3c) and d) indicating that the voiding of the coolant surrounding the fuel has little effect on the shape of the neutron distribution. This is expected since the neutron density in the region around the fuel is already very low due to the high absorptivity of the fuel and thus the perturbation of the coolant density in this region is unlikely to have a large effect on the shape of the neutron distribution. Also since there are equal amounts of the coolant around the fuel we do not expect it to shift the neutron distribution in either radial direction. However if we compare Figs. 3a) and c) or Figs. 3b) and d), in the latter cases the neutron distribution in the center of the lattice cell can be seen to dip when the coolant is voided. This is expected since the neutrons in those areas are no longer being moderated and are instead being reflected back to the fuel at high energies, resulting in either the neutrons being captured by the fuel isotopes or in them flying out to the moderator.



Figures 3: 2D Neutron density histograms calculated from the reference (a), fuel coolant voiding (b), central coolant voiding (c) and full coolant voiding (d) cases of the SCWR quarter lattice cell simulation.

The distribution of fission sites for the SCWR quarter lattice cell reference, fuel coolant voiding, central coolant voiding and full coolant voiding cases, as predicted by G4-STORK can be seen below in Figs. 4a)-d). The inner fuel pins are composed of a larger percentage of plutonium than the outer fuel pins which is why they have a significantly higher density of fission events in Figs.4a) and b) even though the neutron density is not significantly different between the outer fuel and the inner fuel in these two cases as shown by Figs 4a) and b). The neutron density is determined by the absorption of the fuel, which is complete for either fuel type. Gradients in the density of fission events can be seen in all cases of Fig. 4, with clear maxima occurring along the surface of each fuel rod that is facing moderator or coolant and is not facing other fuel rods. This will be due to the higher neutron density in these areas and because the neutrons coming from these direction have been moderated and are more likely to cause fission.

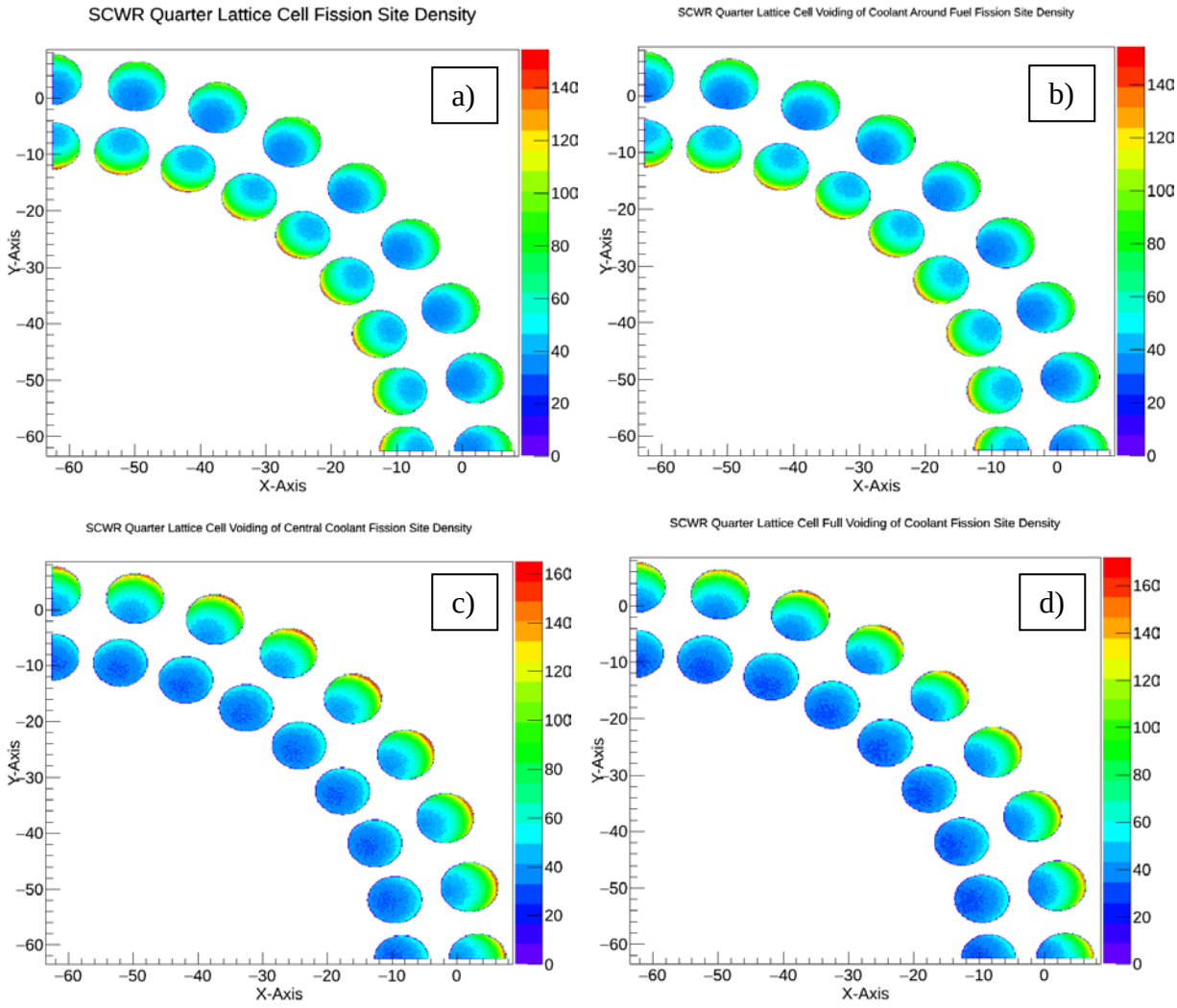


Figure 4: Fission site density 2D histograms calculated from the reference (a), fuel coolant voiding (b), central coolant voiding (c) and full coolant voiding (d) cases of the SCWR quarter lattice cell simulation.

The effective multiplication constantk_{eff} of the SCWR quarter lattice cell for the reference, fuel coolant voiding, central coolant voiding and full coolant voiding cases can be seen below in Table 1. The k_{eff} received from the cooled and voided quarter core simulation was 1.253±0.001 and 1.215±0.002 respectively resulting in a CVR value of -25 mk.[2]These results differ from those achieved with MCNP, which were1.2914 for the cooled lattice cell and 1.2687 for the voided case resulting in a CVR of -14mk.Part of this discrepancy is due to the different data libraries used and the mixing of high and low temperature data libraries to achieve right temperatures in MCNP. Also, further investigation into the methodology used by both codes will likely determine what is causing the difference. Although total voiding of the coolant causes a negative CVR, voiding around just the fuel rods causes a slight increase in the reactivity of the reactor. This effect would exacerbate hotspots occurring along the sheath, caused by local voiding.

[2]CVR is defined as $1/k_{\text{eff}}$(cooled) – $1/k_{\text{eff}}$(voided)

Cases	Coolant around the fuel (k_{eff})	CVR (with respect to case a) In mk
a) Withcoolant	1.253±0.001	
b) No central coolant	1.206±0.001	-31
c) No coolant around fuel	1.258±0.001	+ 3
d) No coolant at all	1.215±0.002	- 25

Table 1: The k_{eff} of the SCWR quarter lattice cell for the reference, fuel coolant voiding, central coolant voiding and full coolant voiding cases, including CVR with respect to reference case.

3.2 Modelling a SLOWPOKE Reactor

The SLOWPOKE reactor is a small, pool-type reactor with a nominal power of 20 kW[8]. It is fuelled with low-enriched uranium (LEU, 19.9 wt%) and featuresa single control rod, centered in the core. With a core that measures 22.1 cm in diameter and 22.7 cm in depth, the SLOWPOKEis neutronicallytightly coupled and is thus a prime candidate for validating G4-STORK in its ability to simulate transient modes of reactor behaviour.However, the results of this paper are limited to static calculations.

Fig. 5 shows the G4-STORK implementation of the SLOWPOKE geometry. The geometrical description and the material composition are from Ref. 9.

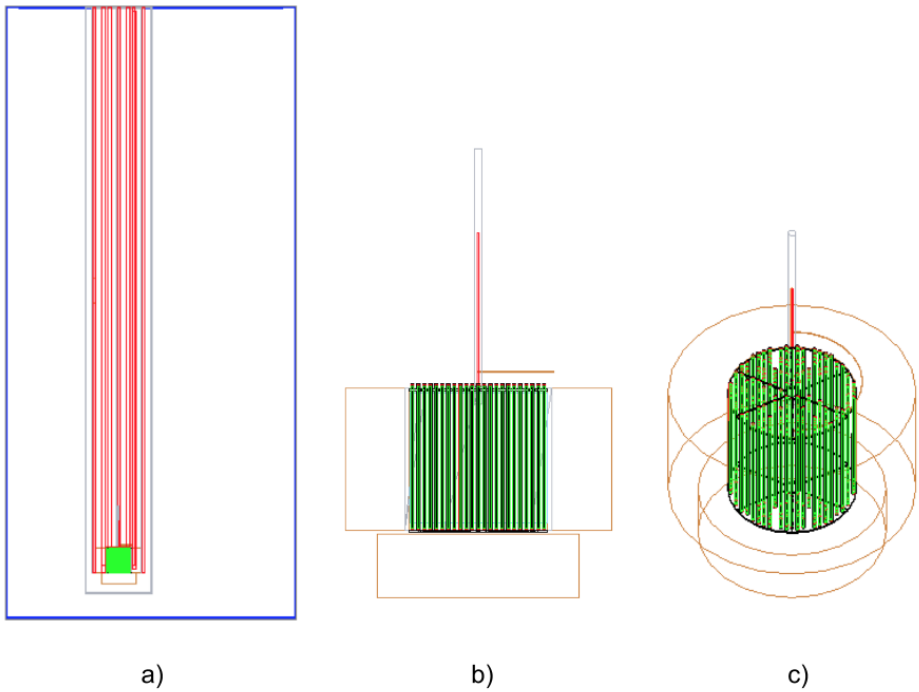


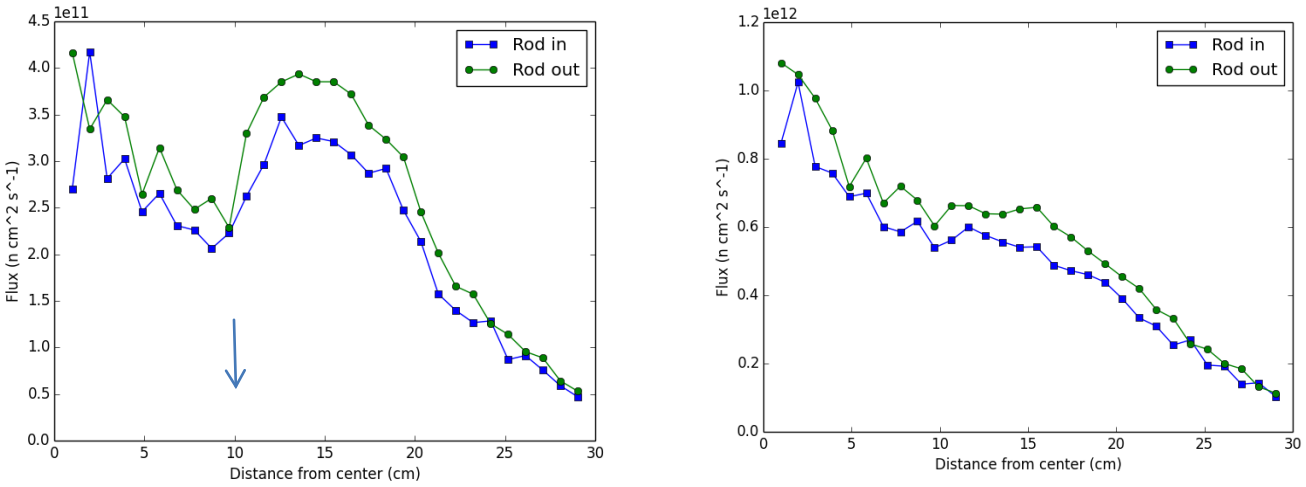
Figure 5: The SLOWPOKE model in G4-STORK, a) the whole simulation world, b) side view of reactor core, c) 3-D view of reactor core.

3.3 Control Rod Worth

To calculate the reactivity worth of the central control rod of the SLOWPOKE, two simulations were run with the same geometry except for the control rod placements.An initial point source was used, placed in the center of the reactor core using a Gaussian distribution of energies centered around 2.0 MeV. The simulations used 20000 initial neutrons, which were followed for 2ms, sufficient to achieve convergence in the distribution. The control rod was moved out by 20.66 cm and the effective multiplication factor was used to calculate reactivity, which was found to be 4.9 ± 2.0 mk.This is to be compared to the value of 5.45 mk measured at the SLOWPOKE-2 of RMC [9].

3.4 Neutron Flux

The SLOWPOKE reactor has been shown to provide a relatively high neutron flux, on the order of 10^{12} n/cm^2/s. From Fig. 6, the simulated fluxes are seen to be of the same order of magnitude, because the same number of initial neutrons was used.As we would expect, the amplitude of the flux is reduced when the control rod is fully inserted with the flux shape remaining relatively intact.This means that our perturbation mainly affects the overall reactivity of the system and supports the use of the SLOWPOKE reactor as a favourable reactor for simulation.



a) b)

Figure6: Neutron flux as a function of radius in the SLOWPOKE reactor, for thermal neutrons (0 – 0.004 eV, a)) and the cadmium absorbing region (0 – 0.40 eV, b)). The arrow indicates the edge of the fuel.

The fuel rods of the SLOWPOKE extend out to a radius of roughly 10 cm and this is demonstrated in the flux of the thermal energy region as it increases quickly in the reflector after 10 cm before dropping off in the outer regions of the reactor.

The neutron fluxes are plotted in Fig. 6 at 20kW of power and for the thermal neutron energies (0.00 - 0.04 eV) as well as the cadmium absorbing energies (0.00 - 0.40 eV).It should be noted that the actual flux for the rod-in case is zero since the reactor is not at power.

4. Discussion and Conclusions

The results produced by G4-STORK for the SCWR, in particular the various CVR values are significantly different from those produced by other simulations. An in-depth analysis of the physics methodology used by G4-STORK is currently being done in order to understand where the discrepancy is coming from and whether it is erroneous. In order to examine how the different data libraries used by G4-STORK and MCNP, affect the results of each code, the MCNP libraries will be converted into the G4NDL format used by GEANT4, so that both codes will be using the same library. To perform this conversion, the undocumented final state libraries of GEANT4 have been analyzed and documented, but the software necessary to convert between the two data sets is currently still under development. More work will also have to be done to finish modifying G4-STORK to use the Doppler broadened data so that simulations can be carried out in a more reasonable time frame.

It is evident from the simulated control rod worth of 4.9 ± 2.0mk compared to the experimental measured worth of 5.45mkand the agreement of neutron flux magnitude on the order of 10^{12} n/cm2/s that the Monte Carlo methods applied in G4-STORK provide an acceptable representation of the SLOWPOKE-2 reactor. Additional simulated experiments are set to be performed.Specifically,the temperature feedback observed during a rod movement in the SLOWPOKE will be modelled as the rod is withdrawn.Further work involves developingand testing methods to allow the code to simulate this transient response.

5. References

1. Ott, K.O., Neuhold, R.J., 1985. Introductory Nuclear Reactor Dynamics. AmericanNuclear Society.
2. Sjenitzer, B.L., Hoogenboom, J.E., 2011. A montecarlo method for calculation on the dynamic behaviour of nuclear reactors. Prog.Nucl. Sci. Technol. 2, 716–721.
3. Shayesteh, M., Shahriari, M., 2009. Calculation of time-dependent neutronic parameters using montecarlo method. Annal.Nucl. Energy 36 (7), 901–909
4. Yun, S., Kim, J.W., Cho, N.Z., 2008. Monte carlo space-time reactor kinetics method and its verification with time-dependent sn method. In: Proceedings of the International Conference on Reactor Physics 2008, 2008
5. Agostinelli, S. et al., 2003. Geant4 - a simulation toolkit. Nucl. Instrum. Meth. Phys. Res. Sect., A 506 (3), 250–303.
6. Russell, L, Buijs, A. and Jonkmans, G., Annals of Nuclear Energy 71 (2014) 451–461.
7. J. Pencer, D. Watts, A. Colton, X. Wang, L. Blomeley, V. Anghel, and S. Yue, Coreneutronics for the Canadian SCWR conceptual design," in The 6th InternationalSymposium on SCWR (ISSCWR-6), (Shenzhen, Guandong, China), 2013.
8. S. Yue, S. Polugari, V. Anghel, J. Hilborn, B. Sur, A New Safety Principle Forthe SLOWPOKE-2 Reactor. (2013), 113-114.
9. J.R.M Pierre, Low Enrichment Fueled Slowpoke-2 Nuclear Reactor Simulationwith the Monte Carlo based MCNP 4A code. (1996), 54-62.