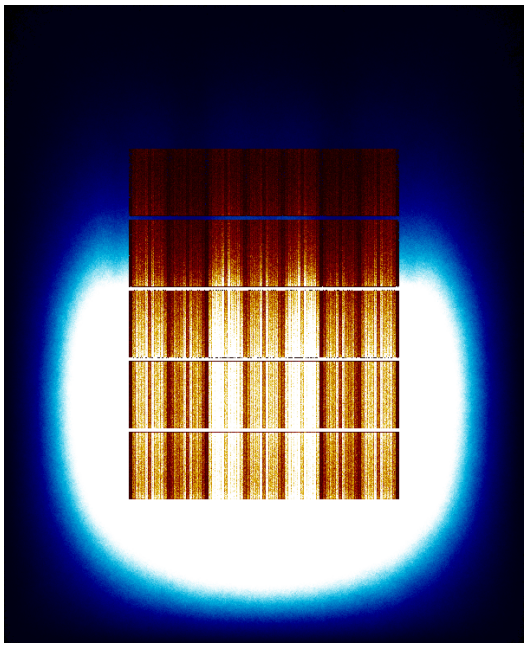
RESEARCH REPORT

VTT-R-01296-10



Serpent Progress Report 2009

Authors: Jaakko Leppänen

Confidentiality: Public

Report's title	
Serpent Progress Report 2009	
Customer, contact person, address Valtion ydinjätehuoltorahasto	Order reference ad 8/2007/SAF
Project name TOPAS 2010/Safir2010	Project number/Short name 41427
Author(s) Jaakko Leppänen	Pages 12
Keywords Monte Carlo, neutron transport, burnup, Serpent, PSG	Report identification code VTT-R-01296-10

Summary

This report summarizes the work carried out for the development of the Serpent Monte Carlo reactor physics burnup calculation code at VTT Technical Research Centre of Finland during year 2009. New features and major bug fixes are listed and four example applications are provided by user organizations.

Confidentiality	Public

February 12, 2010

Written by	Reviewed by	Accepted by
Jaakko Leppänen, Senior Research Scientist	Petri Kotiluoto, Senior Research Scientist	Timo Vanttola, Technology Manager

VTT's contact address
VTT Technical Research Centre of Finland, P.O. Box 1000, FI-02044 VTT

Distribution (customer and VTT)

Contents

1 Introduction 3

2 Background 3

3 Code Development 3

3.1 New features 4
3.2 Major bug fixes 5

4 Applications 6

4.1 VTT Technical Research Centre of Finland 6
4.2 Technical University of Catalonia, Spain 6
4.3 Australian Nuclear Science and Technology Organization (ANSTO) 7
4.4 Lappeenranta University of Technology, Finland 9
4.5 Royal Institute of Technology, Sweden 10

5 Future Plans 10

6 Summary and Conclusions 11

Acknowledgements 12

References 12

1 Introduction

Serpent is a three-dimensional continuous-energy Monte Carlo neutron transport code, capable of performing burnup calculation. The code is specifically designed for reactor physics applications and the original intended use was the production of homogenized multi-group constants for reactor simulator calculations, similar to conventional deterministic lattice transport codes. However, the versatility of the Monte Carlo method and some additional capabilities have allowed the field of applications to be extended well beyond group constant generation and lattice physics calculations, and Serpent is perhaps best characterized as a Monte Carlo "reactor physics code".

This report summarizes the work related to the Serpent project in the year 2009, although it is not intended to give a full description of the code capabilities. The project is being carried out at VTT Technical Research Centre Finland, with some significant contribution from Finnish universities. Code development is an on-going effort and feedback from the growing user's community is a valuable and appreciated resource. Some examples of user experience are given in Section 4.

2 Background

The Serpent code is the direct descendant of the PSG project, started at VTT in 2004 [1]. The name was changed in October 2008 when a pre-release version (1.0.0) was distributed to some research institutes for testing purposes. The source code was completely re-written and the main difference to the predecessor was the built-in burnup calculation capability, based on two fundamentally different solution methods for the Bateman depletion equations [2, 3]

The preparation of public distribution through the OECD/NEA Data Bank started in January 2009, and the code was finally released in May (version 1.1.0). Some users have also acquired the code by bilateral license agreements with VTT. Distribution to North America via RSICC was still being negotiated at the time of this writing and the release is anticipated in early 2010.

A website for the Serpent project was established in October 2008 at http://montecarlo.vtt.fi and it currently serves as the best and the most up-to-date description of code status and capabilities. The website also contains the most recent version of the User's Manual, descriptions of new updates, a complete list of publications, example input files and a list of recent and up-coming events related to the project.

3 Code Development

When the NEA base version 1.1.0 was released in May 2009, there were several new features being planned and under development. In addition, some existing capabilities implemented in the predecessor PSG code were waiting to be transferred into Serpent. These new features were added in various updates,

distributed directly to registered users by e-mail. The base version consisted of some 50,000 lines of C code. The size has grown to over 60,000 lines in version 1.1.8, the last update issued in 2009. The most significant new features and some major bug fixes are described below.

3.1 New features

The most significant new feature in terms of interaction physics was the implementation of probability table treatment for unresolved resonances. Prior to this improvement the code could only use smooth averaged (infinite-dilute) values in the unresolved region. This neglects all resonance self-shielding effects, which may have a noticeable impact in the results, especially in fast reactor calculations. The capability was added in update 1.1.4, released on July 26, 2009.

As a related effort, new ACE format cross section libraries with probability table data were generated at Helsinki University of Technology to replace the old library files included in the 1.1.0 installation package [4]. The new libraries were included in the updated NEA version in January 2010 and also distributed as a separate package without the source code.

Another major addition was the implementation of a built-in Doppler-broadening routine that allows cross sections to be adjusted to a higher temperature. This capability can be particularly valuable in temperature-sensitive applications, such as the calculation of reactivity feedback coefficients. The work was carried out as an M.Sc. project at Helsinki University of Technology [5]. The Doppler routine was completed in update 1.1.2, released on June 30, 2009.

The built-in burnup calculation capability was originally based on the Transmutation Trajectory Analysis method (TTA), which basically implies the analytical solution of linearized depletion chains [2]. A second routine, based on a matrix exponential solution using the Chebyshev Rational Approximation Method (CRAM) was added in early 2009. This method is an entirely new approach to the depletion problem and it was developed at VTT specifically for the Serpent code [3].

The original simplified TTA methodology was extended to account for problems related to closed transmutation chains in a student project at Helsinki University of Technology [6]. The extended methodology was included in update 1.1.2.

To extend the geometry capabilities from conventional LWR assemblies into particle-based HTGR fuels, two particle / pebble fuel models were developed for Serpent. The first, implicit model is based on the on-the-fly sampling of particle positions [7]. The incapability of this model to handle high packing fractions lead to the development of another solution. The second, explicit model reads particle or pebble positions from a separate input file and handles each object without any approximations. The model uses a special search grid for tracking the neutron inside the medium and the routine works almost as fast as tracking in a regular array. The explicit model works at several levels and it can be used for describing the particle distributions inside fuel pebbles and the distribution of pebbles in a pebble-bed type core.

When the PSG source code was re-written in 2008, some of the existing features were not transferred to the new Serpent code. One of such features is the experimental B_1 leakage model discussed in Sec. 10.2.2 of Ref. [1]. The methodology was finally added in update 1.1.5, and accompanied by another leakage correction based on the iteration of albedo boundary conditions. Both models are still considered experimental and the methodology is under development.

3.2 Major bug fixes

Several programming errors and methodological flaws were identified and corrected. Most of these bugs were not severe enough to compromise the results, although there were a few exceptions as well.

The most severe problem was related to the calculation of certain homogenized multi-group constants. This task requires dividing the integral reaction rate by the integral flux. Due to the properties of the basic collision flux estimator, the flux integral was underestimated in geometries with low-density material regions modelled as void. This, in turn, lead to the overestimation of homogenized cross sections.

The errors were in the order of a few percent in LWR fuels, in which the void regions are usually limited to the gas gap between fuel pellet and cladding. The problem becomes much more severe in CANDU and gas-cooled reactor geometries, in which the results may be off by 30%. The flaw in the methodology was corrected in update 1.1.3, released on July 13, 2009 (see the bug report at Serpent website: http://montecarlo.vtt.fi/updates/bugfix113.htm). This problem may also account for some of the differences between PSG and CASMO-4E in previous validation calculations [1].

Another major problem was related to parallel calculation in the burnup mode. Serpent uses a material-wise ultra-fine flux spectrum to speed-up the calculation of microscopic transmutation cross sections [8, 9]. The routine had a flaw which disabled this flux spectrum to be initialized to zero between burnup steps when the calculation was divided into multiple MPI tasks and several materials were burned. The result was a distortion in the distributions, which also affected the transmutation coefficients in all the following burnup steps. The problem was identified and corrected in update 1.1.6, released on September 24, 2009 (see description at Serpent website: http://montecarlo.vtt.fi/updates/update1.1.6.htm).

A few minor methodological flaws were also discovered in calculation routines handling interaction physics. Update 1.1.5 fixed a problem related to the misinterpretation of a $1/E$ dependence in certain elastic S(α,β) reaction channels. The interpolation laws of certain energy distributions were incorrectly handled at high energy. This problem was corrected in update 1.1.8.

4 Applications

The Serpent user's community has grown to some 35 registered users since the official NEA release in May 2009. The code is licensed to 20 institutes in 12 countries around the world.[1]

Although the original intended purpose was the production of homogenized multi-group constants similar for deterministic lattices transport codes, the Serpent code has been extensively used for other applications as well. A few examples of recent work carried out at different user organizations are briefly described below.

4.1 VTT Technical Research Centre of Finland

The majority of Serpent calculations carried out at VTT in 2009 have been directly related to code development and validation. Some of this work has been presented in international conferences [10-12].

The code was also used in a M.Sc. project, for producing homogenized multi-group constants for a fast reactor calculation, carried out using a new nodal diffusion method implemented in the APROS simulator code [13]. Another student project involved the use of Serpent in the calculation of an OECD / NEA burnup credit criticality benchmark [14].

4.2 Technical University of Catalonia, Spain

The code has been employed in some calculations for a M.Sc. thesis being done at the Nuclear Engineering Section of the Technical University of Catalonia, Spain.

The main goals for the work were determining the few-group homogenised constants for the fuel assemblies of a 1 GWe Spanish PWR and use them as point/nodal kinetics parameters within a Thermal-Hydraulics code, i.e. do the NK/TH coupling.

The reactor core was divided in 20 different regions involving 157 fuel assemblies and 24 axial nodes. A 3D-full core model was developed with Serpent to this purpose, although some convergence problems have arisen. The use of 2D-full core and fuel assembly models has shown preliminary better results even considering a good core detail level, including different initial assembly burnups and gadolinium rods with varying concentrations.

Some hundreds of calculations were needed in order to determine the cross section derivatives for the NK/TH feedback.

At some stages of the work, comparison with other neutronics codes have been done, although they were deterministic codes fundamentally different from Serpent, reasonable agreement have been found with Serpent results. As expected, the greater differences were observed in the diffusion coefficient estimation, varying between 1% - 8% in some cases.

[1] Argentina, Australia, Brazil, Canada, Finland, France, Germany, Italy, South Africa, Spain, Sweden and USA

Although most of the work has focused in reactor BOL conditions, some assembly and 2D-full core burnup calculations have been done and the depleted fuel isotopic compositions seemed to agree with real data.

At this moment the work has stuck in the use of the few-group constants and feedback derivatives within the Thermal-Hydraulics code.

4.3 Australian Nuclear Science and Technology Organization (ANSTO)

The Serpent code was acquired to perform burnup calculations for the OPAL research reactor. The Core consists of a square arrangement of 16 Fuel Assemblies, each comprising 21 fuel plates and 20 cadmium wires. Each fuel plate contains a plate of low-enriched uranium silicide dispersed in aluminium sandwiched between two aluminium plates. The type 2 and standard fuel assembly also has 20 cadmium wires, as shown in Figure 1. Figure 2 shows the core with the 16 Fuel Assemblies and 5 control rods. The fuel plates are separated from each other by coolant channels, which allow circulation of cooling water to remove the heat generated in the core.

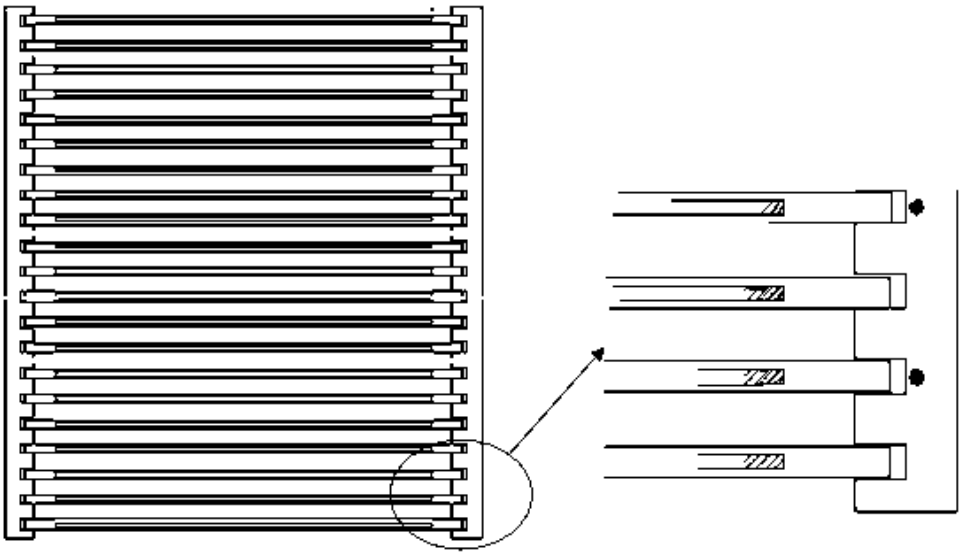


Figure 1: Cutaway view for Type 2 and Standard Fuel Assembly which consists of 21 fuel plates and 20 cadmium wires as burnable poison.

The Core is surrounded by a Reflector Vessel containing heavy water. The Reflector Vessel is cylindrical in shape with two flat ends. The diameter of Reflector Vessel is 2.6 m and the height is 1.2 m. The Reactor Pool, which contains the Core, Reflector Vessel and associated structures, is a bigger cylinder with a diameter of 9.0 m and height of 14.0 m.

Serpent OPAL model

The ideal OPAL model should be a three dimension model. To gradually approach to the final ideal model, we started from build a Fuel Assembly 2-D model, as shown in Figure 3. Then a quarter core with four fuel assemblies, but without control rods 2-D model is made (Figure4).

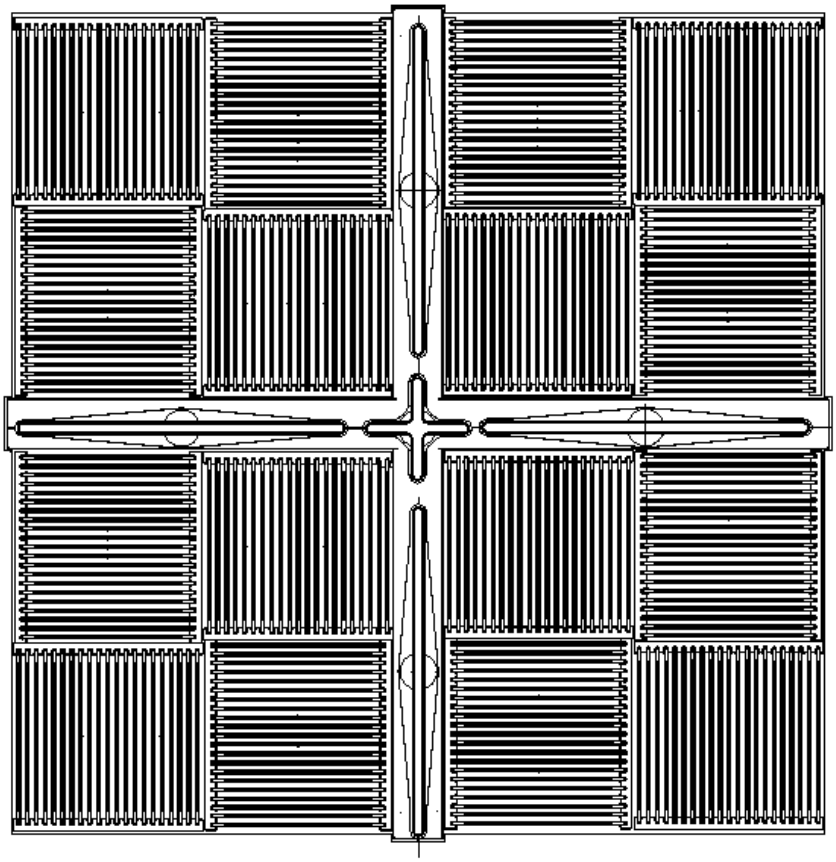


Figure 2: Reactor core

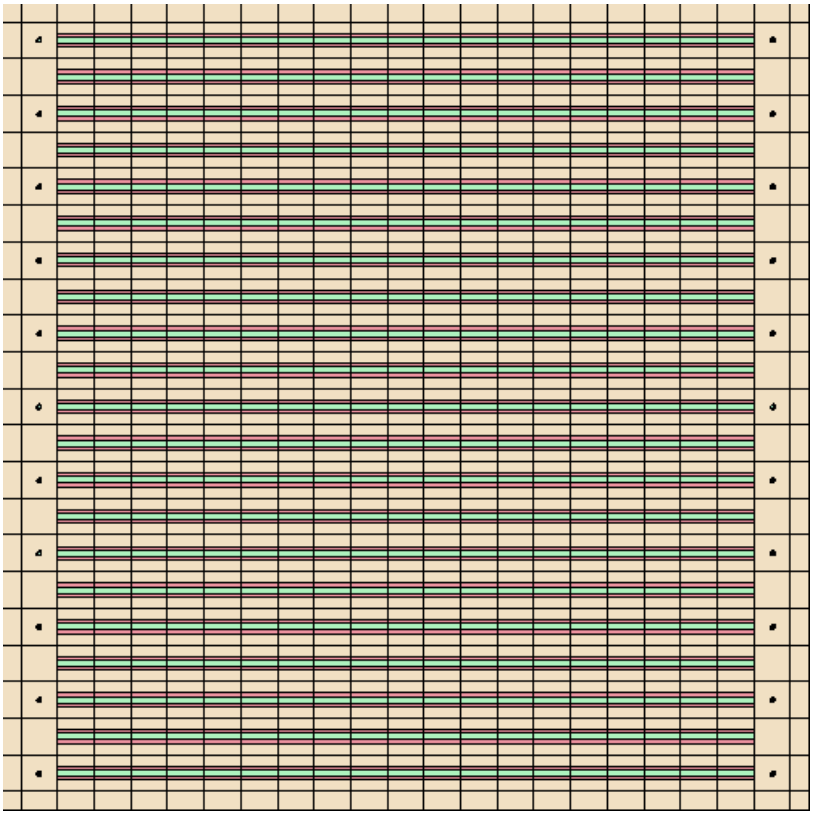


Figure 3: Serpent model for Fuel Assembly.

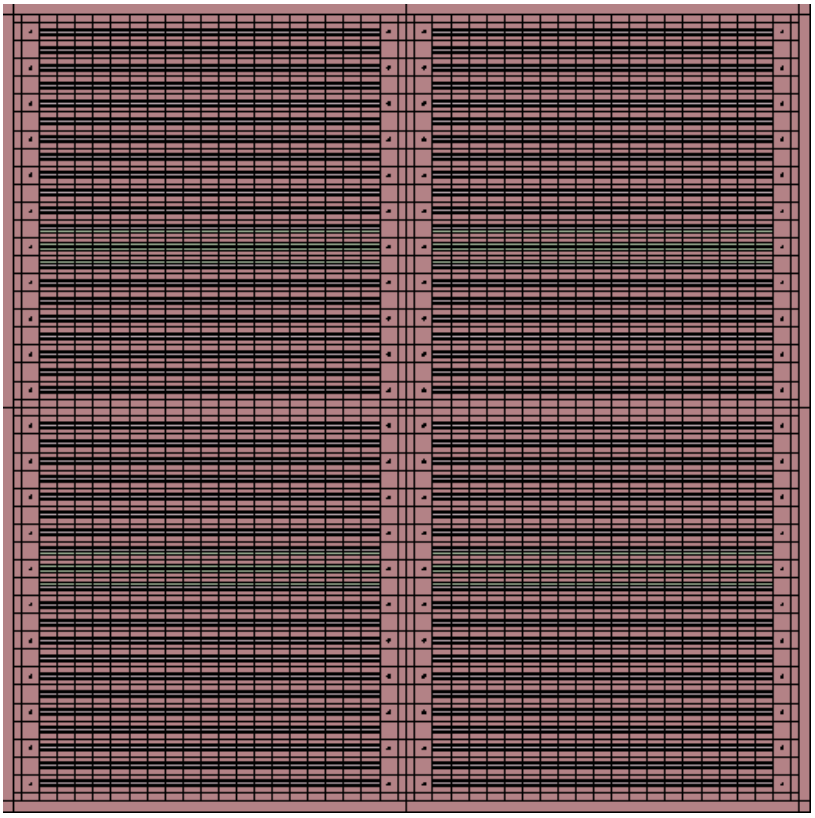


Figure 4: Serpent model for a quarter core with four fuel assemblies.

Problems

To run the basic Fuel Assembly serpent model requires 2.6 GB of allocated memory. When multi-processor version is used, the allocated memory is also multiplied. The available memory in our cluster is 8Gb, which means we can only run 3 processors.

The expected number of materials to be burnt is close to one thousand. The processing of the output file could introduce some challenges.

Future work

Step 1. Include control rod in the current serpent model.
Step 2. Build 2-D full core with 16 fuel assemblies and 5 control rods.
Step 3. Build the 3-D model for OPAL reactor core.
Step 4. Build the 3-D model of the OPAL reactor.

4.4 Lappeenranta University of Technology, Finland

Serpent was taken into use in the Laboratory of Nuclear Engineering in Lappeenranta University of Technology (LUT) during the year 2009. The code has so far been used in the research of pebble-bed type high-temperature gas-cooled reactors (HTGRs). Benchmarks based on available data from past pebble-bed type HTGR experiments are calculated. The work has included development of some new geometrical models to the code and testing HTGR specific geometry types of Serpent. There are also plans of using Serpent for educational purposes in the future.

4.5 Royal Institute of Technology, Sweden

At the Reactor Physics division of KTH (the Royal Institute of Technology, Sweden) the Serpent code has been used for research and development of advanced fast reactors and education in reactor physics on M.Sc. level.

A preliminary design study was performed for a small-scale lead-cooled fast reactor with nitride fuel (ELECTRA – European Lead-cooled Training Reactor) [15]. It allowed to find an optimal core loading scheme and provided input data for transient analysis code SAS4A, such as Doppler coefficient, coolant temperature coefficient, fuel axial expansion coefficient, coolant void worth, etc.

Within the KTH research devoted to development of the Generation VI reactors, various neutronic parameters of a model of the BN-600 core loaded with MOX fuel containing different fractions of americium were calculated with Serpent. These data was used in SAS4A for finding maximum allowable MA content in the BN-600 core [16].

Remarkable simplicity of the Serpent input and output files allowed successful application of the code in education. For example, students of the KTH Master's Programme "Nuclear Energy Engineering" [17] performed simulation of plutonium multi-recycling in a PWR and calculation of safety parameters in fast neutron energy spectra within the "Transmutation of nuclear waste" course [18]. Within the "Generation IV reactors" course [19] the master students improved breeding ratios of Gen-IV reactors by introducing innovative fuels into a fast neutron spectrum and estimated the impact of introducing americium into various fuels which may be used in liquid metal cooled reactors.

Serpent is actively used for analysis of light water reactors and liquid metal cooled fast reactors by students writing their master theses at the Reactor Physics division. For example, an optimization of safety coefficients of BWR with MA containing fuels has been recently done by Erdenechimeg Suvdantsetseg [20].

5 Future Plans

The Serpent user's community has grown fast since the code was released at the NEA Data Bank in May 2009. The code is still not widely used in North America and Asia, even though there are countries with major research programmes in the field of reactor physics. To some extent this can be accounted for by the lack of proper distribution channels. Once the code is released through RSICC, the number of users in North America is expected to start growing.

Future code development is to a large extent based on the users' needs. Some plans for short and long-term development include:

- Equilibrium xenon calculation[2]
- Implicit methods for Monte Carlo transport simulation
- Modeling of continuously-varying material compositions
- Restart features and branch calculations in burnup mode

[2] This capability was realized in the first update in 2010 (Serpent 1.1.9).

- Statistical tests for normality and convergence
- Advanced Monte Carlo leakage models
- Modeling of uncertainty propagation in burnup calculation
- Continuous-energy adjoint calculation
- Methods for sensitivity and uncertainty analysis
- Improved detector features
- Fixed-source calculation mode
- Photon and coupled neutron-photon transport simulation
- Voxel-based geometry models
- Extending the capabilities from reactor physics to radiation shielding and dosimetry calculations and medical physics applications

Code validation and documentation, especially the description of the calculation methods, are seriously lagging behind. New updates are validated by running a set of standard test problems where the results are compared to reference MCNP calculations. These results will be made available at the website within the near future.

Communication between users and developers has so far been carried out via e-mail. Once the user's community reaches a "critical mass", other means of communication, such as an on-line discussion forum need to be considered. Feedback from the users has been a valuable resource and all interactive communication is encouraged in the future as well.

6 Summary and Conclusions

Year 2009 was the first in the history of Serpent development during which the code was widely used outside VTT. Public distribution began in May, when Serpent 1.1.0 was released at the OECD / NEA Data Bank. At the end of the year, the code was licensed to 20 organizations around the world. Distribution to North America via RSICC is anticipated in early 2010.

The code has been extensively updated since the original release and the last update in 2009 contained 10,000 lines of new code. In addition to entirely new capabilities, these updates contained a number of bug fixes, some of which had a considerable impact in the results.

The Serpent code was originally designed to produce homogenized multi-group constants for reactor simulator calculations, similar to conventional deterministic lattice transport codes. It has turned out, however, that the range of applications has been extended well beyond traditional lattice physics calculations.

Considerable contribution to both code development and application can be accounted to various student projects in different universities. In fact, the majority of active code users are probably undergraduate and Ph.D. students working on reactor physics problems that are difficult to approach using general-purpose Monte Carlo codes. This suggests that Serpent has so far filled its expectations, and hopefully new users in the future will bring some interesting new applications and challenges as well.

Acknowledgements

Descriptions of recent work outside VTT were provided by Jose Luis Modrego (Technical University of Catalonia, Spain), George Braoudakis (ANSTO, Australia), Ville Rintala (Lappeenranta University of Technology) and Andrei Fokau (Royal Institute of Technology, Sweden).

References

[1] J. Leppänen. *"Development of a New Monte Carlo Reactor Physics Code."* D.Sc. Thesis, Helsinki University of Technology (2007). VTT Publications 640.

[2] J. Cetnar. *"General solution of Bateman equations for nuclear transmutations."* Ann. Nucl. Energy, **33** (2006) 640-645.

[3] M. Pusa and J. Leppänen. *"Computing the Matrix Exponential in Burnup Calculations."* Nucl. Sci. Eng., **164** (2010) 140-150.

[4] T. Viitanen and J. Leppänen. *"Cross Section Libraries for Serpent 1.1.7."* Document available at Serpent website (http://montecarlo.vtt.fi).

[5] T. Viitanen. *"Implementing a Doppler-Preprocessor of Cross Section Libraries in Reactor Physics Code Serpent."* M.Sc. Thesis, Helsinki University of Technology (2009).

[6] A. Isotalo. *"Improved TTA methods for the reactor physics code Serpent."* Special assignment, Helsinki University of Technology (2009).

[7] J. Leppänen. *"Randomly Dispersed Particle Fuel Model in the PSG Monte Carlo Neutron Transport Code."* In Proc. M&C + SNA 2007. Monterey, California, April 15-19 2007.

[8] W. Haeck and B. Verboomen. *"An Optimum Approach to Monte Carlo Burnup."* Nucl. Sci. Eng., **156** (2007) 180-196.

[9] E. Fridman, E. Shwageraus and A. Galperin. *"Efficient Generation of One-Group Cross Sections for Coupled Monte Carlo Depletion Calculations."* Nucl. Sci. Eng., **159** (2008) 37-47.

[10] J. Leppänen and M. Pusa. *"Burnup Calculation Capability in the PSG2 / Serpent Monte Carlo Reactor Physics Code."* In Proc. M&C 2009. Saratoga Springs, NY, May 3-7 2009.

[11] J. Leppänen. *"On the use of the continuous-energy Monte Carlo method for lattice physics applications."* In Proc. INAC 2009. Rio de Janeiro, Brazil, Sept. 27 - Oct. 2 2009.

[12] J. Leppänen and M. DeHart. *"HTGR Reactor Physics and Burnup Calculations Using the Serpent Monte Carlo Code."* Trans. Am. Nucl. Soc. **101** (2009) 782-784.

[13] P. Inkinen. *"Implementation of nodal diffusion model into APROS."* M.Sc. Thesis, Helsinki University of Technology (2009).

[14] P. Juutilainen. *"Burnup Credit Benchmark Calculation with Monte Carlo Method."* Special assignment, Helsinki University of Technology (2009). VTT-R-06819-09.

[15] J. Wallenius and A. Fokau, *"A Low Power Fast Reactor Cooled by Natural Circulation of LBE"*, In Proc. GLOBAL'2009, Paris, France, September 6-11, 2009. http://sfen.omnibooksonline.com/global/data/papers/265_9150.pdf

[16] Y. Zhang et al. To be submitted.

[17] http://neutron.kth.se/study/masters

[18] http://neutron.kth.se/courses/Transmutation.shtml

[19] http://neutron.kth.se/courses/GenIV.html

[20] Erdenechimeg Suvdantsetseg, Optimization of safety coefficients in BWR transmutation. M.Sc. Thesis, Division of Reactor Physics, KTH (2009).