1st International Serpent User Group Meeting in Dresden, Germany, September 15-16, 2011

Discussion notes

The first international Serpent user group meeting was held at the Helmholtz-Zentrum Dresden-Rossendorf in Dresden, Germany, September 15-16, 2011. The meeting brought together 33 Serpent users from 15 Organizations around the world. The first day of the meeting focused on applications and methods used in the publicly available Serpent 1. The second day began with an introduction to Serpent 2, currently developed under working title "Super-Serpent", followed by presentations related to new features and capabilities.

The afternoon of the second day was reserved for discussion related to methods and applications, as well as some new capabilities suggested by the users. The following is a summary on six specific topics.

1. Group constant generation in non-fissile zones

Presentations from HZDR and BGU pointed out the fact that the generation of diffusion coefficients and assembly discontinuity factors cannot be carried out for non-fissile regions, such as reflectors, using the presently-available methodology. What is missing in particular, is the capability to calculate leakage currents over the reflector boundaries. This fundamental flaw is mainly due to the use of the Woodcock delta-tracking method, which cannot account for surface crossings while transporting the neutrons through the geometry.

Two possibilities were suggested and discussed:

1. Calculation of net current between two universes, based on the beginning and end points of each neutron track. This method should be easy to implement, without additional cost in CPU time, but it comes with certain limitations regarding multiple and re-entrant universe regions.
2. Conventional surface current tally on super-imposed surfaces, i.e. additional surfaces that are not a part of the geometry construction. This type of tally could be used for the purpose without any limitations, with the additional capability to account for inward and outward currents separately. The implementation requires more effort compared to the first option, and may result in a slight increase in the overall calculation time.

It was decided to add both methods in the detector capability. Since the calculation of discontinuity factors and diffusion coefficients requires a few additional steps, detailed instructions with examples will be added in the User's Manual and posted at the Serpent discussion forum. The possibility of implementing the methodology as an automated feature will be evaluated later.

It was also recognized that the code is missing the capability to define partially reflected geometries, in which one of the boundary conditions is black. This flaw is also due to the use of the delta-tracking method, and the fact that boundary conditions are handled by coordinate transformations instead of actual reflections. This type of geometry definition would be useful in reflector calculations, and the possibility of introducing partially reflective boundary conditions will be studied.

2. Production runs

Generation of homogenized group constants for reactor simulator calculations requires repeating the infinite-lattice calculation for every fuel assembly type in all state points within the reactor core. The number of calculation cases increases to several hundred, and managing the production runs and all the input and output data becomes a major challenge, which in practice requires some level of automation.

Presentation of the work carried out at MIT introduced a python script called SerpentXS, developed for this task. It was decided to make the script available to all users at the Serpent website. The author of SerpentXS is no longer working with the project, and the following issues related to future development and maintenance of the script were pointed out:

- There is no manual, but the author's M.Sc. Thesis contains some description with examples. This documentation will be made available to the users in some form.
- The script was written for Serpent / PARCS calculations, and other couplings will require customizing the output data format.
- The script was written for some specific computing environment, and configuration in another system may require additional work.

It was also pointed out that the script was written for the present version of the Serpent code, and new features, such as restart and source read/write capability will most likely change the way the calculations are best organized. This is also the case for the new methods used for parallelization.

Several important parameters were noted missing from the Serpent output. The following parameters will be added in the next update:

- Parameters for I-135 and Xe-135 production
- Decay heat divided into several precursor groups
- Multi-group form factors (relative pin-wise fission energy production in few-group structure)

3. Multi-physics

Coupling of Monte Carlo neutronics to thermal hydraulics and / or fuel performance codes is a hot research topic, also addressed in presentations from BGU and LUT. This topic covers a wide range of applications, each with its characteristic methods and requirements. The main purpose of the discussion was to identify the common needs for the construction of a universal multi-physics interface that could be used to couple Serpent with any TH or fuel performance code.

Regardless of the particular application, it can be assumed that the coupling is based on a wrapper program that handles the input and output and executes the codes in the appropriate order. The parameters that need to be passed into Serpent include:

- Geometry, material compositions and standard input
- Coolant temperature, and density or void distribution
- Fuel temperature

The parameter calculated by Serpent include:

- Power distribution
- Burnup

Some of the parameters (isotopic compositions in input, burnup in output) must be related to homogeneous material regions, defined with the geometry. Density distributions, however, could be defined by laying a super-imposed structure on top of the actual geometry. This type of approach should simplify the data flow by eliminating the need to divide the geometry into several material regions with different density. The implementation of on-the-fly temperature treatment should extend the capability to temperatures as well. Several options for super-imposed structures were discussed, including:

- Structured mesh (Cartesian, hexagonal, cylindrical, spherical)
- Unstructured mesh
- Point-wise distribution with linear interpolation

Apart from the point-wise distribution with interpolation, similar data structures could be used for output as well. Since neutronics and TH meshing are generally different, the corresponding interface parameters should be given separately as well. Developing coupled multi-physics capabilities is a long-term project, and this interface will be eventually implemented in Serpent 2.

It was also noted that pebble-bed reactor geometries are so fundamentally different from any other reactor configuration, that the same type of interface is probably not applicable. Presentation from LUT discussed the coupling of Serpent into porous-medium fluid dynamics calculations. The explicit stochastic geometry model in Serpent allows the accurate description of the pebble-bed core, and the calculation of pebble-wise power distribution. The input (pebble distribution) and output (pebble-wise power distribution) are read and written in separate files. Since coolant temperature has no impact on neutronics, and fuel and moderator temperatures can be easily given for each pebble, this type of interface may turn our to be sufficient for more elaborate pebble-bed multi-physics calculations as well.

4. Parallelization

The parallelization in Serpent 2 will be based on hybrid MPI / OpenMP techniques. Shared-memory parallelization (threading) using OpenMP has already been implemented, and some test results were presented in the introductory talk. The test calculations showed only modest scalability by a factor of 4 using 12 CPU cores. The poor performance is most likely not related to OpenMP barriers required in the transport cycle, since the scalability is equally poor or even worse for burnup and processing routines, where the possibility of writing in the same variable is not a problem.

The following issues were pointed out in the discussion:

- The executable used in the test calculations was compiled using an old version (4.1.2) of gcc. More recent versions or other compilers might produce better results.
- The scalability seems to be hardware-dependent, and similar tests with a 2 CPU Macbook Pro typically result in a speed-up factor of 1.8 or even higher.
- Finding information on poor scalability is difficult, due to the specific nature of the calculation task. Monte Carlo neutron transport simulations are extremely memory-intensive, while most of the other problems in scientific computation tend to be more or less CPU-intensive.
- Similar tests using MCNP might reveal something about the nature of the problem, especially if similar scalability is observed with same hardware.

It was concluded that there is not much that can be done in terms of systematic testing until the release of the beta-version of Serpent 2.

5. Sensitivity and uncertainty analysis

NRG presented their Total Monte Carlo (TMC) method for uncertainty analysis of cross sections, using Serpent as the neutron transport and burnup solver. In addition to this stochastic approach, S/U analysis can be performed using deterministic methods, which will be a major topic for research and Serpent development in the future. Near-term goals for this task include the implementation of the Iterated Fission Probability (IFP) method, developed at LANL for the calculation of the fundamental adjoint function in forward Monte Carlo simulation. Additional research topics include adjoint-weighed detectors and variance reduction techniques based on neutron / photon importance with respect to detector response.

6. Scripts

Conversion and utility scripts is another area where efforts could be shared over the entire user community, and the possibility of setting up a common script repository was briefly discussed. It was mentioned that conversion scripts between Serpent and MCNP input files exist at least at INL and KTH. The union operator, missing from Serpent 1, will be available in Serpent 2. This addition will simplify the geometry conversion, but it will also require some changes in the structure of the conversion scripts.

The interactive plotter script developed at KTH was briefly discussed. It was recognized that the script is an extremely useful and easy-to-use tool, but because it was written in the Ruby scripting language,

not included by default in most Linux distributions, several users had run into problems with installation. The author is no longer working at KTH, and the future of the script remains unclear.

Summary on new features

The efforts in Serpent development are currently divided between the publicly available Serpent 1, and Serpent 2, scheduled for beta-testing by the end of 2011. All major new features will be implemented in Serpent 2, and the development of version 1 is limited to bug fixes and minor modifications. The following table summarizes the new capabilities, brought up during the discussion.

Feature	**Will be implemented into**
Surface current detectors	1.1.17
Partially reflective boundary conditions	1.1.17
Restart capability	Super
Source read / write capability	Super
I-135 and Xe-135 production rates	1.1.17
Decay heat in several precursor groups	1.1.17
Multi-group form factors	1.1.17
Universal multi-physics interface	Super