Business from technology

Serpent Monte Carlo Neutron Transport Code

NEA Expert Group on Advanced Monte Carlo Techniques, Meeting September 17 2012

Jaakko Leppänen / Tuomas Viitanen

VTT Technical Research Centre of Finland

Outline

- Brief history
- Methods and capabilities:
  - Physics in a nutshell
  - Depletion routine
  - Group constant generation
  - HTGR capabilities
  - Typical applications
- Project organization and funding
- User community
- Serpent 2

Brief history

- The development of Serpent started in September 2004, under the working title “Probabilistic Scattering Game”, or PSG
- Topic for a D.Sc. thesis (completed in 2007)[1]
- Built-in depletion routines and public distribution in 2008-2009

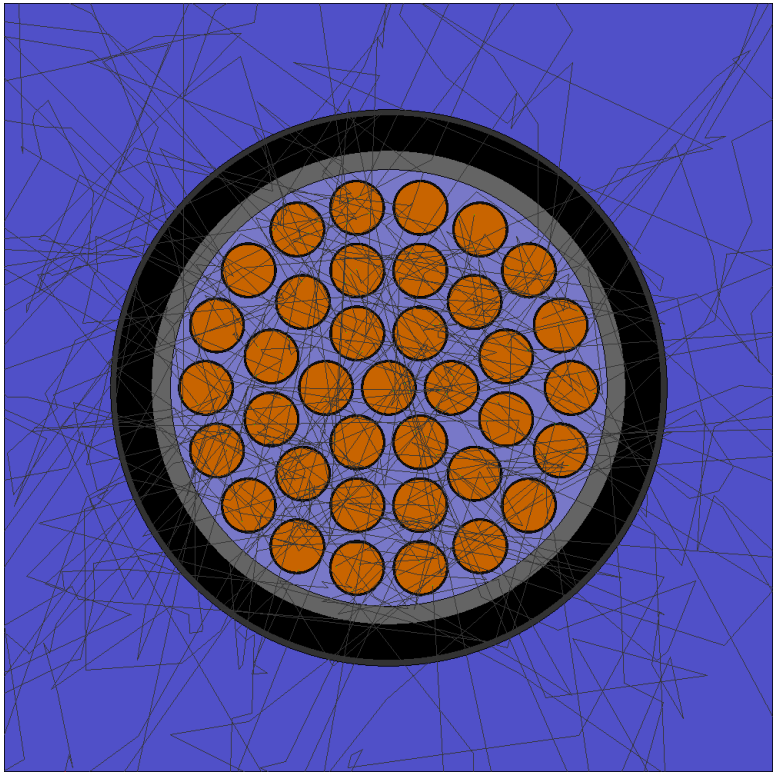
Serpent can be characterized as a three-dimensional continuous-energy Monte Carlo reactor physics code, that is specialized but not limited to lattice physics applications

- Re-writing of source code (Serpent 2) started in October 2010

[1)] J. Leppänen, *“Development of a New Monte Carlo Reactor Physics Code.”* D.Sc. Thesis, Helsinki University of Technology, 2007.

Overview

- Three-dimensional geometry model based on surfaces, cells and universes (similar to MCNP and KENO-VI)
- Additional geometry types for reactor physics calculations:
  - “Nests”: cylindrical fuel pin and spherical particle
  - 3 basic lattice types with variations
  - Explicit model for stochastic HTGR geometries



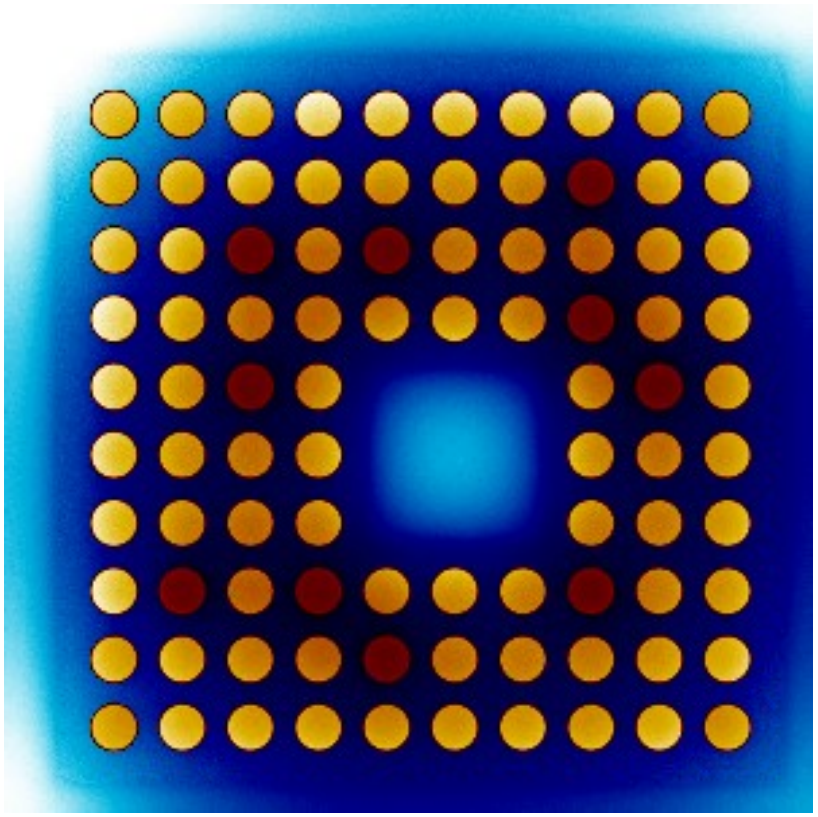
- Criticality and external source modes for neutron transport simulation (Serpent 2 also tracks photons)

Physics in a nutshell (Serpent 1)

- The "laws of physics" are based on ACE format data libraries, with a few additional tricks:
  - Cross sections reconstructed on a unionized energy grid → speed-up in calculation but increase in memory demand[2]
  - DBRC for resonant scattering kernel
  - Built-in Doppler broadening preprocessor routine[3]
- Geometry routine based on combination of surface-tracking and the Woodcock delta-tracking method[4] → well-suited for reactor calculations but poor performance in detector modeling

[2] J. Leppänen. *"Two Practical Methods for Unionized Energy Grid Construction in Continuous-Energy Monte Carlo Neutron Transport Calculation."* Ann. Nucl. Energy, **36** (2009) 878-885.
[3] T. Viitanen. *"Implementing a Doppler-Preprocessor of Cross Section Libraries in Reactor Physics Code Serpent."* M.Sc. Thesis, Helsinki University of Technology (2009).
[4] J. Leppänen. *"Performance of Woodcock Delta-Tracking in Lattice Physics Applications Using the Serpent Monte Carlo Reactor Physics Burnup Calculation Code."* Ann. Nucl. Energy, **37** (2010) 715-722.

Depletion routine (Serpent 1)



- Built-in depletion routines for tracking nuclide concentrations as the fuel is burnt:
  - Radioactive decay and fission yield data read from ENDF format files
  - Decay / transmutation chains generated automatically without user effort
  - Bateman depletion equations solved using the CRAM matrix exponential method, developed at VTT specifically for Serpent[5]
  - Typical depletion calculation tracks the concentrations of ~1500 nuclides
  - Optional equilibrium Xe-135 calculation

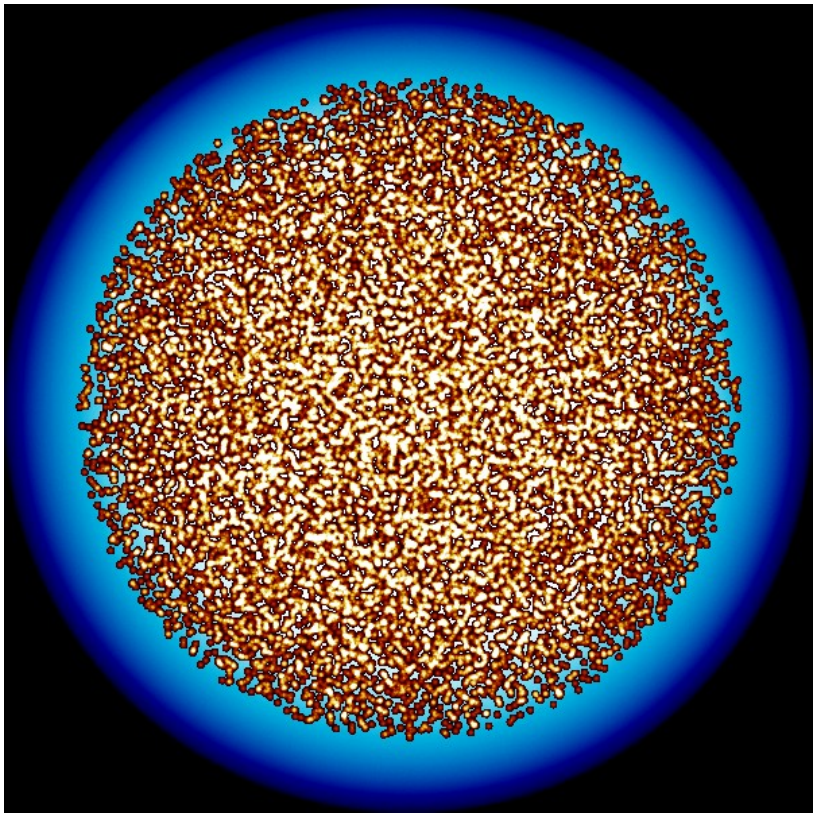
[5] M. Pusa and J. Leppänen,. *"Computing the Matrix Exponential in Burnup Calculations."* Nucl. Sci. Eng., **164** (2010) 140-150.

Group constant generation (Serpent 1)

- Serpent has the capability to automatically produce all input parameters needed for few-group nodal diffusion calculations:
  - Homogenized multi-group cross sections and diffusion coefficients
  - Scattering matrices
  - Point-kinetic parameters (forward flux-weighed)
  - Assembly discontinuity factors
  - Pin-wise power distributions
  - Effective delayed neutron fractions (”Meulekamp's method”)
- A semi-deterministic method for homogenizing group constants in critical spectrum (B1 fundamental mode calculation)[6]

[6] E. Fridman and J. Leppänen. *"On the Use of the Serpent Monte Carlo Code for Few-group Cross Section Generation."* Ann. Nucl. Energy, **38** (2011) 1399-1405.

HTGR calculation (Serpent 1)

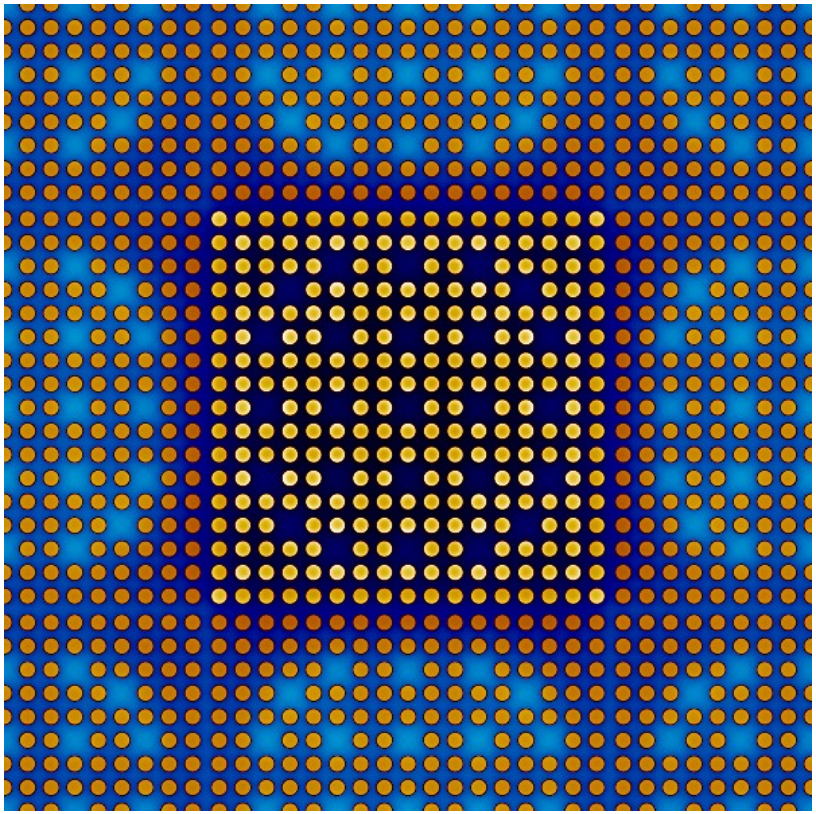


- The delta-tracking method works exceptionally well in HTGR particle fuels
- Explicit model for stochastic geometries:
  - Coordinates of spherical objects are read from a separate input file
  - Works at several levels (TRISO particles in graphite / fuel pebbles in pebble-bed core)
  - Tested in full-core pebble-bed reactor calculations (ASTRA experiment), no major increase in calculation time compared to a regular-lattice calculation[7]
  - Pebble-wise power distribution printed in a separate output file

[7)] H. Suikkanen et al., 2010 *"An Approach for Detailed Reactor Physics Modelling of Randomly Packed Pebble Beds."* In Proc. HTR 2010. Prague, Czech Republic, Oct. 18-20 2010.

Typical applications

- Serpent is specifically developed for:
  - Generation of homogenized multi-group constants
  - Fuel cycle studies and depletion calculations at the fuel assembly level
- But also:
  - Research reactor applications
  - Validation of deterministic transport codes
  - Educational purposes



- Examples of user applications: http://montecarlo.vtt.fi/download/VTT-R-05444-12.pdf

Project organization and funding

- The “Serpent team” at VTT:

  Jaakko Leppänen
  Maria Pusa
  Tuomas Viitanen
  Ville Valtavirta (starting 10/2012)

- Funding from three projects:
  - SAFIR-2014 (National safety research programme): Four-year project (KÄÄRME) started in 2011, total budget of 955 k€
  - HPMC (EU): Three-year project started in 2011, total budget of 209 k€
  - NUMPS (Academy of Finland): Four-year project, started in 2012,
  - total budget 1.3 M€

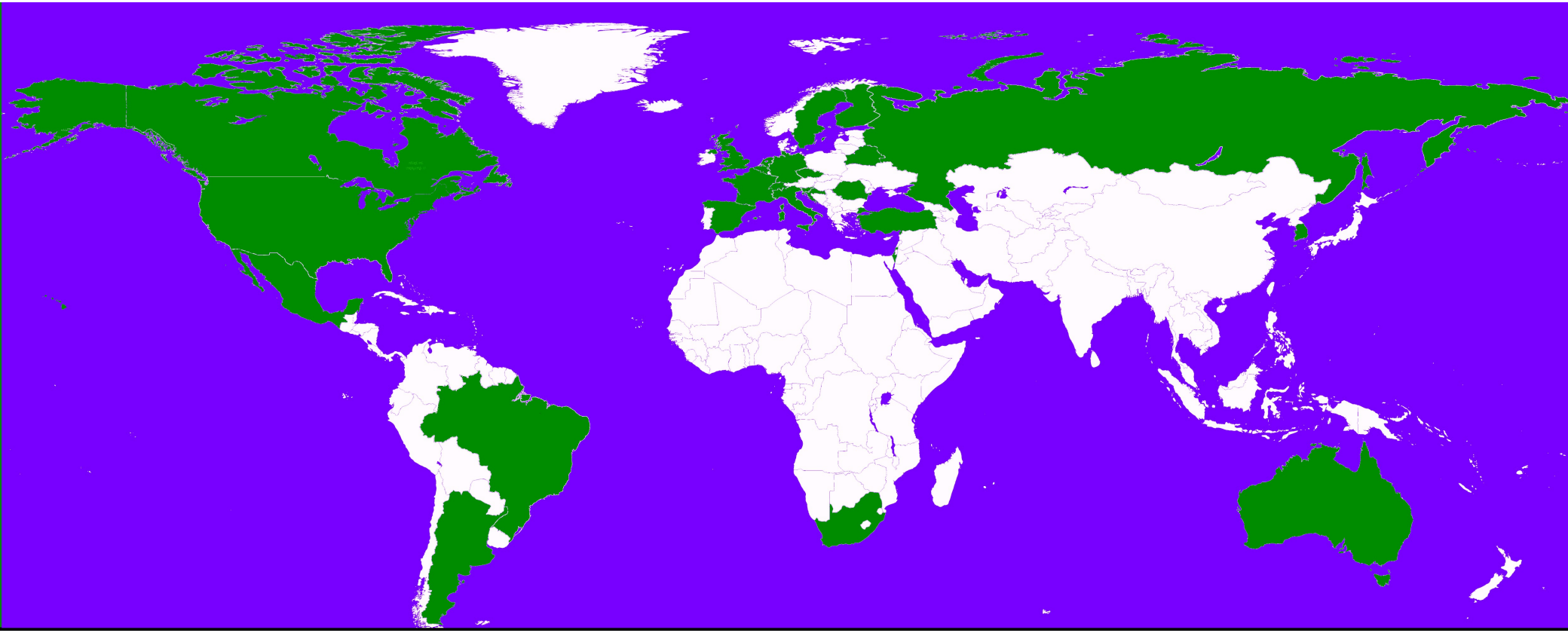
User community

- Serpent is used in 76 organizations in 27 countries around the world, number of registered users is around 180
- Serpent website and discussion forum:

  http://montecarlo.vtt.fi
  http://ttuki.vtt.fi/serpent

- Distribution of Source code by OECD/NEA Data Bank and RSICC
- Updates are distributed to registered users by e-mail (current update 1.1.18, distributed August 16, 2012)
- Second International User Group Meeting in Madrid, Spain, this week (September 19-21, 2012).

The Serpent code – user community



Serpent 2

- The main focus at the moment is the development of Serpent 2, started in October 2012:
  - Removing memory limitations in Serpent 1 and extending burnup capability to ~100,000 depletion zones (done)
  - Parallelization using hybrid OpenMP / MPI techniques insted of just MPI (done)
  - Photon and coupled photon / neutron transport routines mainly for the purpose of reactor physics and shielding applications (started)
  - Development of variance reduction techniques (not yet started)
  - Multi-physics applications (started, major focus from 2013 on)
- Serpent 2 beta testing started in January 2012 (~50 users)
- Public distribution is scheduled for 2013-2014