Development of ICMC-1.0 Monte Carlo Code for Neutron and Particle Transport

H. Kumawat
Nuclear Physics Division
and
P.P.K. Venkata
Computer Division

Abstract
The Intra-nuclear Cascade Monte Carlo (ICMC) code for transport of neutrons, protons, pions and heavy ions has been developed at Nuclear Physics Division, BARC in the last few years, and further developments are underway. We have developed the code for low energy neutron transport using pointwise cross section data below 20 MeV of neutron energy. Constructive Solid Geometry model, based on solid bodies, is adopted to construct geometry. A module for repetitive structure for lattice, core calculations in reactors and detector simulations is developed. A Graphical User Interface (GUI) has been incorporated for making the input, construction and visualization of the geometry and analysis of the output. The code has been validated for simulating benchmarks of accelerator driven sub-critical systems, neutron shielding, heat and neutron flux distribution, and $k_{\text{eff}}$ of the critical and sub-critical assemblies.

Introduction
The statistical nature of nuclear reactions and propagation of particles through matter can be best simulated using the Monte Carlo method. This method is most suited to solve multi-dimensional problems, involving complex geometries and variation of cross sections with energy. The accuracy is limited only by the uncertainty in the input data such as cross sections. It provides a solution to the integral equation using random sampling in space [1]. The Monte Carlo method, used in simulation of nuclear reactions, is based upon generation of individual particle histories using random sampling methods. It can provide estimates of desired quantities such as $k_{\text{eff}}$ or flux which would be obtainable from a solution to the transport equation using random sampling in space without obtaining a detailed or complete solution of the transport equation. The probability of interaction is simulated with the help of random numbers and cross sections and these are primary input quantities which determine the accuracy of the method.

There are several Monte carlo codes named GEANT4 [2], FLUKA [3], PHITS [4], MCNP [5], MARS [6] for particle transport in matter. Some of these codes (GEANT4, FLUKA, and MARS) are suitable for detector optimization and high energy physics simulations. The other Monte Carlo codes viz. MCNP and TRIPOLI [7] use the continuous energy neutron cross-sections and are suitable for reactor simulations.

The Monali code [8] developed several years back uses multigroup cross section data library. Thus a need was felt to develop a continuous energy code in this centre.

The development of ICMC code along with GUI was started a couple of years ago. We have adopted Constructive Solid Geometry (CSG) model [9] for the construction of Geometry. Repetitive structure is introduced to perform lattice calculations. Continuous energy cross-section representation is used to take into account the details of the resonance structure. The article is organized as follows: Section 2 describes the implementation of
the geometry and the neutron transport is detailed in Section 3. Sec. 4 contains a description about the GUI. The high energy part of the code is described in Sec. 5. Conclusions and further developments are discussed in Sec. 6.

Construction of Geometry

The most important and difficult task in the Monte Carlo code is to build a complicated geometry in a user friendly manner. We have chosen the CSG model to build the geometry. In this model, there are simple basic geometrical bodies viz. Sphere, Cylinder, Box, Cone, Ellipse, Hexagon etc. Boolean operations (Union, subtraction and intersection) are used to construct complex zones using these bodies. We must also provide a universe that contains all the geometrical structures. Fig.1 gives an example to construct the zones from the bodies. Eight heterogeneous zones from three spherical bodies and a universe are made which can be filled with various materials.

Scaling, rotation and translation of the bodies are used to make more complicated structures. Repeated geometry structures are invoked to perform lattice and core calculations of complicated reactor assemblies. Fig. 2 shows an example of repeated structures which are made of different types of fuel rods. Bare minimum information viz. number of rods, radius of the ring on which rods are to be placed, and the center of the repeated structures has to be provided by the user. Reflective boundary conditions are used to do single lattice/cell or partial core calculations.

The transport algorithm is generalized for the non-intersecting assemblies/bodies (physically separated by vacuum), white/reflecting boundaries, and repeated structures. A geometry debugger is introduced to search overlapping/common volumes due to improper dimensions supplied by the user.

Low energy neutron transport

The Low energy neutron transport code has been developed recently. We have developed the package for reading pointwise cross sections for neutron in ACE (A Compact ENDF) format using arrays with dynamically allocatable memory. The ACE library generated using ENDF VII.0 is used for the present investigations. Interaction of neutrons is considered using the Monte Carlo method as per the following steps.

1) Identification of the initial zone number and point of interaction of the neutron,
2) Selection of the collision nuclide, 
3) Type of interaction (elastic, non-elastic, fission, capture, others).

ICMC-1.0 assigns the X, Y, Z, cosø, sinø, cosø, energy (MeV), charge, and mass(MeV/C^2) coordinates with each neutron. The code identifies the zone number constituted from the given bodies of the configuration defined in the input file. The macroscopic cross section is calculated to get the mean free path in the identified zone which is used to sample the distance to the next collision. The nuclide with which the collision takes place is identified using the fact that probability of interaction with a given nuclide is proportional to the total macroscopic cross-section of that nuclide. The final search is made for reaction type with the identified nuclide. One complete history consists of nuclear interactions; secondary particle production and their transport till predefined cutoff energies are reached. The neutron cutoff energy is defined to be 1×10^{-12} MeV in the present version of the code.

The criticality calculations in ICMC-1.0 are based on four methods (neutron population, Collision Estimator, Absorption Estimator, and Track Length Estimator). The k_{eff} is a ratio between the number of neutrons in successive generations in a fission chain reaction. For critical systems, k_{eff} = 1, for sub-critical systems, k_{eff} < 1 and for supercritical systems, k_{eff} > 1. The number of neutrons in successive generations is obtained from number of neutrons generated by fission. Whenever (n, xn) reactions occur, the neutrons generated are again transported within the same fission cycle. At present fission source points as well as neutron generations are as usual allowed as other reactions but stored for the next cycle. At the end of each cycle the total weight is maintained constant by increasing or decreasing weight of neutrons in case of (k_{eff} > 1) and (k_{eff} < 1), respectively. The maximum likelihood k_{eff} of the system from all four estimators is calculated using weighted mean where weight is given by inverse of the squared error from individual estimators. Error in the mean k_{eff} is also calculated similarly. The prompt energy spectrum is used in place of the delayed energy spectrum in case the latter is not available. The criticality calculation requires number of inactive cycles which need to be skipped to get the fundamental mode of fission source, active cycles for actual k_{eff} and number of source neutrons. Mono energetic neutron can be defined very easily in the input file and spectrum can be provided through a separate file. In case of high energy proton or other beam, the source distribution is generated using the high energy part of the code and that is transported below 20 MeV.

More than fifteen problems for fast, thermal and ADS systems for criticality benchmark have been simulated. Some of the results for few problems are given in Table 1. The geometries for the first five problems are shown in Fig. 3. Brief description of the simulated systems is given below:

Prob.1 is an enriched 235U (93.71%) sphere of radius 8.741cm consisting 52.42kg of mass and density 18.74g/cc rest is 238U.
Prob.2 is an enriched 239Pu (95.5%) sphere of radius 6.385cm consisting 17.02kg of mass and density 15.61g/cc rest is 240U.
Prob.3 is an enriched 239Pu (80%) sphere of radius 6.66cm consisting 19.46kg of mass and density 15.73g/cc rest is 240U.

<table>
<thead>
<tr>
<th>Problem #</th>
<th>Code</th>
<th>ICMC-1.0</th>
<th>MCNP</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>0.9963 ±.0008</td>
<td>0.9962 ±.0009</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>1.0047 ±.0003</td>
<td>1.0052 ±.0006</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>1.0086 ±.0004</td>
<td>1.0097 ±.0002</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td>0.9910 ±.0008</td>
<td>0.9915 ±.0005</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>0.9906 ±.0004</td>
<td>0.9908 ±.0006</td>
</tr>
<tr>
<td>6</td>
<td></td>
<td>0.8872 ±.0004</td>
<td>0.8865 ±.0001</td>
</tr>
<tr>
<td>7</td>
<td></td>
<td>0.9948 ±.0005</td>
<td>0.9952 ±.0003</td>
</tr>
<tr>
<td>8</td>
<td></td>
<td>0.9946 ±.0007</td>
<td>0.9951 ±.0006</td>
</tr>
</tbody>
</table>

Table 1: Values of k_{eff} for some of the experimental assemblies along with MCNP values from literature [10] are given. Here errors are estimated with most likelihood method for all four estimators.
Prob.4 is an enriched $^{235}$U (10.9%) cylinder of radius 26.65cm, Height=119.392cm and density 18.63g/cc rest is $^{238}$U.

Prob.5 is an enriched $^{235}$U (14.11%) cylinder of radius 26.65cm, Height=44.239cm and density 18.41g/cc rest is $^{238}$U.

Prob.6 is an enriched $^{239}$Pu (100.0%) cylinder of radius=4.935cm, Height=6.909cm and density=18.80g/cc. It is surrounded with Natural uranium reflector of Thickness=5cm, Height=6.909cm.

Prob.7 is an enriched $^{235}$U (93.5%) sphere of radius 7.3984cm and density=18.6g/cc. The sphere is surrounded by graphite of 5.1cm thickness. The graphite consists of 99.5% Carbon, 0.34% iron and 0.16% sulfur. Density of graphite is 1.67g/cc.

Prob.8 is an enriched $^{235}$U (97.67%) sphere of radius 6.5537cm consisting of 22.16kg mass and density 18.794g/cc. The sphere is surrounded by water tank of radius 30cm and height 70cm.

It is found that the calculated values from ICMC-1.0 are very close to the results obtained with another standard code like MCNP. The average value of $k_{eff}$ is obtained from the maximum likelihood method of the values obtained from all four estimators.

Graphical User Interface

Graphical User Interface (GUI) along with data visualization is a powerful tool required for supporting such ambitious software. The GUI and the visualization modules are developed by Computer Division. Development of these modules is done in Python language using the base libraries of Visualization Toolkit [11] for visualization and WxPython for GUI. The communication between the GUI and the Monte-Carlo code is through loose coupling, i.e. both these modules are independent of each other and the communication is through external files. The GUI and visualization modules are developed for cross-platform usage, so that they can be run on all windows and Linux platforms. One snapshot of the geometry from Ubuntu Linux machine is given in Fig.4.

To construct the geometry, all Boolean operations viz. union, subtraction, intersection are available in this framework to make complex zones from the basic bodies. Scaling, rotation and translation of the basic bodies is supported. This information is saved in a text input file and then ICMC code can be run either through terminal/command prompt or from the GUI button itself. Standard features viz. showing 3D-axis around bodies, taking snapshots are provided.
The most important feature of the GUI is to visualize and correct the geometry in 3-dimensions. Overlapping regions (if any) after scaling, rotation, and translation can be easily identified and corrected before running the Monte-Carlo code.

**High energy particle transport**

Monte Carlo program ICMC-1.0 also has the capability of high energy particle transport which is borrowed from CASCADE.04 [12-14] and its further developments [15-16]. It incorporates Intra-nuclear Cascade, Pre-equilibrium, Evaporation and Fission models to simulate spallation reaction mechanism for thin and thick targets. Treatment of cutoff energy from Intra-nuclear to pre-equilibrium and next to evaporation stage was modified later [17].

Benchmark of spallation models for experimental values of neutron, charged particles, and pions double differential production cross-sections, particle multiplicities, spallation residues and excitation functions was organized by IAEA and is given in Ref. [17]. Heat Deposition algorithm for thick spallation targets and thin films was modified and benchmarked as mentioned in Ref. [13]. The code was further developed for the Neutron shielding and dosimetry applications and published [14]. The high energy part of this code can be used for single nucleus interaction for basic reaction studies and transport of particles in thick target. Energy loss of the charge particle is calculated during the transport through thick target.

The flow chart of the code is given in Fig. 5, where, particle transport as well as single nucleus interactions are mentioned. This is an integral code to study Accelerator Driven Sub-critical systems with user defined options to be supplied by the user.

**Conclusion and Future Development**

The Monte Carlo code ICMC-1.0 has been developed for ADS, Spallation reactions, reactor physics, dosimetry, and shielding applications. New CSG model with Union, Subtraction and Intersection Boolean operations is developed to make the heterogeneous zones. Scaling, rotation, and translation operations are used to make more complex zones. Repeated geometry model has been developed for simulation of any complex reactor designs as well as the detector simulations. The pointwise cross section data for neutrons below 20MeV are used and we have developed a package for reading these cross sections in the ACE format using dynamically allocatable memory. The $S(\delta, \delta)$ scattering matrices for neutron energy $\leq 4$eV is used if it is available in the library for the given compound element, otherwise Fermi-gas treatment is used. The code has been benchmarked for $k_{\text{eff}}$ values simulated for many simple experimental assemblies and is under extensive benchmark for different assemblies including real Thorium Plutonium MOX fuel based AHWR system. List of fission products and their spatial distribution can be analyzed using this code at time $T=0$.

The code will be further developed for reactor burnup, decay heat, and waste management issues. In this development, we have generated one library of $\sim 3700$ isotopes to simulate the decay quantities viz. decay heat, neutrino flux etc. This library is generated from the ENDF VII.0 in which isotopes up to nano-second half lives are included. The library includes decay through $e^-$, $e^+$, EC, $\alpha$ and $\gamma$ decay channels. Ingestion and inhalation toxicity is also included in this library to include these aspects of waste management. This work package will be completed after the development of decay model. The code is under development for photon and...
electron transport. The parallel version of the code is also in progress.

**Acknowledgement**

The authors are immensely thankful to Dr. R.K. Sinha (Chairman, AEC), Dr. S. Kailas (Director, Physics Group), Dr. A. Chatterjee (Head, NPD) who supported this work. Constant encouragement of Dr. P.D. Krishnani (RPDD), Dr. S.B. Degweker (ThPD), Shri S.K. Bose (Computer Division), is highly acknowledged. We are indebted to Shri Anek Kumar (RPDD) for helping us in reading one of the ACE data files.

**References**