

PSG2 / Serpent

a Monte Carlo Reactor Physics Burnup Calculation Code

Jaakko Leppänen



Business from technology

Outline

- Background
- History
- The Serpent code:
 - Neutron tracking
 - Physics and interaction data
 - Burnup calculation
 - Output
 - Results
 - Performance
- Code distribution and maintenance

Background

- Monte Carlo codes traditionally used for criticality safety analyses, shielding and dosimetry calculations, detector modeling and validation of deterministic transport codes.
- "New" applications:
 - Burnup calculation
 - Homogenization and group constant generation
- Existing general-purpose codes not particularly well-suited for the above tasks.
 - ⇒ Development of a dedicated Monte Carlo reactor physics code

History

- Development of Serpent code started in 2004, under the working title "Probabilistic Scattering Game", or PSG.
- A related doctoral thesis published in 2007:
 - Available on-line at: www.vtt.fi/inf/pdf/publications/2007/P640.pdf
 - Currently the best available documentation of code methodology
- 120 subroutines, 40000 lines of C-code. Source code written from scratch (re-written in 2005 and 2007).
- Burnup calculation capability in 2008.
- Name changed to "Serpent" and public release in October 2008 (pre-release of version 1.0.0).
- Current version 1.0.3 distributed to 4 institutes outside VTT.
- NEA release in late 2008 / early 2009.

Code overview

- Serpent is a continuous-energy Monte Carlo reactor physics burnup calculation code.
- Specifically designed for two-dimensional lattice physics calculations (geometry not restricted to 2D).
- Applications:
 - Generation of homogenized few-group constants for deterministic reactor simulator calculations
 - Fuel cycle studies involving detailed assembly-level burnup calculations
 - Validation of deterministic lattice transport codes
 - Reactor physics calculations at pin, assembly and core levels
 - Educational purposes and demonstration of basic reactor physics phenomena
- Parallel calculation using MPI (both transport and depletion calculation parallelized).

Neutron tracking

- Three-dimensional universe-based geometry model.
- Analog Monte Carlo game.
- K -eigenvalue criticality source method \Rightarrow applications limited to multiplying systems.
- Capability to treat soluble absorber concentration as a free variable \Rightarrow iteration of critical boron concentration.
- Neutron transport based on a combination of conventional ray-tracing and the Woodcock delta-tracking method:
 - Calculation routine efficient in lattice calculations
 - Combination of two tracking methods overcomes efficiency problems encountered with delta-tracking with localized heavy absorbers
 - Integral reaction rates calculated using the collision flux estimator \Rightarrow code poorly suited for detector calculations

Physics and interaction data

- Continuous-energy cross sections read from ACE format data:
 - Data format shared with MCNP \Rightarrow validation by direct comparison to reference results
 - Cross sections reconstructed on a unionized master energy grid used for all nuclides:
 - \Rightarrow Considerable increase in efficiency
 - \Rightarrow Computer memory wasted for storing redundant data points
 - Interaction physics modeled according to classical collision kinematics and ENDF reaction laws
 - Bound-atom scattering libraries for important moderators
 - No probability table treatment for unresolved resonances

Burnup calculation

- Burnup calculation in stand-alone and coupled modes.
- Transmutation cross sections calculated from flux spectrum after transport cycle:
 - A major speed-up compared to direct calculation (factor of 3 to 4)
 - Main energy grid structure used in spectrum calculation to maximize resolution
- Bateman equations solved using the TTA method:
 - Analytical solution of linearized chains
 - Closed chains terminated (e.g. successive (n,γ) and $(n,2n)$ reactions)
- An advanced matrix exponential solution under development.

Burnup calculation

- Optional predictor-corrector calculation.
- Radioactive decay and fission yield data read from raw ENDF format data libraries.
- Number of depletion zones not restricted (available memory may become a limiting factor).
- Irradiation history defined in units of time or burnup.
- Reaction rates normalized to power, power density, flux or fission rate.
- Normalization and moderator boron concentration can be changed during the cycle.

Output

- All numerical output written in matlab m-format files.
- Serpent calculates by default:
 - Effective multiplication factors using different methods
 - Homogenized few-group cross sections
 - Group-transfer probabilities and scattering matrices
 - Diffusion coefficients using two fundamentally different methods
 - P_n scattering cross sections up to order 5
 - Assembly discontinuity factors (surface and corner)
 - Assembly pin-power distributions
 - Point-reactor kinetic parameters
 - Physical and effective delayed neutron parameters
 - Parameters for the six-factor formula
- Few-energy group structure defined by user.

Output

- Fission source entropy calculated automatically (total and x-, y- and z-components).
- User-defined detectors (Tallies) for calculating volume-integrated reaction rates:
 - Integration domain defined by a combination of cells, universes, lattices and materials or using a super-imposed 3D mesh
 - Arbitrary energy bin structure
 - Various response functions (material total and isotopic reaction cross sections, ACE format dosimetry data)
 - Differential and integral flux and reaction rate spectra
- Geometry and thermal flux / fission rate plotter producing png-format graphical output.

Output

- Output in burnup calculation :
 - Isotopic compositions
 - Transmutation cross sections
 - Activities
 - Decay heat data
- All results given as both material-wise and total values.
- Group constants and other output printed for each step.

Results

- Code extensively validated by comparing to MCNP results in LWR lattice calculations:
 - Differences generally within the range of statistical accuracy
- Promising results in burnup calculation:
 - K-eff and Depletion of U-235 and Gd consistent with CASMO-4E calculations
 - Minor over-prediction of Pu-buildup (library effect?)
- Validation of burnup calculation routines complicated. Comparisons to other Monte Carlo codes (Monteburns, MCB, ABURN) needed.

Results

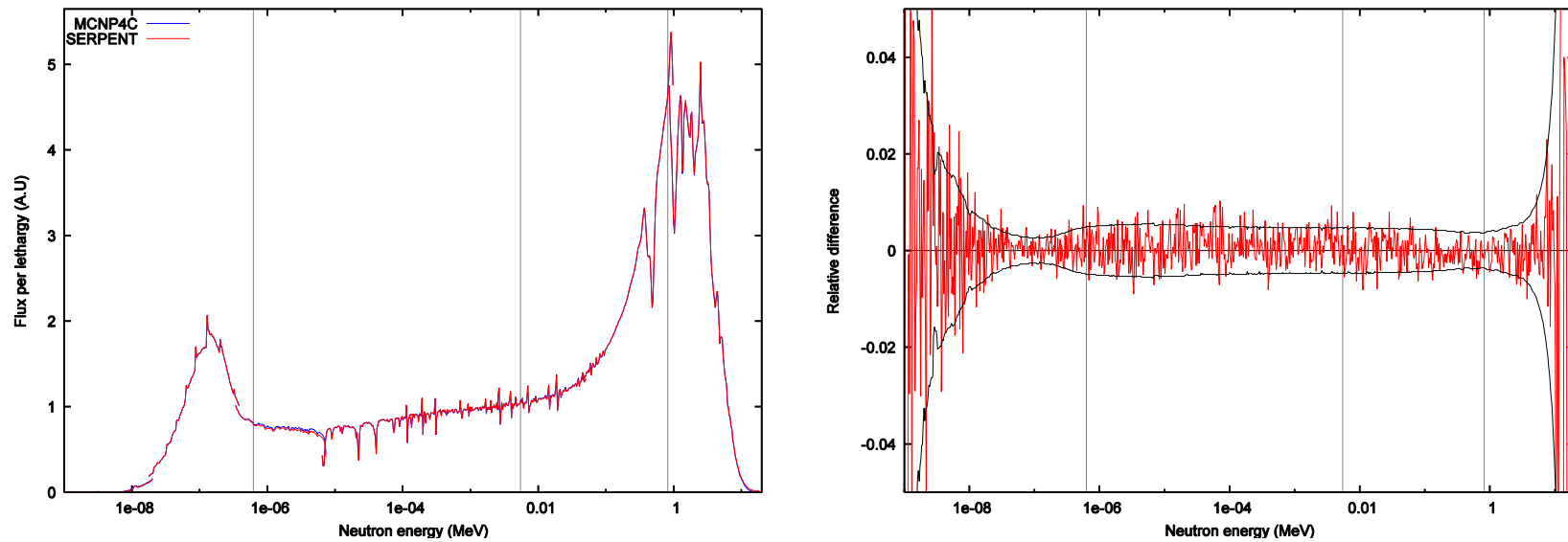


Figure 1: Comparison of LWR flux spectra.

Results

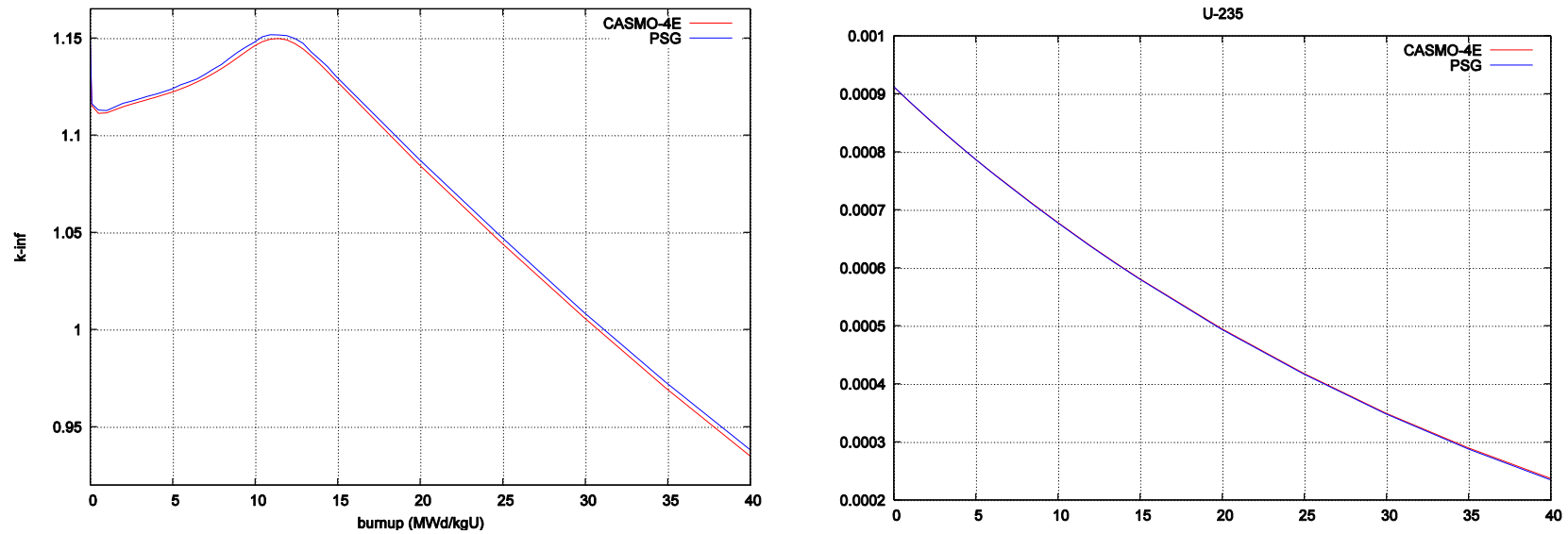


Figure 2: Comparison of k_{eff} and U-235 concentration in PWR assembly burnup calculation. Serpent vs. CASMO-4E.

Results

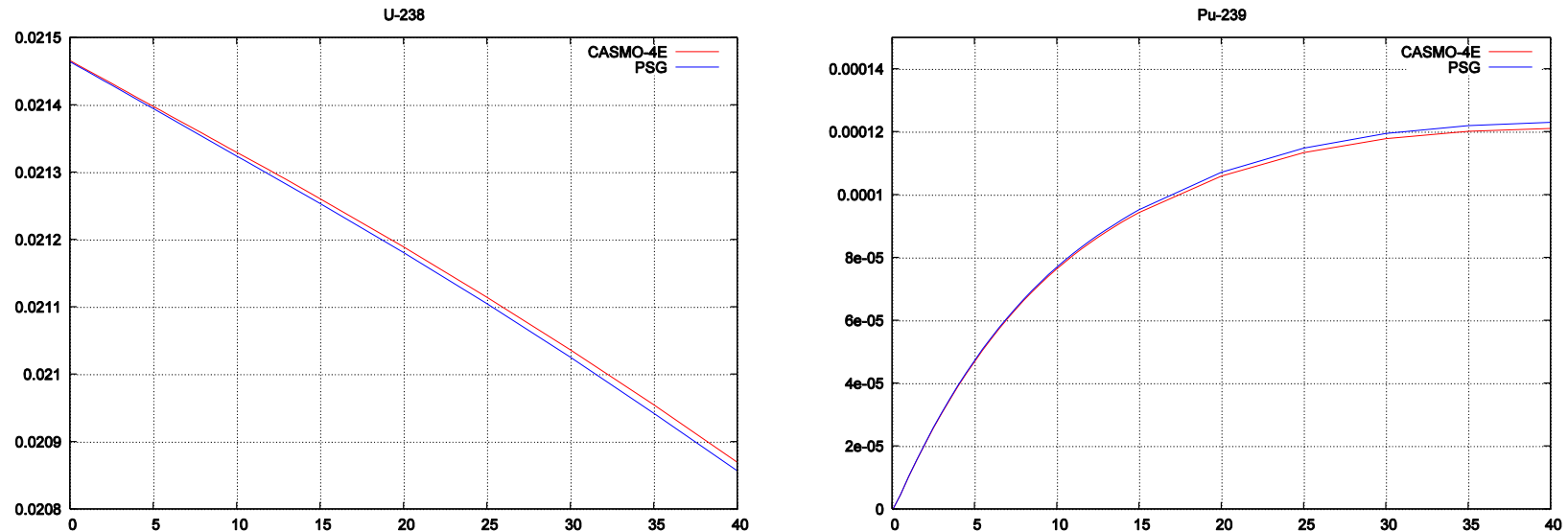


Figure 3: Comparison of U-238 and Pu-239 concentrations in PWR assembly burnup calculation. Serpent vs. CASMO-4E.

Results

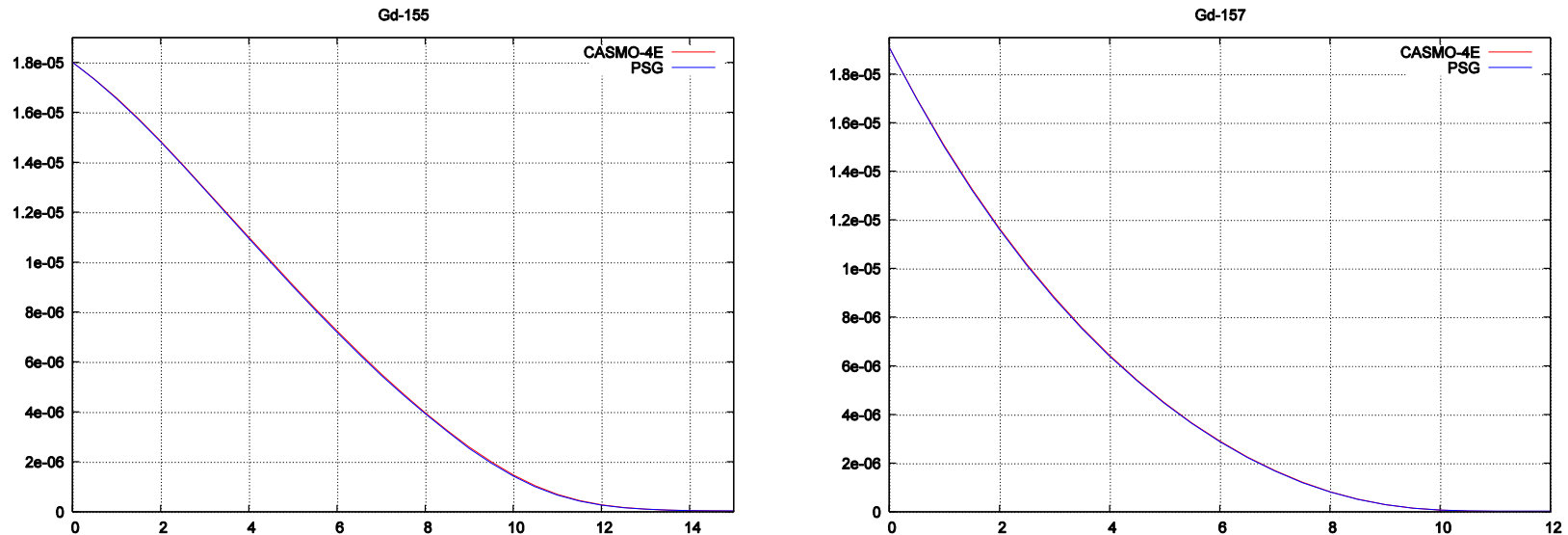


Figure 4: Comparison of Gd-155 and Gd-157 concentrations in PWR assembly burnup calculation. Serpent vs. CASMO-4E.

Results

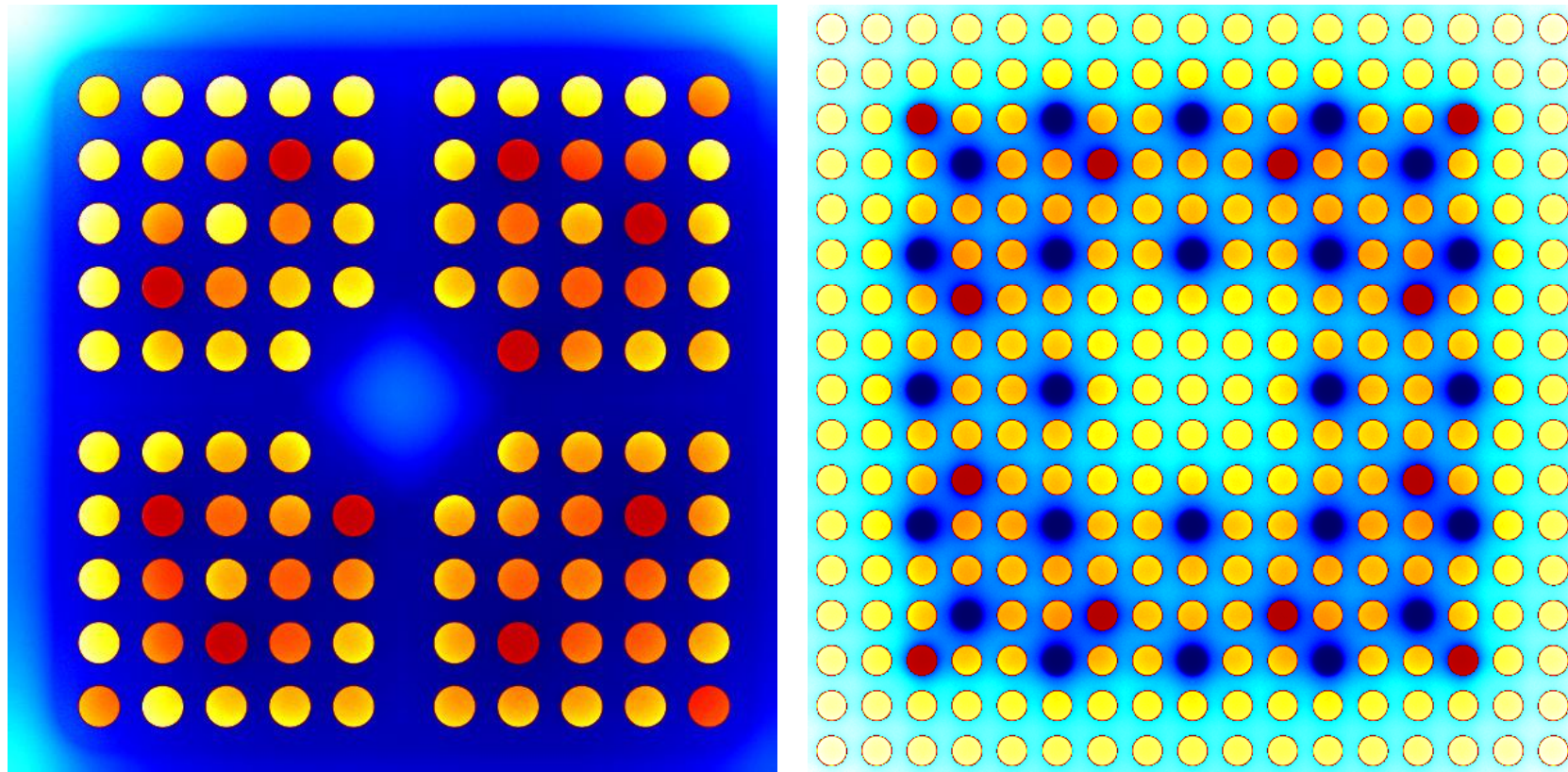


Figure 5: Thermal flux and power distributions.

Results

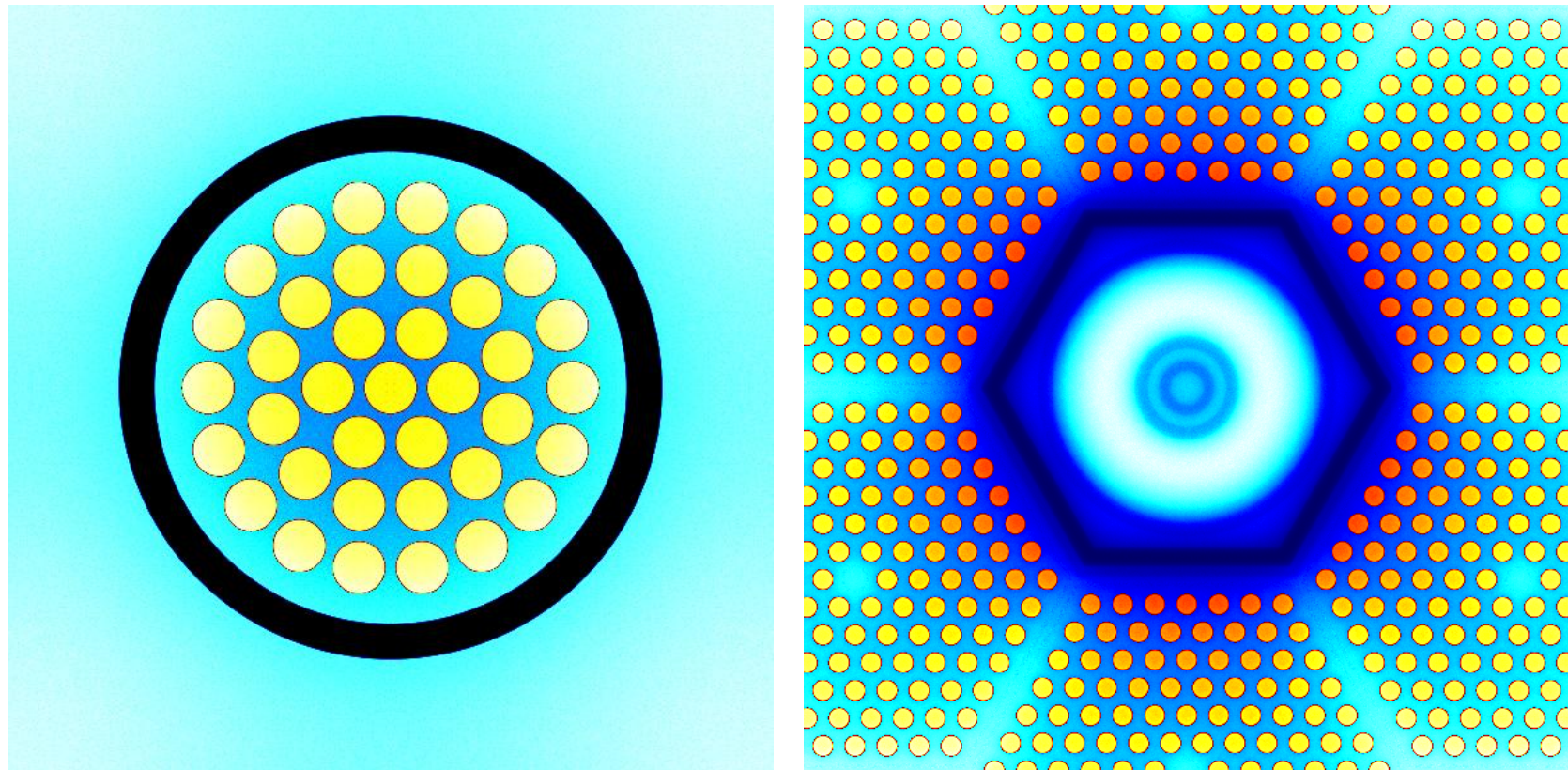


Figure 6: Thermal flux and power distributions.

Results

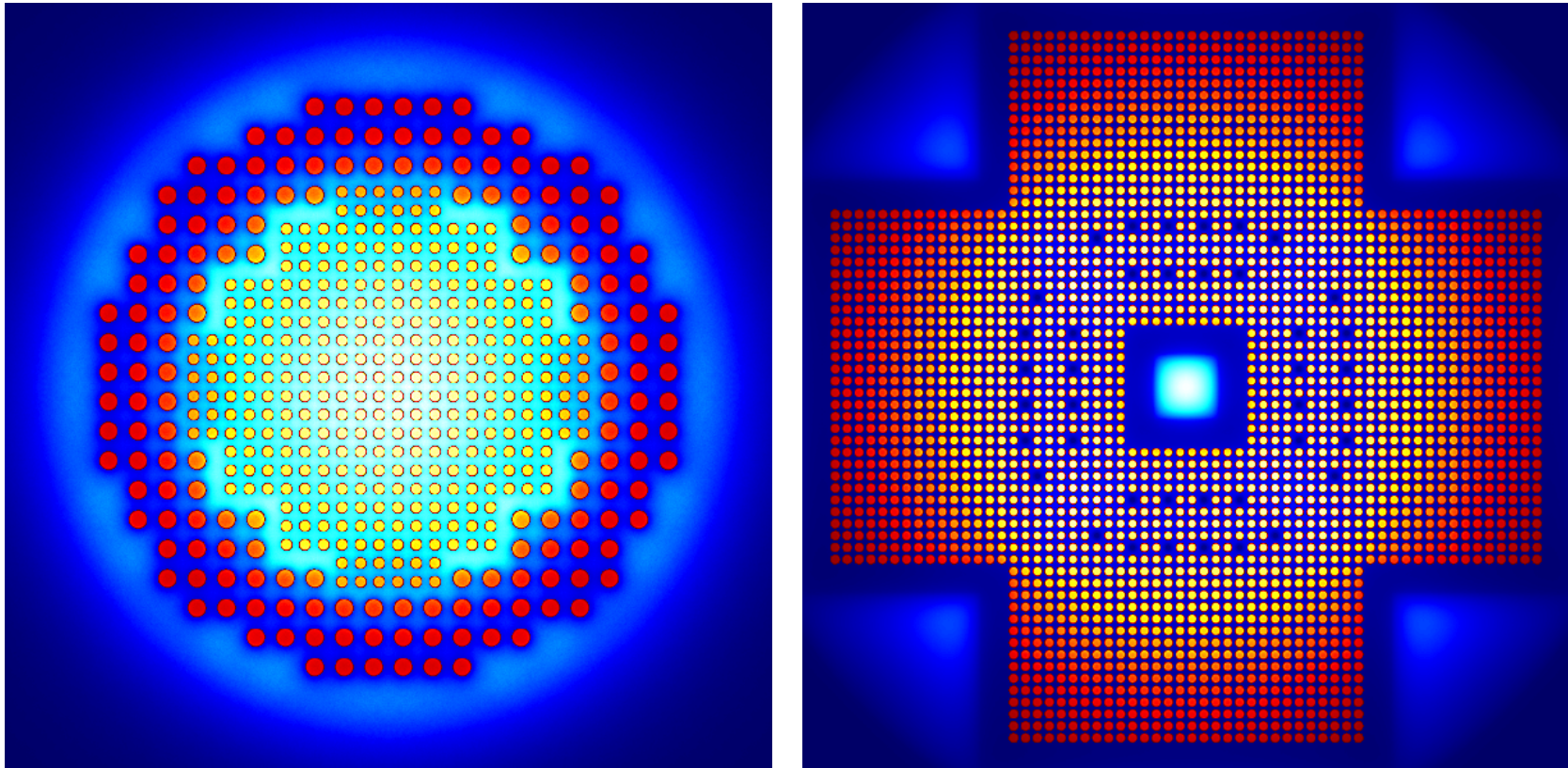


Figure 7: Thermal flux and power distributions.

Performance

- Serpent is optimized for performance in infinite-lattice calculations.
- Typical running time on a 2.6 GHz AMD Optron PC varies from 10 to 20 minutes (3 million neutron histories).
- Comparison to MCNP (with the corresponding reaction rate tallies): Serpent runs 5-15 times faster.
- Code performance results from two factors:
 1. Combined "conventional" and delta-tracking method for neutron transport
 2. Unionized energy grid for all nuclides
- Overall calculation time not strongly dependent on the number nuclides.

Performance

- Overall running time in burnup calculation scales linearly with respect to the number of depletion zones.
 - Comparison to Monteburns (PWR assembly burnup calculation, 65 depleted materials, 37 burnup steps with predictor-corrector calculation, 3 million neutron histories):
 - Running time of MCNP is strongly dependent on the number of nuclides and tallies.
 - Running time of Serpent is not strongly dependent on the number of nuclides.
 - Calculation of transmutation cross sections does not significantly increase the running time.
- ⇒ Serpent runs over 80 times faster than Monteburns!
- ⇒ The overall running time is counted in hours or days, rather than weeks or months.

Distribution and maintenance

- Free software licence for non-commercial research and educational purposes.
- Source code distribution via OECD/NEA Databank or directly from VTT.
- Minor updates in source code distributed to registered users by e-mail.
- Serpent website: <http://montecarlo.vtt.fi>
 - User's manual
 - General description
 - Etc...

Related publications

- J. Leppänen. *Diffusion Code Group Constant Generation Using the Monte Carlo Method*. In Proc. *XII Meeting on Reactor Physics Calculations in the Nordic Countries*. Halden, Norway, May 17-18 2005.
- J. Leppänen. *A New assembly-level Monte Carlo neutron transport code for reactor physics calculations*. In Proc. *M&C 2005*. Avignon, France, Sept. 12-15 2005.
- J. Leppänen. *Current Status of the PSG Monte Carlo Neutron Transport Code*. In Proc. *PHYSOR-2006*. Vancouver, BC, Canada, Sept. 10-14 2006.
- 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.
- J. Leppänen. *Development of a New Monte Carlo Reactor Physics Code*. D. Sc. Thesis, Helsinki University of Technology (2007). VTT Publications 640.

Related publications

- J. Leppänen. *PSG Status Report (March 2008)*. VTT-R-02580-08. VTT Technical Research Centre of Finland (2008).
- J. Leppänen. *On the feasibility of a homogenised multi-group Monte Carlo method in reactor analysis*. In Proc. *PHYSOR-2008*. Interlaken, Switzerland, September 14-19 2008.
- J. Leppänen. *On the Calculation of Reactor Time Constants Using the Monte Carlo Method*. In Proc. *IYNC-2008*. Interlaken, Switzerland, September 20-26 2008.