Calculating Neutron Dosimeter Activation in VVER-440 Surveillance Chains Using New Variation Reduction Techniques of Serpent 2.1.27

<u>Tuomas Viitanen</u>

Serpent User Group Meeting

Milan, Italy

Sep. 26 – 29, 2016

Outline

- Introduction: pressure vessel surveillance of Loviisa 1
- Calculation model
- Results

Introduction (1/2)

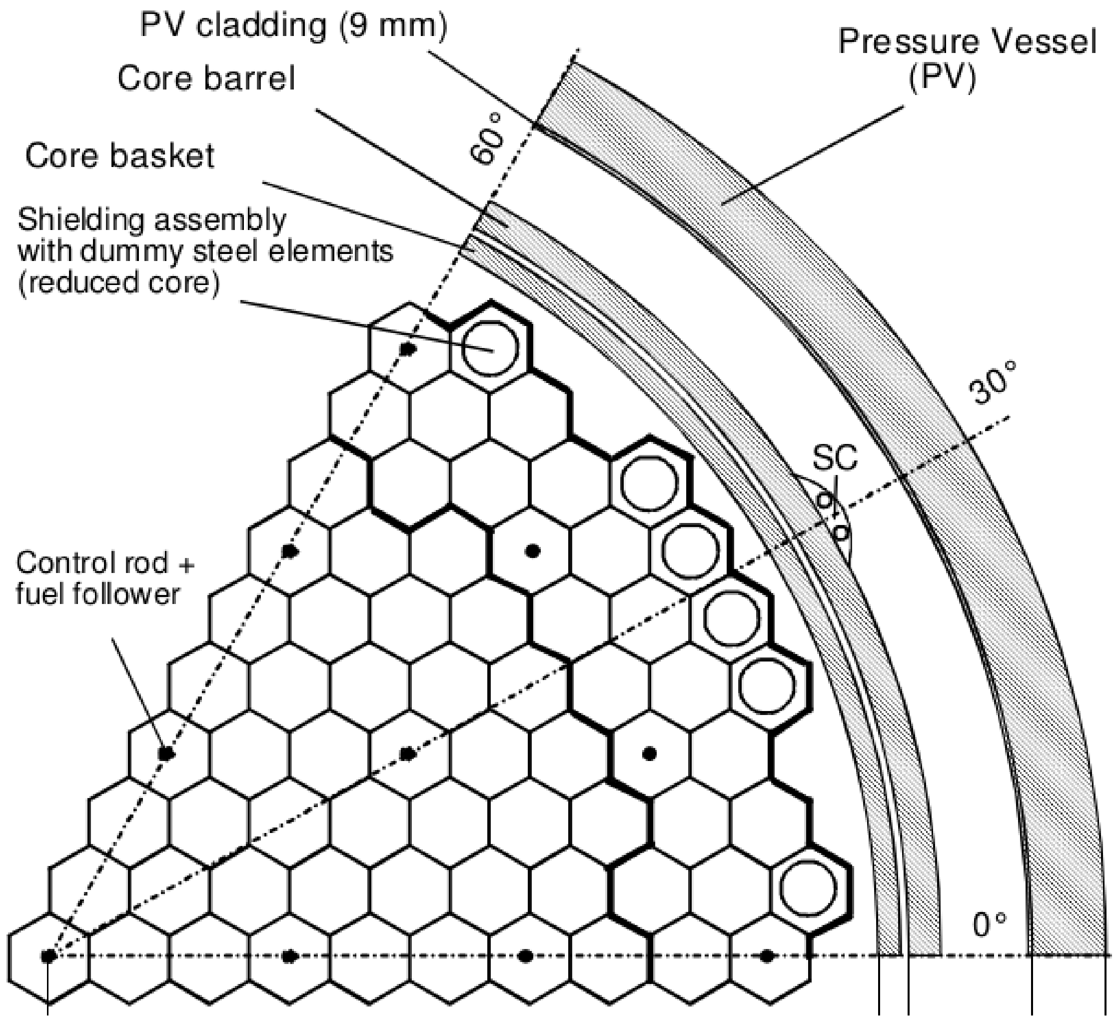
- Exposure to (fast) neutrons causes embrittlement of RPVs.
  $\rightarrow$ The structural integrity of reactor pressure vessels (RPVs) has to be monitored.
- The surveillance programs involve:
  1. irradiation of surveillance specimens,
  2. measuring the material properties of the specimens and
  3. matching the measurements to the state of the RPV by determining the neutron exposures (specimens and RPV)

3. requires solution of the neutron flux outside the reactor

Introduction (2/2)

- Solution of neutron flux outside reactor core is a bit cumbersome:
  - Core simulators based on diffusion are not designed to solve neutron flux outside the reactor.
  - Full-core Monte Carlo solution of neutronics (+ collecting statistics far from core) requires a lot of CPU time
    $\rightarrow$ Flux is often solved using deterministic transport codes (e.g. TORT @ VTT)
- To decrease uncertainties in the neutron flux, the surveillance specimens are irradiated together with **neutron dosimeters**.

Case Loviisa-1 (VVER-440)



Surveillance chains of Loviisa

- Surveillance samples (RPV steel)
- Temperature monitors
- **Neutron dosimeters:**
  - Fe/Ni discs
  - 6 dosimeter wires (Fe,Ni,Cu,Ti,Nb,Co)

(Confidental) measurement data is available!

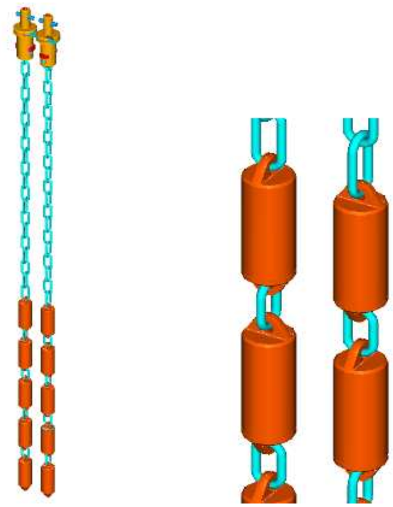


Figure 1: Surveillance capsules are attached to each other with links to form chains.

Cross section responses for dosimeters

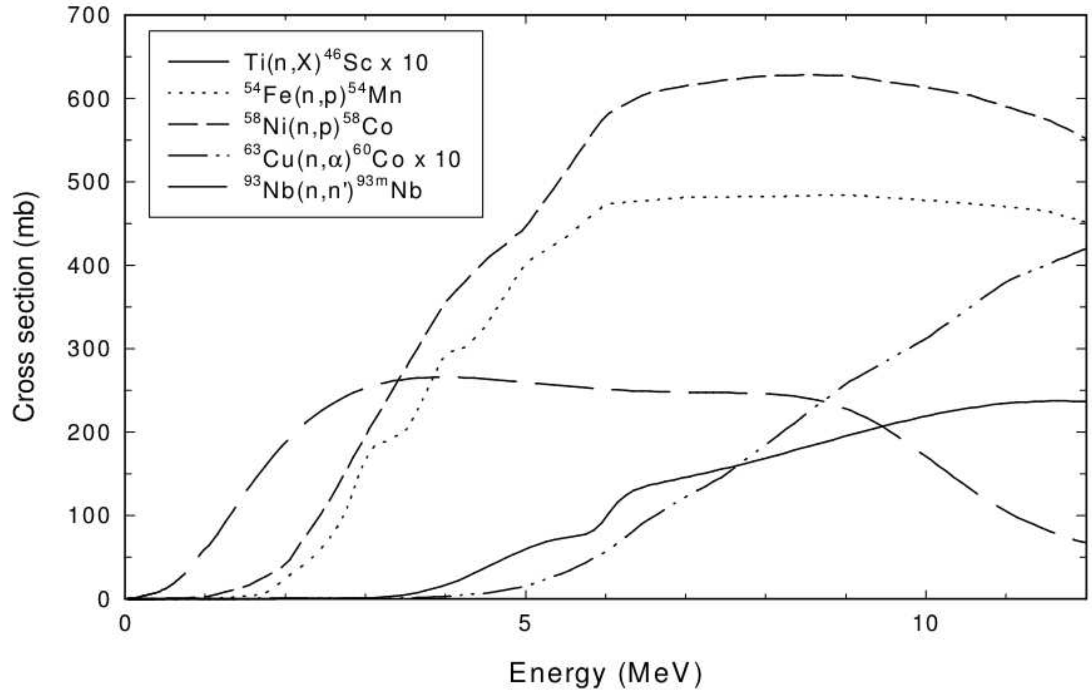


Figure 2: With multiple neutron dosimeters, integral reaction rate data is obtained for different energy responses. Energy thresholds of typical dosimetry reactions vary between 0.8–4.7 MeV.

Two-step approach to flux solution

- Simultaneuously solving (full-core 3D) neutronics and flux at surveillance position is out of question.
  $\rightarrow$ Generate full-core neutron source based on nodal code (HEXBU) calculation and import in Serpent

- Point-wise neutron sources are first generated in 5 MWd/kgU intervals with Serpent (2D lattice).
- For each calculation node (53×10):
  - pick the neutron source closest to the average burnup of the node (from HEXBU)
  - reject points according to the ratio of the power density in the node (from HEXBU) to the maximum power density.

Calculation scheme

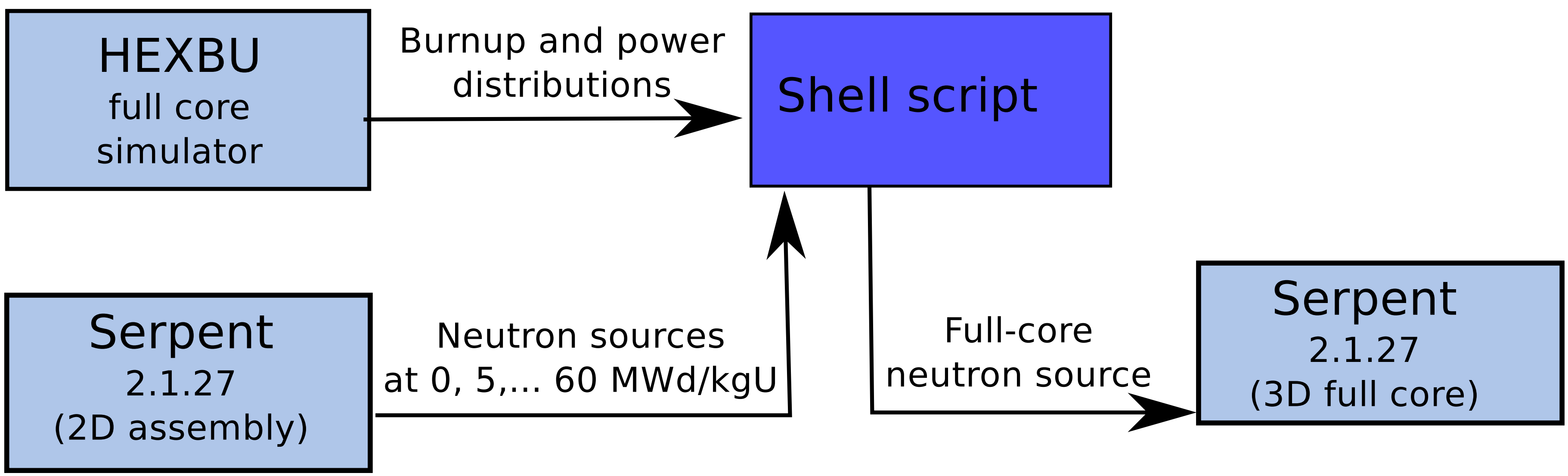


Figure 3: A shell script is used to generate a 3D (point-wise) neutron source for the final Serpent calculation.

Inaccuracies in the Serpent model

- Fuel assembly data taken from public sources
- T and ρ are constant for all materials
- All material compositions taken from public sources
- Axial source profile is piece-wise constant (within each node)
- All fuel is fresh in the transport calculation.
- etc...

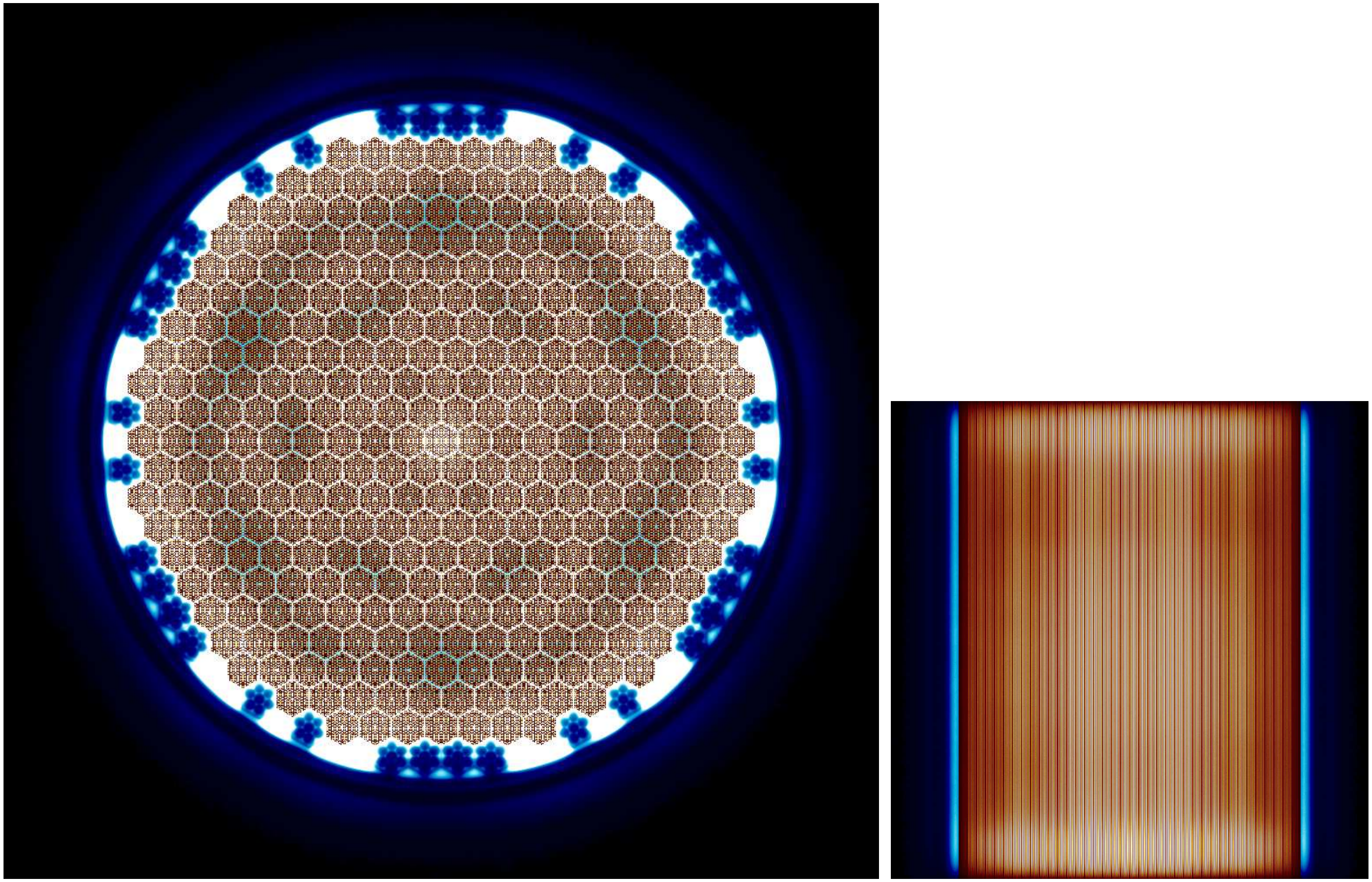


Figure 4: Neutron histories are distributed evenly within the core.

First attempt

3.75 Billion histories, 12 CPUs, Calculation time 18.7 h

Response	Threshold	$t_{1/2}$	Std.dev.	FOM
^{54}Fe(n,p)^{54}Mn	2.2 MeV	312 d	35 %	0.44 1/h
^{58}Ni(n,p)^{58}Co	1.9 MeV	71 d	29 %	0.64 1/h
^{63}Cu(n,α)^{60}Co	4.7 MeV	5.3 y	67 %	0.12 1/h
$^{\mathrm{nat}}$Ti(n,X)^{46}Sc	3.7 MeV	84 d	65 %	0.13 1/h
^{93}Nb(n,n')$^{93\mathrm{m}}$Nb	0.8 MeV	16 y	14 %	2.56 1/h
^{59}Co(n,γ)^{60}Co	-	5.3 y	10 %	5.93 1/h
^{93}Nb(n,γ)^{94}Nb	-	20300 y	19 %	1.54 1/h
^{58}Fe(n,γ)^{59}Fe	-	45 d	11 %	4.74 1/h

Getting all std. deviations below 1 % would require about 100 CPU years.

Calculation scheme

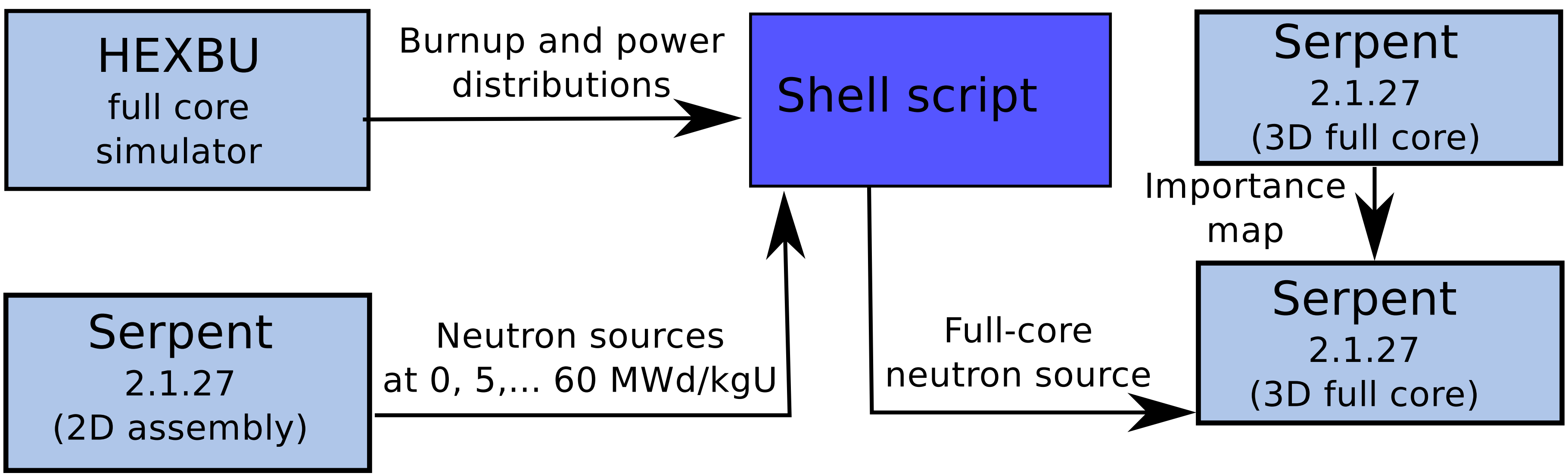


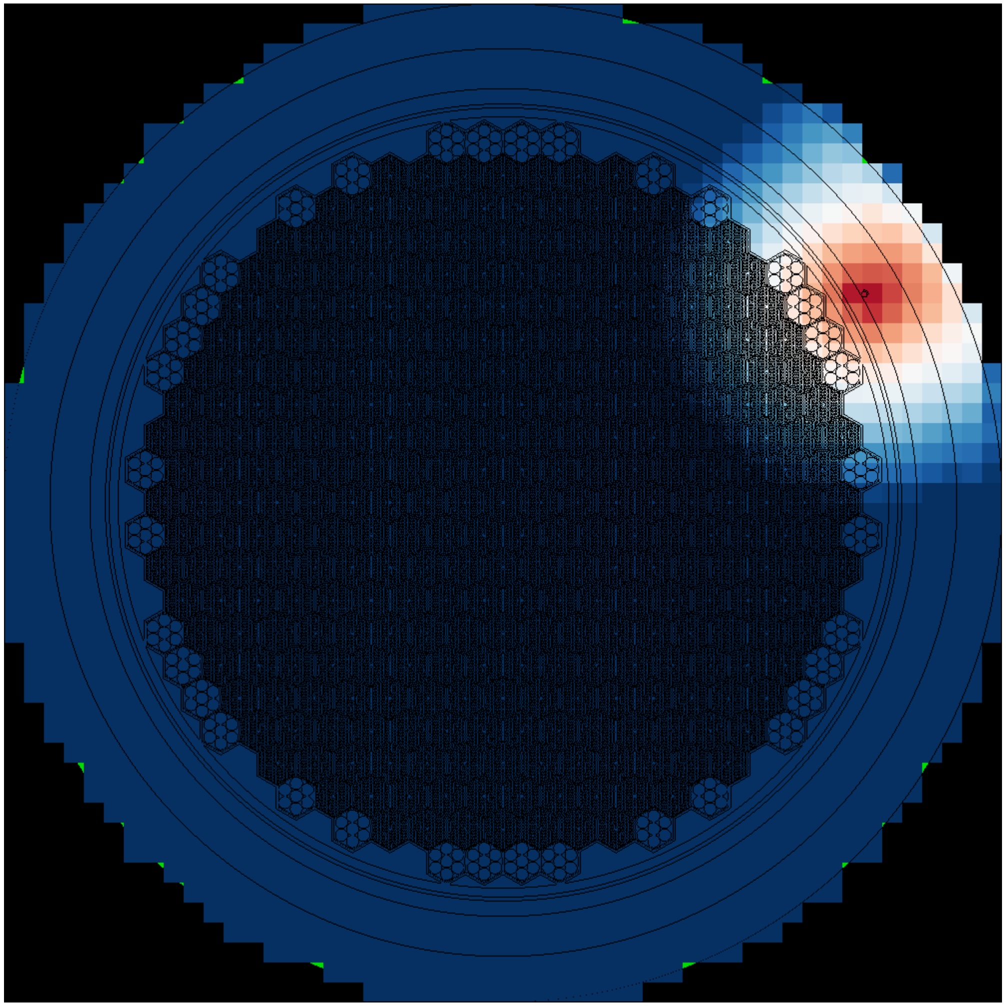
Figure 5: The importance map for weight-window based variance reduction is calculated in a separate calculation.

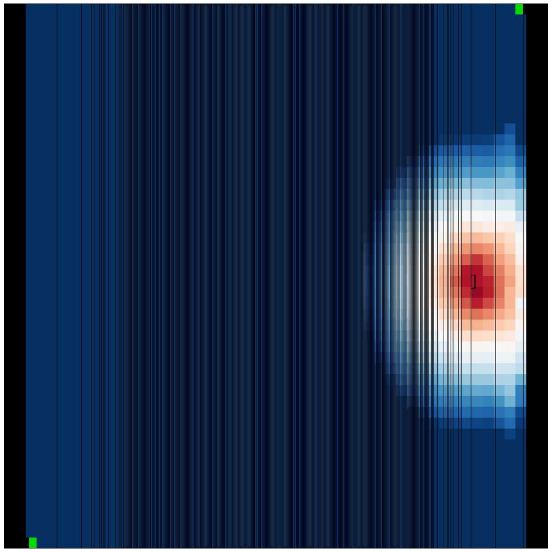
Serpent inputs

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%% Importance map generation input %%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Define target response detector

det F1 dc 771

% Call response matrix based importance map
% generator

rmtx    % nowadays called "wwgen", check Wiki!
1000 -1 % max iterations, no energy dependence
1       % Cartesian mesh
-195 195 50 % Bounding coordinates for mesh
-195 195 50 % and number of mesh cells
-128 122 50
F1 % Target response detector
```

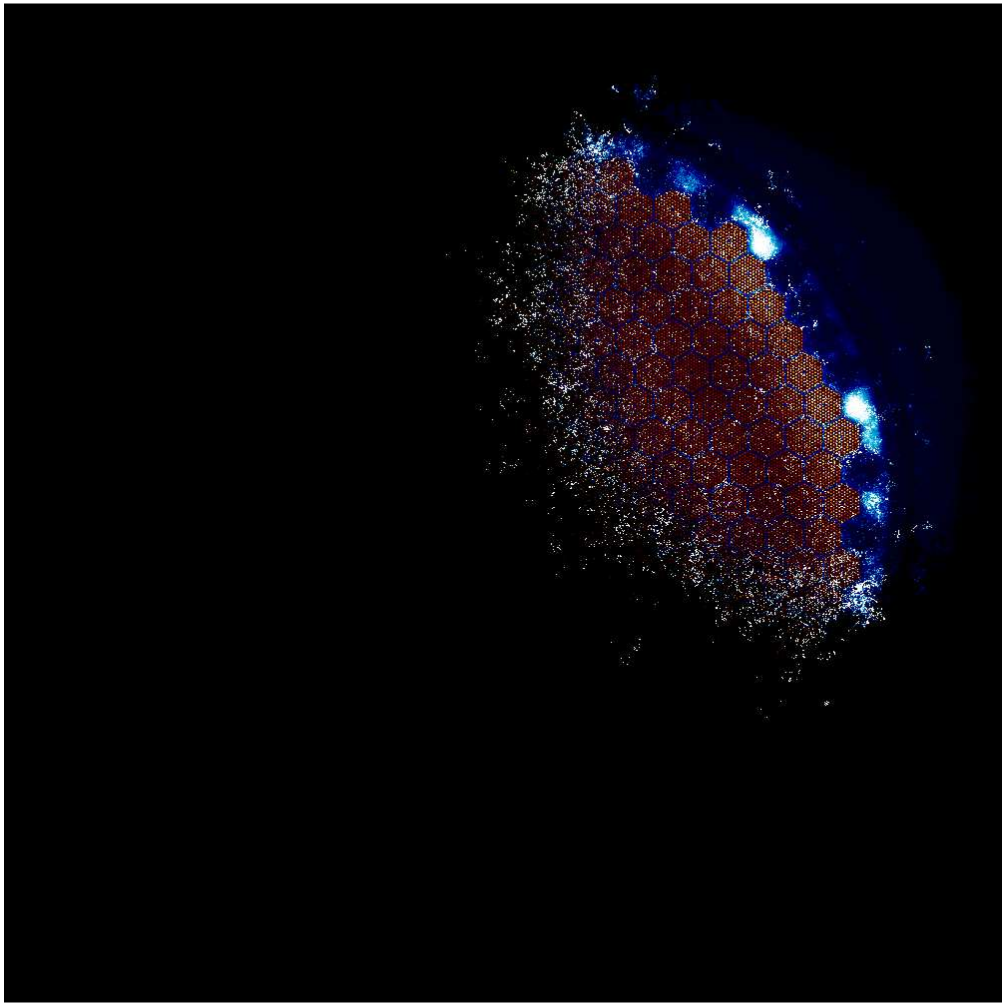
```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%% Final calculation input %%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Use weight-window variance reduction &
% Read importance map from file

wwin
wf "vver0.wwd" 1
```

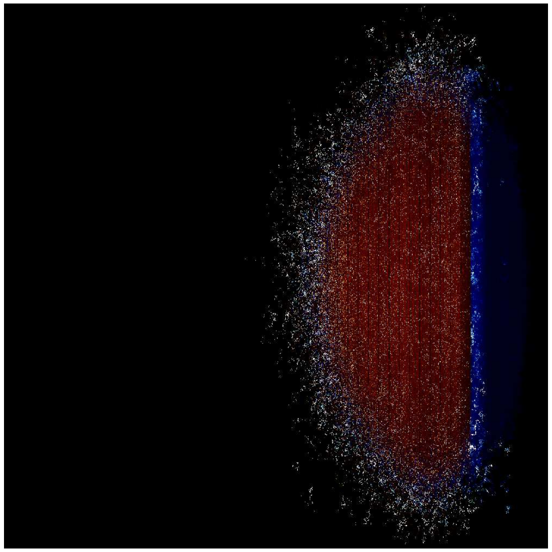
Importance map for variance reduction





Mesh plots with variance reduction





Final results: performance

60 Billion histories, 12 CPUs, Calculation time 50 h

Response	Threshold	$t_{1/2}$	Std.dev.	FOM
^{54}Fe(n,p)^{54}Mn	2.2 MeV	312 d	1.34 %	110 1/h
^{58}Ni(n,p)^{58}Co	1.9 MeV	71 d	1.18 %	142 1/h
^{63}Cu(n,α)^{60}Co	4.7 MeV	5.3 y	4.05 %	12 1/h
$^{\mathrm{nat}}$Ti(n,X)^{46}Sc	3.7 MeV	84 d	2.49 %	31 1/h
^{93}Nb(n,n')$^{93\mathrm{m}}$Nb	0.8 MeV	16 y	0.55 %	644 1/h
^{59}Co(n,γ)^{60}Co	-	5.3 y	0.43 %	1069 1/h
^{93}Nb(n,γ)^{94}Nb	-	20300 y	0.71 %	386 1/h
^{58}Fe(n,γ)^{59}Fe	-	45 d	0.37 %	1468 1/h

Variance reduction reduces the CPU time requirement by factor $\sim$100–200.

Final results: C/E ratios

Response	Threshold	$t_{1/2}$	C/E
^{54}Fe(n,p)^{54}Mn	2.2 MeV	312 d	1.065
^{58}Ni(n,p)^{58}Co	1.9 MeV	71 d	1.154
^{63}Cu(n,α)^{60}Co	4.7 MeV	5.3 y	0.960
$^{\mathrm{nat}}$Ti(n,X)^{46}Sc	3.7 MeV	84 d	1.277
^{93}Nb(n,n')$^{93\mathrm{m}}$Nb	0.8 MeV	16 y	0.957
^{59}Co(n,γ)^{60}Co	-	5.3 y	1.659
^{93}Nb(n,γ)^{94}Nb	-	20300 y	1.060
^{58}Fe(n,γ)^{59}Fe	-	45 d	1.991

Conclusions

- Variance reduction of Serpent 2.1.27 is easy to use and seems to be working as intended (in this case:)
- The variance reduction technique decreases the CPU time requirement by factor 100 (or even above)
  - Makes out-of-core calculations computationally feasible!
- The results obtained with the current (inaccurate) model were very promising. Updating the model would possibly bring results even closer to measurements.

Questions?

tuomas.viitanen@vtt.fi

http://montecarlo.vtt.fi

References

[1] G. Yesilyurt, W.R. Martin and F.B. Brown, “On-the-fly Doppler broadening for Monte Carlo codes”, Nucl.Sci.Eng., 171, 239–257 (2012).

[2] B. Forget, S. Xu and K. Smith, “Direct Doppler broadening in Monte Carlo simulations using the multipole representation”, Ann. Nucl. Energy, **64**, 78–85, (2014).

[3] A. Pavlou and Wei Ji, “On-the-fly sampling of temperature-dependent thermal neutron scattering data for Monte Carlo simulations”, *Ann. Nucl. Energy*, **71**, pp. 411-426, (2014).

[4] T. Viitanen and J. Leppänen, “Explicit treatment of thermal motion in continuous-energy Monte Carlo tracking routines”, *Nucl. Sci. Eng.*, **171**, 165-173 (2012).

[5] T. Viitanen and J. Leppänen, “Explicit temperature treatment in Monte Carlo neutron tracking routines – First results.” In proc. PHYSOR 2012, Knoxville, TN, Apr. 15-20, (2012).

[6] T. Viitanen and J. Leppänen, “Optimizing the implementation of the target motion sampling temperature treatment technique – How fast can it get?”, In proc. M&C 2013, Sun Valley, ID, May 5-9, (2013).

[7] T. Viitanen and J. Leppänen, “Target motion sampling temperature treatment technique with elevated basis cross section temperatures.”, *Nucl. Sci. Eng.*, **177**, 77-89 (2014).

[8] T. Viitanen and J. Leppänen, “Temperature majorant cross sections in Monte Carlo neutron neutron tracking”, *Nucl. Sci. Eng.*, Accepted for publication on Aug 31. 2014.

[9] T. Viitanen and J. Leppänen, “Effect of the Target Motion Sampling Temperature Treatment Method on the Statistics and Performance“, *Ann. Nucl. Energy*, accepted, in-press.