

MCkeff: An Indigenous Monte Carlo Code for estimation of Neutron Multiplication factor of Fissile Systems

Sachin Shet^{1*}, Kapil Deo Singh², S. Anand^{2,3} and K.V. Subbaiah¹

¹*Manipal Centre for Natural Sciences, Centre of excellence,*

Manipal Academy of Higher Education (MAHE), Manipal, Karnataka-576104.

²*Health Physics Division, Bhabha Atomic Research Centre, Trombay, Mumbai-400085.*

³*Homi Bhabha National Institute – BARC, Anushaktinagar, Mumbai-400094*

*shet.sachin@manipal.edu

Abstract

An Indigenous Monte Carlo code, MCkeff, for solving the steady-state neutron transport equation is currently under development at the Manipal Centre for Natural Sciences to estimate the neutron multiplication factor (k_{eff}) of systems containing fissile materials such as U-235 and Pu-239. Any arbitrary configuration of materials contained in cells can be described by its bounding signed analytical linear or quadratic surfaces (half-spaces) combined with Boolean operators of intersection and union. In the present code, the events simulated are elastic scattering, fission, inelastic scattering, (n, xn) and several other absorption reactions (n, c) given in any particular ENDF evaluation. For reactions which results with one or more secondary neutrons, energy and angles are sampled as per the laws prescribed in the ENDF 6 file. Special treatment is employed in the code for thermal neutrons (<4eV) using $S(a, \beta)$ cross-sections (for light nuclides) and for intermediate energy neutrons (few keV to tens of keV) in heavy elements to account for self-shielding effects on cross-sections using probability tables of cross-sections. At the end of the simulation, the neutron multiplication factor (k_{eff}) of a system is estimated as a collision estimator ($k_{\text{eff}}^{\text{col}}$), absorption estimator ($k_{\text{eff}}^{\text{abs}}$), track length estimator ($k_{\text{eff}}^{\text{tra}}$) and combined estimator ($k_{\text{eff}}^{\text{com}}$), which is deduced from the former 3 estimators along with their statistical uncertainties. Several new algorithms are incorporated, such as binary search and logarithmic energy grid to optimize the computer time, with dynamic memory allocation, which saves a lot of computer memory. Geometry viewer for the code comes with the new feature of identifying the void and overlapping regions of the user geometry model of the problem.

Keywords: Monte Carlo, Fissile systems, criticality, steady-state transport equation, eigenvalues, neutron multiplication factor, k_{eff} .

1. Introduction

U-235 and Pu-239 isotopes of uranium and plutonium are called fissile materials undergoing thermal neutron-induced fission. During the fission process, these isotopes release two or more neutrons and a large amount of energy. These neutrons, while moving in the system, may interact either with fissile atoms causing further fissions or non-fission capture or get absorbed in the surrounding material as radiative capture. This may lead to a critical, subcritical (convergent) or supercritical (divergent) system.

Safety in the Nuclear industry is of utmost importance as it involves handling large quantities of fissile isotopes of uranium and plutonium. Some places where a large amount of fissile material is involved are fuel storage, fuel reprocessing, fuel fabrication, etc. The crucial parameter in handling fissile materials is the neutron multiplication factor (k_{eff}) of the system. A precise estimate of this parameter is essential to avoid criticality accidents [1]. Monte Carlo Codes are preferred due to flexibility in handling complicated geometrical arrangements of materials, ease of incorporating detailed physical processes, and simplicity in estimating errors associated with the results. The theory behind this method is described in detail in the references [2-4]. Several Monte Carlo codes [5-8] exists to estimate the k_{eff} of a system. An overview of general-purpose Monte Carlo radiation transport codes existing in the ORNL/RSICC library is published by Kirk [9]. It is known that the Monte Carlo methods consume a lot of computer resources and computation (CPU) time to get the result of a problem with a desirable precision. This prohibits the usage of such previously available codes for computation as they are obsolete due to sea changes in the hardware and software of the computer industry besides the development of new algorithms. A new Monte Carlo code MCkeff has been written from scratch, keeping in mind the advances made over time.

Further, Monte Carlo methods are inherently suitable for parallel computers, which reduces CPU time significantly. The MCkeff code developed at present uses Parallel computing software libraries [10]. The amount of random-access memory (RAM) requirement varies as per the size of the input problem. The code has been developed with dynamic memory allocation, which saves a lot of RAM. MCkeff is a flexible modular code, and it is easy to adapt to the later changes that are taking place in the computer industry and the advances made in physics models and the computing algorithms of the method. MCkeff uses the continuous energy point-wise A-Compact-ENDF (ACE) [11] formatted cross-section data similar to that of OpenMC [7] and other Monte Carlo codes. MCkeff developed as a cross-platform code for Windows and Linux based operating systems with GNU Fortran compiler [12]. The code also has been compiled using the Intel Fortran compiler [13]. The code has been developed in a modular form using the latest Fortran 2008 features [14]. For geometry visualization in 2D, a module was developed using a graphics plotting library package DISLIN [15], which is compatible with the Fortran programming language.

The salient features of the MCkeff code are:

- Written from scratch incorporating FORTRAN 2008 features for Windows and Linux Operating systems.
- The input file is a text file. The structure and format followed are the same as that of other Monte Carlo codes.
- Input checking is done extensively to detect all possible errors, and messages of fatal errors and warnings are displayed on the screen.
- It allows building a complex configuration of materials in geometric regions (cells) bounded by linear and/or quadratic surfaces combined with Boolean operators of intersection and union. Besides, it allows unary complement operator of a cell.
- It has provision for specifying universes and lattices, enabling built of complex repeated structures with ease
- Built with latest computational advances as dynamic memory allocation and also incorporated several new algorithms such as binary search and logarithmic energy grid for the optimization of time.
- Interactive debug features of 2D geometry plotting with void and overlapping region detection capabilities. Attempts are being made to run on Parallel computers.

The neutron cross-section data used and the solution method for estimating k_{eff} are given under section 2, results are presented in section 3, Interactive graphics are explained in section 4, and section 5 shows the details of parallel computing.

2. Methods of solution

For criticality problems, a steady-state neutron transport equation, which is in the Eigen value-form, is solved [16-17]. The eigenvalue equation(ref [18] eq. 1) is:

$$[\Omega \cdot \nabla + \Sigma_t(\vec{r}, E)] \Psi(\vec{r}, E, \Omega) = \iint \Psi(\vec{r}, E', \Omega') \Sigma_s(\vec{r}, E' \rightarrow E, \Omega \cdot \Omega') d\Omega' dE'$$

$$+ \frac{1}{k_{eff}} \frac{\chi(E)}{4\pi} \iint v \Sigma_f(\vec{r}, E') \Psi(r, E' \rightarrow E, \Omega' \rightarrow \Omega) dE' d\Omega'$$

Where,

$\Psi(\vec{r}, E, \Omega)$	- neutron angular flux
$\Sigma_t(\vec{r}, E)$	- total cross section
$\Sigma_f(\vec{r}, E')$	- fission cross-section
v	- the average number of neutrons emitted per fission
$(r, E' \rightarrow E, \Omega' \rightarrow \Omega)$	- the probability of transfer from (E', Ω') to (E, Ω)
k_{eff}	- is the neutron multiplication of the system.

The left-hand terms represent the neutron leakage and absorption and the first term on the right-hand side is the scattering term, and the second term describes the neutron production term due to fission. The constant k_{eff} is an eigenvalue included in the above equation to match both sides of the equation. The Monte Carlo method proceeds for the solution by the methods of successive generations or power iteration method [17] to estimate the largest eigenvalue of the system. The definition of k_{eff} is given as

$$k_{eff} = \frac{\text{fission neutrons produced in the current generation}}{\text{fission neutrons started in the previous generation}}.$$

In Monte Carlo simulations, generation means tracking neutron events from the birth of a neutron (site of fission) to its death (parasitic absorption or escape from the system) over a batch of neutrons. Any individual history (sequence of recorded events) from the batch may not have any physical meaning. However, an average of histories over a large number contains the physical characteristics of the medium. The required parameter can be deduced from the histories. At the end of tracking neutrons, the predetermined number of neutrons, k_{eff} is estimated, and fission sites are stored for subsequent iterations. The iterations are stopped when the value of k_{eff} of a system converges. At the end of the simulation, the code estimates from the recorded histories the neutron multiplication factor (k_{eff}) of a system in three different ways, namely collision estimator (k_{eff}^{col}) (ref. [19] page 2-165), absorption estimator (k_{eff}^{abs}) (ref. [19] page 2-166), and track length estimator (k_{eff}^{tra}) (ref. [19] page 2-167), along with their standard deviations. Further, the combined multiplication factor (k_{eff}^{com}) (ref. [19] page 2-170) is estimated using these three along with their uncertainties to give a reduced uncertainty band [20].

For the precise estimation of k_{eff} , it is necessary to account for all predominant modes of interaction of neutrons with the nuclei present in the system. Generally, neutron interactions with nuclei are radiative capture, elastic scattering, inelastic scattering, nuclear fission, and particle emission reactions such as $(n,2n)$, $(n,3n)$, (n, d) and (n, p) etc. NJOY [21] nuclear data processing system generates ACE formatted files from ENDF/B [22] data

for nuclides of interest. All the reactions given in ENDF are simulated, including reactions which produces secondary neutrons, such as (n, 2n), (n, 3n), level inelastic scattering and fission, according to the secondary energy and angle distribution laws given in the ACE file. MCkeff uses a free gas approximation [23] to treat scattering interactions when the target nucleus is in motion, where the velocities are sampled from a Maxwellian distribution. For thermal neutrons below 4 eV, S (α , β) scattering law data is used as the free gas approximation are inaccurate. For intermediate energy neutrons in the unresolved resonance range where the self-shielding effects are dominant (few keV to tens of keV in heavy elements), MCkeff uses the probability table method [24]. The nuclear data generated for present calculations is from ENDF/B-VII.I.

To improvise the computation time MCkeff uses binary search algorithm and logarithmic energy grid method where, Each equal-lethargy segment is mapped to a bounding index on the energy grids. For the ease of modelling the geometry the geometry plotter is developed with the new feature of identifying the overlap and void region of the geometry.

3. Results and Discussions

The MCkeff code developed accepts the input file in the field free format of a text file containing the information on system geometry, material information and the data to control the criticality simulation. Each line can be a maximum of 256 characters long, “&” signifies the following line is the continuation of the data line, “\$” symbol denotes the end of a line comment, the characters beyond it is ignored, and “C” followed by space is a comment line. This code accepts cross-section data in the ACE [11] formatted file and the results reported here are generated using ENDF version 7.1. Geometric cells bounded by surfaces containing materials are specified by combining the surfaces with binary operators such as union, intersection and complement of previously defined cells. It is possible to construct any complex configurations of materials with a limited number of surfaces. In the present code, we use 21 basic surfaces [25] of the first and second degree, which are enough to describe any conceivable geometry. The MCkeff is capable of handling repeated structures (Lattices) with the help of the universe concept similar to that of OpenMC.

The newly developed code testing and validation of the results are carried out in the following four steps:

- Testing of a variety of geometrical configurations of materials in cells,
- Validation of special physics treatment S (α , β) and probability table (ptable) method
- Validation of full core problem and other criticality benchmark problems, and
- Testing applicability for subcritical assembly

3.1 Testing of a variety of geometrical configurations of materials in cells

The geometry of the problem is described in terms of cells containing homogenous material regions. The cells can be formed by signed linear and quadratic surfaces (21 in number and sub-set of those permitted in MCNP (ref [25] Page No.3-13, Table 3.1: MCNP Surface Cards) along with the use of intersection and union operators. To test the proper implementation of geometrical features in the code, we have chosen the criticality problems depicted in the manual on ‘Criticality Calculation with Monte Carlo with MCNP5: A Primer [27]’. The results are generated by the MCkeff code running all the problems reported with the same ACE-format data file ENDF/B-VII.I [22]. All the validation problems were run with 5000 particles per cycle over 250 cycles skipping 50 cycles before accumulating the results unless otherwise explicitly stated. 2D-view of geometric cells for each problem, and the results generated are discussed in the sequel.

3.1.1 Jezebel model, Bare Plutonium sphere with Nickel shell (input file: Example 1-2a. of ref [27] Page 161): We start with a simple benchmark problem of a plutonium sphere of 1D geometry. This is a bare sphere of metallic plutonium of radius 6.38493 cm with a coating of nickel shell of thickness 0.0127 cm (also known as Jezebel reactor). 2D view of the geometry obtained with plot option of MCkeff code is shown in Fig.1. Since the Nickel coating is very thin, its zoomed view is shown in the adjoining figure.

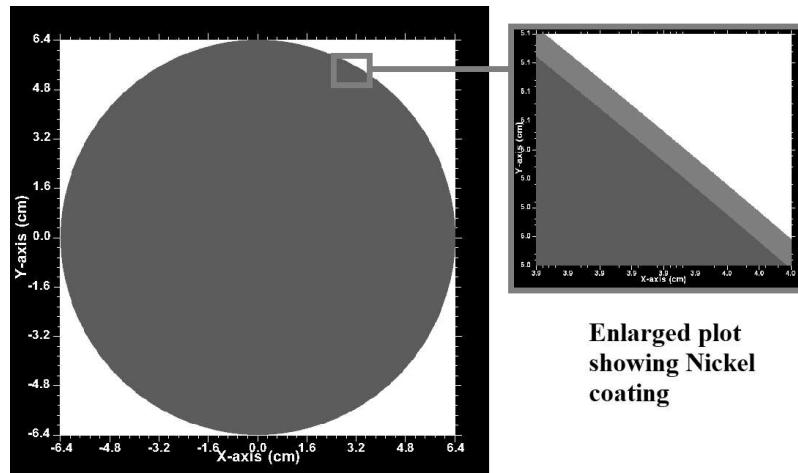


Fig.1. Jezebel model. Bare Plutonium sphere with Nickel shell (Shell thickness is too small and is shown in expanded view in the inset)

The MCNP and the MCkeff code combined keff values ($k_{\text{eff}}^{\text{com}}$) along with $\pm 1\sigma$ are 0.99902 ± 0.000057 (ref. [27] page 21) and 0.99981 ± 0.000060 respectively. The agreement between them is excellent and less than 1%.

3.1.2 Pu cylinder radial U(nat) reflector (input file: Example 2.5 of ref. [27] page 179): This problem is chosen to test the 2-dimensional cylindrical geometry capability of the MCkeff code. This criticality benchmark problem consists of a Plutonium metal cylinder of radius 4.935 cm and a height of 6.909 cm reflected on all sides with a 5cm thick shell of natural uranium. The horizontal and vertical sectional plot of the geometry is shown in Fig.2.

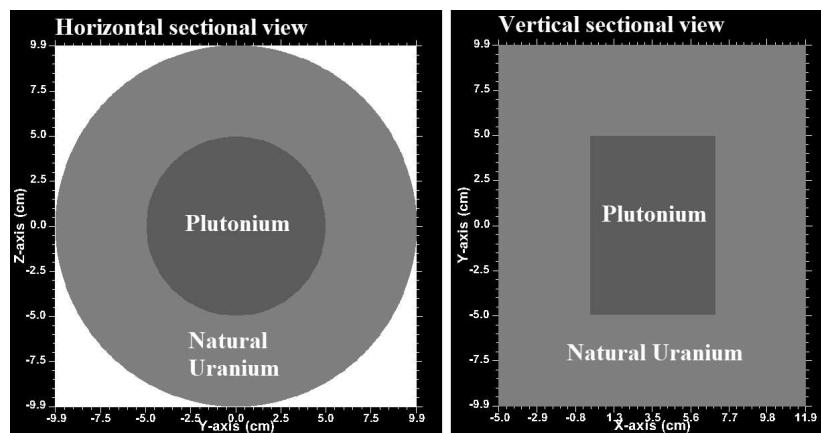


Fig.2. Pu cylinder, radial U(nat) reflector-horizontal and vertical views

The combined $k_{\text{eff}}^{\text{com}}$ values along with an estimate of $(\pm 1\sigma)$ standard deviations of for MCNP and MCkeff codes are 1.02486 ± 0.00066 (ref [27] page 40) and 1.02663 ± 0.00077 respectively. The agreement between them is remarkable, and the difference in values is around 0.1%.

3.1.3. Square Lattice of 3x2 Pu Cylinders (input file: Example 5.3 of ref [27] page 183): This problem demonstrates the capability of handling rectangular lattices by the MCkeff code. The problem consists of partially filled Pu cylindrical containers, and it is designated as universe 1. Cell card 5 defines the boundaries of lattice-cell in terms of signed surface numbers (8,9,10,11), given in the input file is reproduced below.

```
5 0 -8 9 -10 11 u=2 lat=1 fill=1 imp:n=1 $cell(lattice type) filled with universe 1.
```

This card defines a square cell of 35.58×35.58 cm and infinite in Z-direction, which is filled with universe 1, i.e., Pu solution container of cylindrical shape. The number of lattice elements is undefined (infinite in x and y directions). To limit these lattice elements to a finite number, lattice elements are kept in a window cell of finite size. The specification given in the input file for window cell 6, which is filled with universe 2 (lattice cell) is reproduced below.

```
6 0 91 -121 111 -141 3 -4 fill=2 imp:n=1 $ window cell filled with lattice elements.
```

The cell dimensions are 106.74×71.16 which limits the lattice elements 3 in x-direction and 2 in the y-direction. The plots of this input file obtained from the MCkeff code are shown in Fig.3.

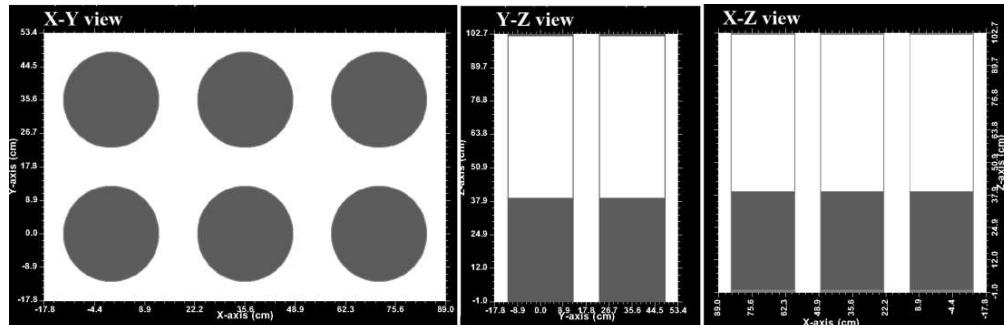


Fig.3. Square Lattice of 3x2 Pu Cylinders

The purpose of this problem is only to demonstrate the geometrical treatment adopted in the code, and hence $S(\alpha, \beta)$ treatment present in the original input file is commented out while running the problem. The source points are the midpoints in all these lattice elements given as initial source points. The MCNP-and the present code combined $k_{\text{eff}}^{\text{com}}$ along with an estimated $(\pm 1\sigma)$, standard deviations are 0.98536 ± 0.00102 (ref [27] page. 80) and 0.99462 ± 0.00103 , respectively. The agreement between them is less than 1% and validates the geometrical lattice treatment implemented in the MCkeff code.

3.1.4. 3-D Lattice (3x2x2) Pu Cylinders (input file: Example 7.3 of ref. [27] page 190): This is the case of a 3D lattice. This input is similar to that given in the example in 3.1.3 above, except it has two Z-direction levels. Universe 1 is defined in the first four cells of Pu solution cylinders and outside space. This is filled in cell 5, which is a lattice cell (given below) and designated as universe 2, with signed bounding surfaces (8,9,10,11).

```
5 0 -8 9 -10 11 -16 3 u=2 fill=1 lat=1 imp:n=1 $ lattice cell filled with universe 1.
```

And cell 6 is a window cell with enough dimensions to fill the universe 2, with $3 \times 2 \times 2$ lattice elements. The window cell six dimensions are “ $106.74 \times 71.16 \times 226.69$ ”, limiting the lattice elements 3 in the x-direction and 2 in the y-direction and 2 in the z-direction. The plots of this input file are shown in Fig.4.

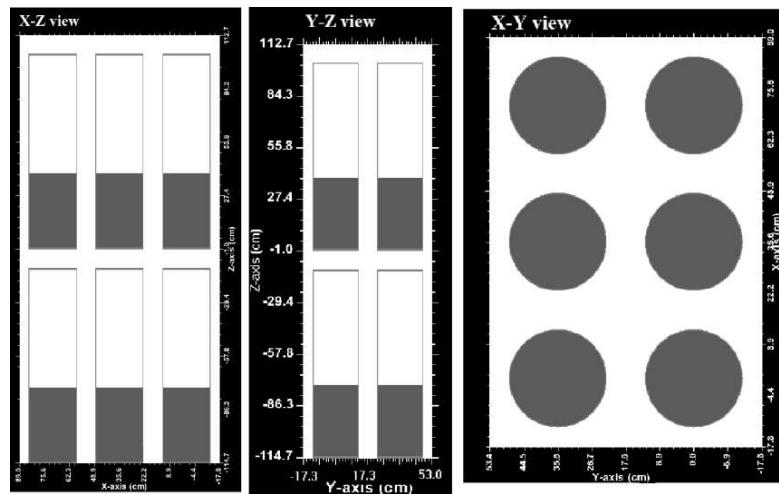


Fig.4. 3D Lattice o($3 \times 2 \times 2$) Pu Cylinders

The MCNP and the MCkeff code combined k_{eff} values along with $(\pm 1\sigma)$ values are 0.99588 ± 0.00102 (ref [27] page 110) and 1.00137 ± 0.00111 respectively. The agreement between them is very good, and the difference in values is less than 1 %.

3.1.5. Hexagonal Lattice of Uranium Cylinders (input file: Example 6.3 of ref. [26] page 187): To test the implementation of hexagonal lattice capability of MCkeff, this input has been chosen from ref [27] with UO_2F_2 (93.2% enrichment) solution of seven cylinders. The separation between the cylinders is 7.60 cm, which also has a 20 cm of water reflection below and radially about the cylinders. Opposite surfaces of the hexagonal lattice cell must be parallel and equal in length. Here we created an infinite array of solution cylinders which are truncated to 7 in number by a filling cell. The plot of geometry is shown in Fig.5.

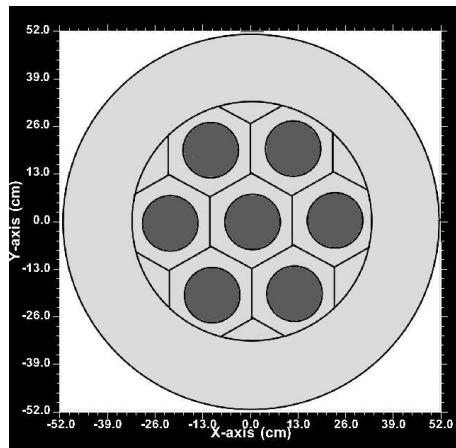


Fig.5. Hexagonal Lattice of Uranium Cylinders

Here the aim is only to test the treatment of hexagonal lattice geometry implementation in the code. Hence, the S (α , β) treatment present in the original input file is suppressed in generating results. The MCNP and the present code MCkeff combined k_{eff} values are 1.01637 ± 0.00267 (ref [27] page 100) and 1.08392 ± 0.00098 , respectively. The differences in the results between the two code is due to the different cross-sections used for the simulation.

3.2 Validation of special physics treatment S (α , β) and ptable method

This section presents to test the physics models implemented in the code, i.e., random sampling of secondary energy and angle distributions. There are two essential aspects to be taken care of in the Monte Carlo simulation. In addition to normal scattering and absorption, at low neutron energies (\sim eV), the thermal motion of target nuclides plays a significant role [27], and at intermediate energies (\sim keV), unresolved resonance effects [28] influence the simulated results. Without addressing these problems, the Monte Carlo simulation results might be misleading in some situations. The following sample problems are chosen to depict the changes that result in the k_{eff} values with and without considering these effects.

3.2.1. Thermal Neutron Scattering S (α , β) Laws for light materials: This criticality benchmark problem was chosen to demonstrate two aspects of MCkeff: thermal scattering laws and the implementation of reflective geometry boundary conditions. As a default, MCkeff assumes all surfaces are transmissive. The input file under surface cards contains the following lines:

```

1    cz    1.27
*2   px    -2.54
*3   px    2.54
*4   py    -2.54
*5   py    2.54.

```

In the above definitions of surfaces, surface number 1 is a cylindrical surface on the z-axis and surface numbers 2 to 5 are planar surfaces, precedes with *, which stands for reflective boundary condition. Further under data block, the materials are given as

```

m1  92238.72c 0.9902 92235.72c 0.0098 1.0
m2  1001.72c   2.0      8016.72c  1.0
mt2 hh2o.71t.

```

In the above data cards, the presence of mt2 card signals the code to use S (α , β) treatment for hydrogen in the water for material 2 below 4 eV of neutrons. It is essential to precisely treat the scattering of neutrons from bound scatterers such as hydrogen in water etc. Two scattering processes are addressed (1) inelastic scattering with cross-section σ_{in} and a coupled energy-angle representation from an ENDF S (α , β) scattering laws, and (2) elastic scattering without change in the outgoing neutron energy for solids with cross-section σ_{el} and an angular treatment derived from lattice parameters. Cullen [28] had proposed a simple pin cell model, which probably may highlight the differences in answers due to a variety of bound scattering treatments and the details can be found in the same reference. By having a very simple geometrical model, any differences in answers can be almost solely attributed to the bound-scattering treatment.

All six sets of pin cell problems with and without S (α , β) have been run with two codes, namely MCkeff and OpenMC identical physics treatments and criticality control parameters. The geometry plot edited is shown in Fig.6.

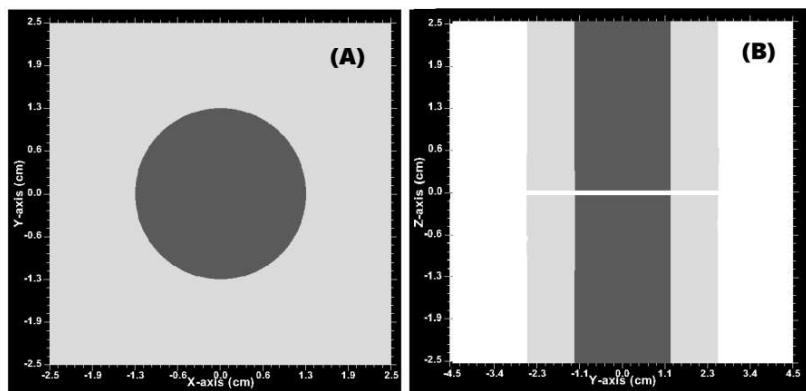


Fig 6. $\frac{1}{2}$ inch pin cell 2D plot of benchmark criticality problem (A) X-Y plot, (B) Y-Z plot (z-infinite)

Table 1. Comparison of combined ($k_{\text{eff}}^{\text{com}} \pm 1\sigma$) values for six different cases

Case of fuel pin	MCkEff ($k_{\text{eff}}^{\text{com}} \pm 1\sigma$)	OpenMC-0.8.0 ($k_{\text{eff}}^{\text{com}} \pm 1\sigma$)
1/2" pin, without S (α, β)	1.01710 ± 0.00004	1.01703 ± 0.00004
1/2" pin, with S (α, β)	0.96822 ± 0.00004	0.96815 ± 0.00004
1/4" pin, without S (α, β)	1.01519 ± 0.00005	1.01533 ± 0.00005
1/4" pin, with S (α, β)	0.92228 ± 0.00005	0.92227 ± 0.00005
1/8" pin, without S (α, β)	1.01780 ± 0.00005	1.01769 ± 0.00005
1/8" pin, with S (α, β)	0.90942 ± 0.00005	0.90943 ± 0.00005

Table 1 presents the comparison of combined ($k_{\text{eff}}^{\text{com}} \pm 1\sigma$) values for the six different cases. From the above table, it is clear that the results from the MCkEff are in good agreement with a difference in reactivity of less than 170 pcm with respect to OpenMC.

3.2.2. Unresolved resonance region (URR) or probability table treatment method: It is impossible to resolve the details of all resonances experimentally since they are narrow for heavy elements. Not accounting for these effects will result in severe errors in generated answers for the problems which are sensitive to these unresolved resonance ranges. For validating this physics treatment implementation in MCkEff, the Big Ten critical assembly [29,30] problem was chosen. The details are shown in the geometry plots of Fig.7.

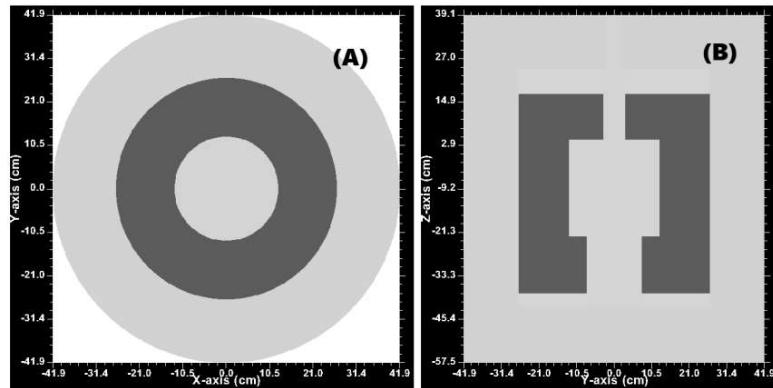


Fig.7 Big Ten benchmark. (Benchmark – IEU-MET-FAST-007)

The above problem was run with OpenMC and MCkeff with identical criticality control parameters. Table 2 results inspection shows that the unresolved resonance probability table treatment in MCkeff has been implemented correctly, with results agreeing with the OpenMC values with less than 100 pcm in reactivity.

Table 2. Comparison with/without ptable $k_{\text{eff}}^{\text{com}}$ for Big Ten benchmark (Benchmark – IEU-MET-FAST-007 [mit-crpg/benchmarks])

ptable	MCkeff ($k_{\text{eff}}^{\text{com}} \pm 1\sigma$)	OpenMC ($k_{\text{eff}}^{\text{com}} \pm 1\sigma$)
on	1.00473 ± 0.00011	1.00460 ± 0.00011
off	1.00052 ± 0.00011	1.00086 ± 0.00011

3.3 Validation of full core problem and other criticality benchmark problems

Hoogenboom et al. [30] had proposed a criticality Benchmark problem of a typical reactor core to test the efficacy of newly developed Monte Carlo codes. Fig. 8 shows the layout of the (A) 1/4th of the core, (B) fuel assembly within a core, including guide tubes and (C) the fuel pin. MCkeff and OpenMC were run with 100,000 particles per cycle, 150 inactive batches, and 4000 active batches. These two codes do not use any biasing techniques or convergence acceleration techniques, using the same ENDF/B-VII.1 libraries. Table 3 shows the k_{eff} results with their standard deviations.

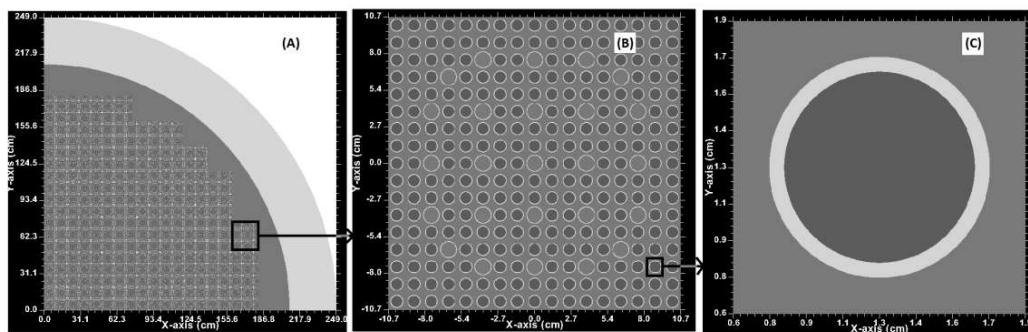


Fig. 8. shows the layout of (A) the 1/4 th core, (B) fuel assembly within a core including guide tubes and (C) the fuel pin.

Table 3. Comparison of $k_{\text{eff}}^{\text{com}}$ values for performance Benchmark problem with 3 different codes

Performance benchmark	$k_{\text{eff}}^{\text{com}} \pm 1\sigma$
MCkeff	1.00088 ± 0.00004
OpenMC	1.00105 ± 0.00004

From the above table, it is clear that the k_{eff} value of MCkeff is in good agreement with OpenMC with a minimal difference of less than 100 pcm in reactivity.

Tables 4 to 6 show the validation of k_{eff} values obtained with MCkeff and other standard codes. These problems were run with codes, namely MCkeff and OpenMC, for different KCODE control parameters such as number of neutrons per cycle, number of inactive cycles and the total number of cycles as per the input decks given in the appendix G of reference [27]. The cross-sections used are ENDF\B version 7.1 instead of ENDF\B version

6.6. In all cases, combined k_{eff} values are presented along with one standard deviation (1σ). The agreement of results between the two codes lends further confidence to implementing various geometry and neutron-nuclide collision physics models in the MCkeff code. Table 4 shows comparison results of Various geometrical configurations such as cylinders, lattices with multiple combinations for the problems given in the appendix-G of reference [27].

Table 4. Comparison of Combined k_{eff} values for input decks given in the appendix-G of reference [26] with cross-section set of ENDF version 7.1 with Plutonium fuel for various simple geometrical configurations.

SN	Problem Title	MCkeff* ($k_{\text{eff}}^{\text{com}} \pm 1\sigma$)	MCNP($k_{\text{eff}}^{\text{com}} \pm 1\sigma$) ref [26]
1	Bare Pu sphere w/ Ni shell (Example 1-2a.)	0.99981 ± 0.00060	0.99902 ± 0.00057
2	Bare Pu Cylinder (Example 2-3)	1.01577 ± 0.00069	1.01403 ± 0.00066
3	Pu cylinder, radial U(nat) reflector(Example 2-4)	0.88640 ± 0.00062	0.88367 ± 0.00052
4	. Pu cylinder, radial U(nat) reflector(Example 2-5)	1.02663 ± 0.00077	1.02486 ± 0.00066
5	Repeated Structures: Two Cylinders (Example 4-3)	1.05971 ± 0.00107	1.00803 ± 0.00091
6	Square Lattice of 3x2 Pu Cylinders (Example 5-3)	0.99462 ± 0.00103	0.98536 ± 0.00102
7	Hexahedral Lattices. (Example 5-4)	0.94478 ± 0.00100	0.94631 ± 0.00092
8	Hexahedral Lattices. Lattice with 1 empty element. (Example 5-5)	0.91206 ± 0.00072	0.91204 ± 0.00104
9	Hexahedral Lattices. Different Fill Cell Size(Example 5-6)	0.93945 ± 0.00095	0.93707 ± 0.00105
10	3-D (3x2x2) Lattice (Example 7-3)	1.00137 ± 0.00111	0.99588 ± 0.00102
11	3-D Lattice with one water element. (Example 7-9)	0.99316 ± 0.00098	0.99243 ± 0.00095

*All Values obtained from MCkeff are in good agreement with the published result of MCNP given in reference [27], and the minor differences are due to the change in the cross-section library used.

Table 5 presents the comparison results among the Monte Carlo codes, namely MCkeff, MCNP and OpenMC. All these codes use ACE format cross-section set and are run with the set of ENDF\B version 7.1 for identical criticality control parameters. These problems are chosen to test MCkeff code capability in handling different fuel compositions. The input files are from the International Criticality Safety Benchmark Evaluation Project (ICSBEP) [31] obtained from reference [30].

Table 5. Comparison of ($k_{\text{eff}}^{\text{com}} \pm 1\sigma$) of few criticality benchmark problems using MCkeff with MCNP and OpenMC

S. N	Title	MCkeff*	OpenMC	MCNP (ref [32] table 5a, 5b, 5c, 5d)
1	Reflected U-Hydride critical assembly, HEU- COMP-INTER-003 case 6	0.99700 ± 0.00127	0.99613 ± 0.00017	0.99748 ± 0.0004

2	Bare HEU sphere (Godiva), ref. HEU-MET-FAST-001 detailed model	1.00032 ± 0.00054	0.99988 ± 0.00012	0.99940 ± 0.0002
3	Topsy Oralloy sphere 2-in Tuballoy reflector, HEU-MET-FAST-003-case-1	0.99527 ± 0.00113	0.99492 ± 0.00013	0.9949 ± 0.0002
4	ZEUS-1 10 Uniform Units, HEU-MET-INTER-006 case 1	0.99323 ± 0.00053	0.99282 ± 0.00055	0.9929 ± 0.0002
5	Un-reflected sphere of enriched Uranium nitrate,	0.97928 ± 0.00051	0.99894 ± 0.00054	0.9985 ± 0.0002
6	Jemima #1, Idealized Model, IEU-MET-FAST-001 Case 1	1.00136 ± 0.00074	1.00030 ± 0.00013	1.0009 ± 0.0002
7	Pu sphere surrounded by HEU, Planet assembly, MIX-MET-FAST-001	0.99991 ± 0.00062	0.99939 ± 0.00012	0.9998 ± 0.0002
8	Bare Pu-239 Jezebel, PU-MET-FAST-001	1.00030 ± 0.00055	0.99921 ± 0.00041	0.9993 ± 0.0002
9	Bare U-233 Jezebel, U233-MET-FAST-001	1.00095 ± 0.00067	1.00030 ± 0.00012	1.0000 ± 0.0002
10	Uranyl-Flouride Solution with Beryllium reflector, U233-SOL-INTER-001 Case 1	0.99019 ± 0.00125	0.98509 ± 0.00020	0.9845 ± 0.0005
11	Un-reflected sphere of U233 nitrate, U233-SOL-THERM-001 Experiment 1	0.97839 ± 0.00094	1.00104 ± 0.00017	1.0010 ± 0.0002

From table 5, it can be inferred that the $k_{\text{eff}}^{\text{com}}$ and standard deviation ($\pm 1\sigma$) values obtained from the MCkeff code are in good agreement with the other two Monte Carlo codes values for all the cases. This demonstrates the physics treatment of neutron-nuclei interactions and secondary energy angular distributions modelled correctly in the MCkeff code.

Table 6. shows the comparison of combined k_{eff} values obtained from MCkeff and OpenMC codes. This set of inputs is prepared to demonstrate the code's ability to handle special features of geometry configurations incorporated in the code. Inputs contain cell translations, 2D and 3D lattices with and without fill patterns and partially filled 2D and 3D lattices. Problems with reflective and periodic boundaries are also given.

Table 6. Comparison of ($k_{\text{eff}}^{\text{com}} \pm 1\sigma$) MCkeff results with that of OpenMC for testing of geometry capability.

S.No.	Problem Title	MCkeff ($k_{\text{eff}}^{\text{com}} \pm 1\sigma$)	OpenMC ($k_{\text{eff}}^{\text{com}} \pm 1\sigma$)
1	Fill and universe attributes test	1.02912 ± 0.00213	1.02912 ± 0.00213
2	Test for complement operator	0.99981 ± 0.00393	0.99981 ± 0.00393
3	Cell translation test (trcl)	0.99048 ± 0.00047	0.98999 ± 0.00045

4	k-inf Experiment, PU-COMP-INTER-001 reflective boundary	1.01025 ± 0.00065	1.01129 ± 0.00056
5	k-inf Experiment, PU-COMP-INTER-001 periodic boundary	1.01133 ± 0.00066	1.01024 ± 0.00075

From table 6, it can be inferred that the $k_{\text{eff}}^{\text{com}}$ and standard deviation ($\pm 1\sigma$) values obtained from the MCkeff code are in good agreement with OpenMC generated values for all the cases. This provides the confidence the special features of geometry are incorporated in the code correctly.

3.4. Testing applicability for subcritical assembly

Nuclear fuel storage facilities and fuel reprocessing vessels for safe operation point of view are designed to have a k_{eff} value well below 1. Most often, it is required to predict k_{eff} values precisely for these configurations also. Here, a typical fuel storage cage containing PuO₂ powder is shown in Fig.9.

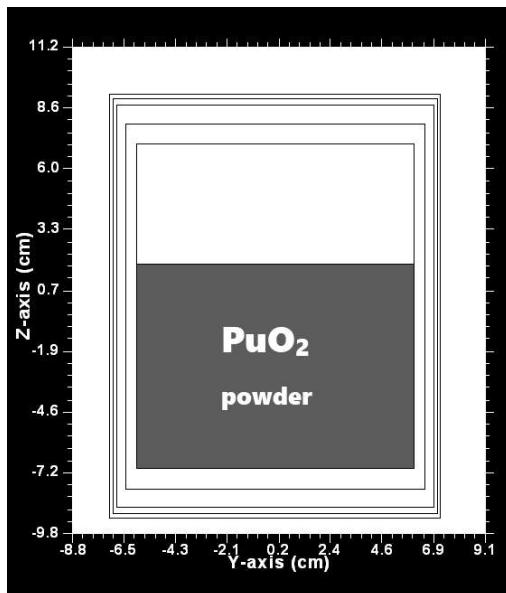


Fig. 9. Y-Z plot of an isolated fuel storage facility

Table 7 shows the comparison of results of sub-critical assembly problems. They are related to typical fuel reprocessing vessels and fuel storage facilities. Both the codes have been run with the cross-section set of ENDF version 7.1 with identical criticality control parameters. The agreement between them is very good, and the values are within $\pm 1\sigma$ confidence interval

Table 7. Comparison of ($k_{\text{eff}}^{\text{com}} \pm 1\sigma$) of sub-criticality values between MCkeff and OpenMC.

SN	Title	MCkeff ($k_{\text{eff}}^{\text{com}} \pm 1\sigma$)	OpenMC ($k_{\text{eff}}^{\text{com}} \pm 1\sigma$)
1	K _{eff} of Mixing vessel	0.62147 ± 0.00165	0.61958 ± 0.00168
2	fuel storage cage model (Fig.9)	0.32242 ± 0.00144	0.32303 ± 0.00137

All the values lie within $\pm 1\sigma$ values and show the validity of the MCkeff code for employing for estimation of neutron multiplication of sub-critical assemblies.

4. GUI developed for geometry Visualization.

To aid the user in preparing complex geometries and avoiding errors, it is desirable to have a visualization capability. An interactive GUI (Graphic User Interface) has been developed using the “DISLIN” library, which contains FORTRAN compatible graphic library routines for windows OS. Figure 10 shows the snapshot of the GUI developed. The left-hand side of the screen with text boxes containing data are for interacting purposes, and the right side is the display portion of geometry. The command buttons help to refresh the image with changed parameters.

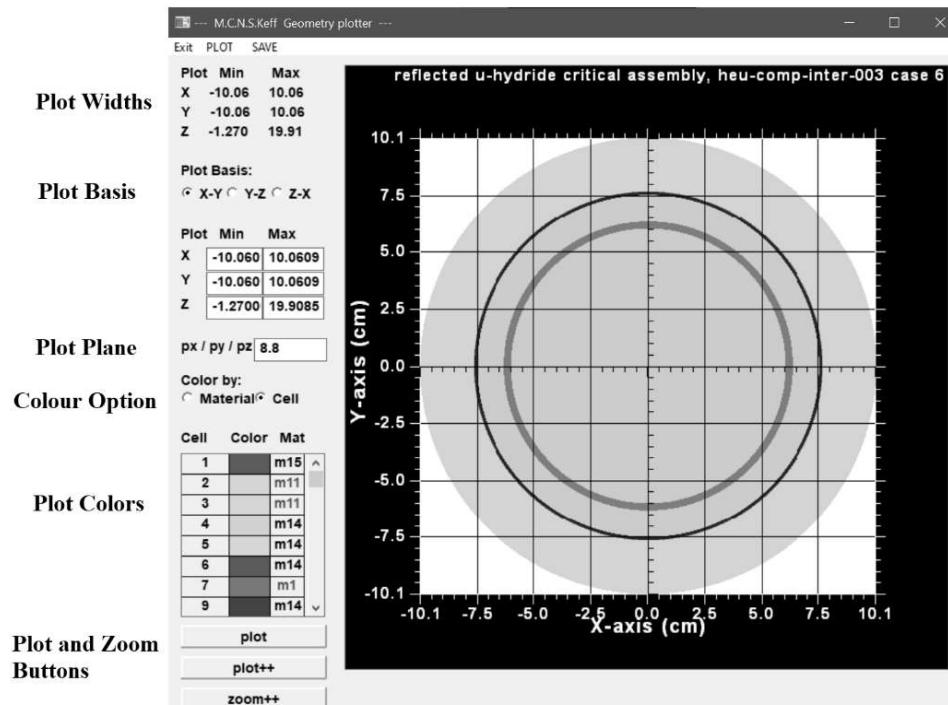


Fig 10. Geometry visualizing GUI of MCkeff.

The interactive features available are

- Users can change plot basis; as a result, geometry can be viewed in all dimensions
- Plot widths can be changed to look at the geometry in the case where geometry is very complex.
- Users can change the plot colour format either to colour by cell or materials used.
- Multiple plots can be seen with the use of the PLOT++ button.
- Zoom option provided with the ZOOM++ button.
- The plots can be saved in PNG or PDF format with the help of the save button in the menu.
- Overlapping and void regions will appear in black color, if present.

5. Parallel Computing time statistics

Monte Carlo algorithms are inherently suitable to run on parallel computers. The MCkeff code employs OpenMP [10] parallel computing libraries, which helps the code use all the cores present in the processor. However, MPI parallel computing libraries with multiple nodes are intended to be incorporated later in this project. The timing statistics of the serial

and parallel computing modes are shown in table 8. The Problems of ‘Reflected U-Hydride critical assembly, HEU-COMP-INTER-003 case 6’ have been chosen from ICSBEP with 100000 particles per batch and 1000 batches for demonstration purposes.

Table 8. Parallel Execution timing statistics of MCkeff

Processor	Serial mode(1CPU)	Parallel mode (4 CPU)
Intel(R) Core (TM) i5-8250U CPU @ 1.60GHz 1.80 GHz	78.2 mins	24.8 mins

It is very clear from the above table that the usage of parallel computing libraries in MCkeff reduced the computation time very significantly.

Conclusions

A new Monte Carlo neutron transport code, called MCkeff, has been developed to estimate the neutron multiplication factor of fissile systems. The code is developed in Fortran language using GNU compiler, which includes Fortran 2008 features, and is compatible with both Windows and Linux based operating systems. A point-wise, continuous energy ACE format cross-section was used to treat neutron-nuclei collision physics accurately (without approximations). To treat kinematics correctly at low energies (\sim eV), one can use $S(\alpha, \beta)$ scattering law data and for unresolved resonance energy ranges(\sim keV) probability tables. For constructing geometrical shapes containing homogenous material regions, first and second degree signed surfaces along with intersection and union operators are permitted. Since MCkeff is developed from scratch, it will be rather easy to incorporate advances happening in the computer industry as well as in the physics models.

Acknowledgements

This work was performed under the Department of Atomic Energy – Board of Research in Nuclear Sciences (DAE-BRNS) project No.36(4)/14/40/2015. The authors are thankful to DAE-BRNS for the sanctioning of the project. The authors are grateful to Manipal Centre for Natural Sciences, Centre of excellence, MAHE for allowing to use the facilities. Authors are thankful to Jayant Krishan and P S Sharma of HPD, BARC and Shri. Narsiah, of RSSD, BARC for evaluating the code and for providing useful suggestions.

References

- [1] Knief, R. A., Society, A. N., & Commission, U. S. N. R. (1985). *Nuclear Criticality Safety: Theory and Practice*. The Society.
- [2] Carter, L. L., & Cashwell, E. D. (1975). *Particle-transport simulation with the Monte Carlo method*.
- [3] Hammersley, J. M., & Handscomb, D. C. (1964). *Monte Carlo methods*. Methuen; Wiley.
- [4] Lux, I., & Koblinger, L. (1991). *Monte Carlo particle transport methods: Neutron and photon calculations*. Boca Raton: CRC Press.
- [5] Cramer, S. N. (1984). *KENO-V* code. https://inis.iaea.org/search/search.aspx?orig_q=RN:16061204
- [6] X-5 Monte Carlo Team, 2008a. *MCNP - A General Monte Carlo N-Particle Transport Code, Version 5*. LA-UR-03-1987, Los Alamos National Laboratory

- [7] Romano, P. K., & Forget, B. (2013). *The OpenMC Monte Carlo particle transport code*. *Annals of Nuclear Energy*, 51.
- [8] Leppänen, J. (2013). *Serpent—a continuous-energy Monte Carlo reactor physics burnup calculation code*. VTT Technical Research Centre of Finland, 4.
- [9] Kirk, B. L. (2010). *Overview of Monte Carlo radiation transport codes*. *Radiation Measurements*, 45(10), 1318–1322.
- [10] Dagum, L., & Menon, R. (1998). *OpenMP: an industry-standard API for shared-memory programming*. *IEEE Computational Science and Engineering*, 5(1).
- [11] X-5 Monte Carlo Team, “*MCNP - A General Monte Carlo N-Particle Transport Code, Version 5, Volume III: Developer’s Guide*,” LA-CP-03-0284, Los Alamos National Laboratory (2008).
- [12] *The GNU Operating System and the Free Software Movement. [A GNU head]*. (n.d.). <https://www.gnu.org/>.
- [13] Intel Parallel Studio XE: Advanced coding tools for developers: Design-In Tools Store. [\(n.d.\).](https://designintoools.intel.com/Intel_Parallel_Studio_XE_p/sftintparstu.htm) https://designintoools.intel.com/Intel_Parallel_Studio_XE_p/sftintparstu.htm.
- [14] Revised on July 16, 2020. *Fortran Wiki Fortran 2008 status*. *Fortran 2008 status in Fortran Wiki*. <http://fortranwiki.org/fortran/show/Fortran+2008+status>.
- [15] Michels, H. (n.d.). *Dislin*. <https://www.dislin.de/>.
- [16] Fynn Scheben, (2010). *Iterative Methods for Criticality Computations in Neutron Transport Theory*, PhD Thesis, University of Bath, Bath.
- [17] Lieberoth, J. (1968). *A Monte Carlo technique to solve the static eigenvalue problem of the Boltzmann transport equation*. *Nukleonik*, 11(5).
- [18] Forrest B. Brown (2009), *A Review of Monte Carlo Criticality Calculations - Convergence, Bias, Statistics*, LA-UR-08-06558, American Nuclear Society – LANL
- [19] X-5 Monte Carlo team, (2003), *MCNP — A General Monte Carlo N-Particle Transport Code, Version 5 Volume I: Overview and Theory* LA-UR-03-1987, Los Alamos National Laboratory
- [20] Todd J. Urbatsch, R. Arthur Forster, Richard E. Prael, and Richard J. Beckman, “*Estimation and Interpretation of k_{eff} Confidence Intervals in MCNP*,” *Nucl. Technol.*, 111, pp. 169-182 (August 1995).
- [21] R. E. MacFarlane, D. W. Muir, R. M. Boicourt, and A. C. Kahler, “*The NJOY Nuclear Data Processing System, Version 2012*,” LAUR-12-27079, Los Alamos National Laboratory (2012).
- [22] Chadwick, M. B. et al, (2011). *ENDF/B-VII.1 Nuclear Data for Science and Technology: Cross Sections, Covariances, Fission Product Yields and Decay Data. Nuclear Data Sheets*, 112(12), 2887–2996. <https://doi.org/10.1016/J.NDS.2011.11.002>
- [23] T. M. Sutton, T. H. Trumbull, and C. R. Lubitz, “*Comparison of Some Monte Carlo Models for Bound Hydrogen Scattering*,” *Proc. Int. Conf. Mathematics, Computational Methods, and Reactor Physics*, Saratoga Springs, New York, May 3–7, 2009.
- [24] Levitt, L. B. (1972). *The Probability Table Method for Treating Unresolved Neutron Resonances in Monte Carlo Calculations*. *Nuclear Science and Engineering*, 49(4).
- [25] *OpenMC documentation, Users guide, Page 430 chapter 5. Releases · openmc-dev/openmc (github.com)*
- [26] X-5 Monte Carlo team, (2003), *MCNP — A General Monte Carlo N-Particle Transport Code, Version 5 Volume II: User’s Guide* LA-CP-03-0245, Los Alamos National Laboratory
- [27] X-5 Monte Carlo Team. (2004). *Criticality Calculations with MCNP: A Primer*. *World Wide Web Internet And Web Information Systems*, 836.

- [28] Cullen, D. E., Blomquist, R. N., & Dean, C. J. (2004). *How accurately can we calculate thermal systems? (No. INDC (USA)--107)*. International Atomic Energy Agency.
- [29] Levitt, L. B. (1972). *The Probability Table Method for Treating Unresolved Neutron Resonances in Monte Carlo Calculations*. Nuclear Science and Engineering, 49(4).
- [30] Paul, R. (n.d.). *mit-crpg/benchmarks: Collection of models for reactor physics benchmarks*. GitHub. <https://github.com/mit-crpg/benchmarks>.
- [31] Hoogenboom, J. E., Martin, W. R., & Petrovic, B. (2011, July). *The Monte Carlo performance benchmark test-aims, specifications and first results*. In International Conference on Mathematics and Computational Methods Applied to (Vol. 2, p. 15).
- [32] Briggs, J. B. (2003). *The international criticality safety benchmark evaluation project (ICSBEP)*.
- [33] Brown, Forrest B., Rising, Michael Evan, and Alwin, Jennifer Louise. *Verification of MCNP6.2 for Nuclear Criticality Safety Applications*. United States: N. p., 2017. Web. doi:10.2172/1356287.