Available online at ScienceDirect

Nuclear Engineering and Technology

journal homepage: www.elsevier.com/locate/net

Original Article

Domain Decomposition Strategy for Pin-wise Full-Core Monte Carlo Depletion Calculation with the Reactor Monte Carlo Code

Jingang Liang [a,*], *Kan Wang* [a], *Yishu Qiu* [a], *Xiaoming Chai* [b], *and Shenglong Qiang* [b]

[a] *Department of Engineering Physics, LiuQing Building, Tsinghua University, Beijing, 100084, China*
[b] *Science and Technology on Reactor System Design Technology Laboratory, Nuclear Power Institute of China, Changshun Ave 328, Shuangliu, Chengdu 610041, China*

ARTICLE INFO

Article history:
Received 19 October 2015
Received in revised form
14 December 2015
Accepted 19 January 2016
Available online 9 February 2016

Keywords:
Domain Decomposition
Full-Core Depletion
Monte Carlo
Reactor Monte Carlo
RPHA15

ABSTRACT

Because of prohibitive data storage requirements in large-scale simulations, the memory problem is an obstacle for Monte Carlo (MC) codes in accomplishing pin-wise three-dimensional (3D) full-core calculations, particularly for whole-core depletion analyses. Various kinds of data are evaluated and quantificational total memory requirements are analyzed based on the Reactor Monte Carlo (RMC) code, showing that tally data, material data, and isotope densities in depletion are three major parts of memory storage. The domain decomposition method is investigated as a means of saving memory, by dividing spatial geometry into domains that are simulated separately by parallel processors. For the validity of particle tracking during transport simulations, particles need to be communicated between domains. In consideration of efficiency, an asynchronous particle communication algorithm is designed and implemented. Furthermore, we couple the domain decomposition method with MC burnup process, under a strategy of utilizing consistent domain partition in both transport and depletion modules. A numerical test of 3D full-core burnup calculations is carried out, indicating that the RMC code, with the domain decomposition method, is capable of pin-wise full-core burnup calculations with millions of depletion regions.

1. Introduction

With the higher requirements for the safety and economy of nuclear reactors, as well as the developments of new types of nuclear systems, traditional methods and tools for reactor analysis are being challenged. The Monte Carlo (MC) method is becoming an important area of research for the next generation of methods for reactor physics calculations. With the development of parallel computing technology, the expectations are rising to see the MC method being truly applied in nuclear reactor engineering design practices [1]. However, a prohibitive amount of data is required for storage in large-

* *Corresponding author.*
E-mail address: liangjg2008@gmail.com (J. Liang).

http://dx.doi.org/10.1016/j.net.2016.01.015

scale calculations in MC codes. Such excessive memory demands turn into a key obstacle for the application of MC method in accomplishing pin-wise three-dimensional (3D) full-core calculations. In particular, for whole-core burnup calculations with millions of burnup regions, the data storage reaches up to hundreds of gigabytes or even terabytes, which far exceed the capacity of current computers.

Data decomposition [2,3] and domain decomposition [4,5] are two feasible ways to solve the memory problem of MC. Through the former method, specific types of data are decomposed and distributed on different processors and parallel communications are called for data operation in neutron-simulating processes. For the latter method, the idea is to divide the problem model into smaller geometry domains, which are assigned to different processors. The domain-related data are, meanwhile, decomposed. The particles are communicated between processors in the domain decomposition method as the tracks of particles are cut into pieces.

In previous studies, tally data decomposition (TDD) algorithms [6] have been designed and implemented based on the Reactor Monte Carlo (RMC) code [7]. Thereafter, a combination of TDD and depletion isotope data decomposition [8] is utilized to alleviate the memory problem, enabling simple 3D whole-core MC burnup calculations with hundreds of thousands of depletion regions [9]. However, the memory problem still exists for larger scale or fine 3D whole-core burnup, because the material data cannot be decomposed in the TDD method. In this paper, the domain decomposition method is investigated to solve the memory problem thoroughly. Through this work, fine 3D whole-core burnup calculations with millions of depletion region are achieved.

2. Memory evaluation of MC codes

For an in-depth knowledge of the memory problem of MC, it is necessary to classify data and analyze each data class quantitatively. Taking RMC as the reference, normally suitable to other MC codes, the data can be classified into six categories: geometry, material, nuclear data, particles, tallies, and burnup. The memory model can be constructed by going deep into each data type and evaluating their memory sizes in detail, as shown in Eqs. (1) and (2):

$$M = M_{geo} + M_{mat} + M_{cs} + M_{part} + M_{tally} + M_{burn} + M_{temp} \tag{1}$$

$$M \approx N_{cell}\overline{m}_{cell} + N_{mat} + \overline{m}_{mat} + N_{tot\_nuc}\overline{m}_{nuc\_cs} + N_{part}m_{part} + N_{tally}\overline{m}_{tally} + N_{burncell}m_{burncell} \tag{2}$$

where total memory usage of a code, M, is the sum of the memory of different data types, M_x, with x as the data category. For example, M_{geo} defines memory of geometry data. M_{temp} represents all other temporary and supporting data, which are generally negligible in the memory footprint. Furthermore, all data of concern have a vector structure, and their memory sizes are proportional to the amount of unit data. Total memory approximates into Eq. (2), where N_y and $m_y/\overline{m}_y$ are the number and unit or average storage size of specific data structure y, respectively.

Specifically, for the RMC code, the unit or average storage of each data type can be estimated. For example, data of one material contain names (12 bytes), ID (4 bytes), and atom/mass densities (16 bytes) of all nuclides in the material. For a depletion calculation, assuming there is an average of 150 nuclides/region, the memory storage of one material is about $32 \times 150 = 4.8 \times 10^3$ bytes. Particle data contain eight double-precision floating variables (3 for coordinates, 3 for direction, 1 for energy, and 1 for weight) to record particle state information, and therefore, its unit size is 64 bytes. Similarly, unit storage of tally, which is composed of statistics and filter data, is about 70 bytes. For burnup data, RMC accounts for 1500 nuclides in the depletion chain and the predictor–corrector method is used, and therefore, the unit datum is 3.6×10^4 bytes. Finally, Eq. (3) is obtained to describe the memory consumption in burnup simulations using RMC.

$$M_{RMC} \approx (100 \times N_{cell} + 4.8 \times 10^3 \times N_{mat} + 2 \times 10^6 \times N_{tot\_nuc} + 64 \times N_{part} + 70 \times N_{tally} + 3.6 \times 10^4 \times N_{burncell})\text{bytes} \tag{3}$$

Table 1 summarizes unit storage, scale of unit, and maximum storage of each data type. It can be seen that three types of data (i.e., tally data, burnup, and material) are the main sources of memory problems of MC codes.

The Hoogenboom–Martin whole core [10] was chosen as a case study of large-scale MC burnup calculations. There are a total of 241 assemblies and 63,624 fuel rods in the Hoogenboom–Martin core. In the modeling, each rod contains 24 burnup regions (12 axially by 2 radially) to perform the depletion calculation. Table 2 predicts the memory storage using the memory model.

3. Domain decomposition method

Different from the data decomposition method, spatial domain decomposition (SDD) divides spatial geometry into domains, which are simulated separately by parallel processors, and particles crossing domains are communicated for continuing tracking.

As indicated in Fig. 1, the main steps involved in implementing SDD in particle transport MC code include (1) dividing

Table 1 – Memory storage evaluation of data types.

Data types	Unit storage	Scale	Maximum storage
Geometry	100 bytes	10^0–10^6	0.1 GB
Material	4.8 KB	10^0–10^7	10 GB
Nuclear data[a]	2 MB	10^0–10^2	1 GB
Particle	64 bytes	10^4–10^6	0.1 GB
Tally	70 bytes	0–10^{10}	100 GB
Burnup	36 KB	0–10^7	100 GB

GB, gigabyte; KB, kilobyte; MB, megabyte.

[a] Assuming nuclides are in single temperature.

Table 2 – Memory storage of H–M whole-core burnup.

Data types	Memory storage
Material	7.3 GB
Nuclear data	400 MB
Particle	64 MB
Tally	64.1 GB
Burnup	55.0 GB
Total	126.9 GB

GB, gigabyte; MB, megabyte.

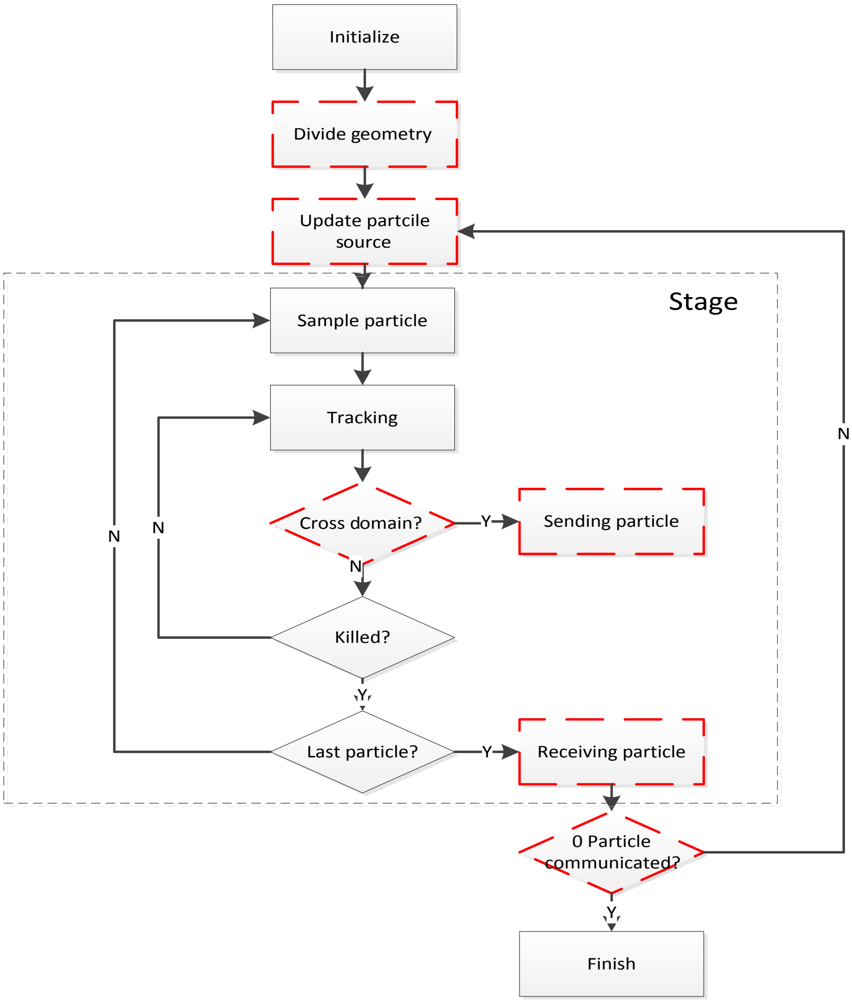

Fig. 1 – Flow chart of the domain decomposition method.

model geometry and decomposing data of tallies and materials simultaneously; (2) tracking particle histories by "stages," which are bounded with particle communications; and (3) finishing simulation by checking all particle stages and communications.

It should be noted that except for the domain partition strategy, the particle communication algorithm is a key part of SDD as well. The asynchronous particle communication algorithm is designed using the message passing interface [MPI] [11], as shown in Fig. 2. In the algorithm, particles flying out of current domains are buffered in a local processor, and a buffer will be sent to the remote processor when it is full, or when the current processor finishes tracking all particles. The MPI

```
decompose Geometry into Domains to all processes
for cycle = 1 to CycleNum
|  do     // stages
|  |  post non-blocking receives to all other processes(MPI_Irecv)
|  |  for neutron = 1 to NeutronNum
|  |  |  tracking history
|  |  |  |  test all receives(MPI_Test)
|  |  |  |  |  while(one receive success)
|  |  |  |  |  |  add received particle to local bank
|  |  |  |  |  |  if(not last recv)post receive to the same target (MPI_Irecv)
|  |  |  |  end while loop
|  |  |  |  if(cross domain)
|  |  |  |     save it to particle buffer
|  |  |  |     if(buffer is full)send this buffer to target(MPI_Isend)
|  |  |  end one history
|  |  end for loop
|  |  send all buffer out(MPI_Isend)
|  |  wait all send and receive success(MPI_Wait)
|  |  reduce all received particles number(MPI_Allreduce)
|  while(received particle exist)
|  process cycle end
end for loop
```

Fig. 2 – Asynchronous particle communication algorithm in the domain decomposition method.

nonblocking functions (MPI_Isend/MPI_Irecv and MPI_Test/MPI_Wait) are used for overlap of computing and passing messages, to shorten total communication time as much as possible.

Note that the particle buffer has a fixed size (Line 14 in the algorithm), which determines the period of communications. Theoretically, greater buffer size leads to less communication time, but the memory of buffers will increase as well. In practice, the utilization of nonblocking communication makes the parallel performance of the algorithm insensitive with the buffer size in a reasonable range. The buffer size is usually set as 10^3 in the SDD implementation in RMC.

4. Coupling domain decomposition with MC burnup

4.1. *Coupling strategy and memory estimation*

It is known that the MC burnup calculation process is a combination of MC neutron transport simulation and depletion equation computation. The basic calculation unit in depletion, that is, a burnup cell, is a geometrical region as well. Therefore, it is possible to couple domain decomposition with MC burnup by utilizing consistent domain partition in the transport and depletion processes. In other words, burnup regions can be decomposed automatically according to geometry decomposition in transport.

Table 3 – Main calculation parameters used.

Parameters	Data
Number of histories in Monte Carlo transport	1,000,000 particle/cycle 750 cycles (250 inactive)
Number of burnup regions	1,526,976
Number of isotopes/burnup region	1,487
Number of time steps	14
Step length (ΔMWD/kgU)	0.1, 0.4, 0.5, 1 × 11
Number of inner steps in depletion	10
Burnup strategy	Predictor–corrector
Power density (W/gU)	30
Total burnup (MWD/kgU)	12
Number of parallel processors	96

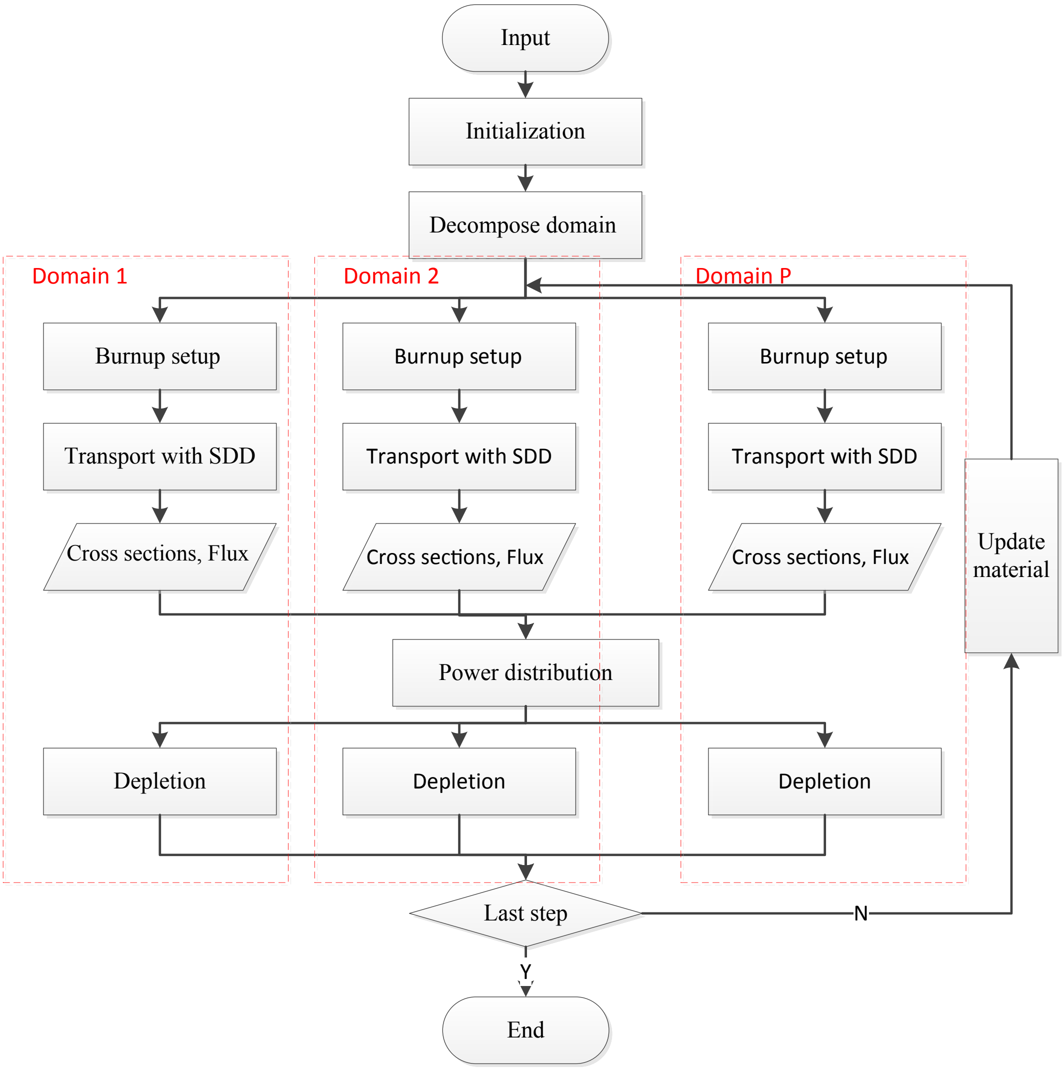

Fig. 3 – Frame of coupled domain decomposition. SDD, spatial domain decomposition.

Through the coupling, geometry-based data including data on tally, material, and isotope densities (i.e., the main memory-consuming data types) are all decomposed.

From the memory model in the "Memory Evaluation of MC Codes" section, assuming the geometry is evenly decomposed into P domains, the memory size of one domain can be estimated as Eq. (4).

$$M_{Domain} \approx M_{geo} + M_{cs} + M_{part\_Domain} + \frac{M_{mat} + M_{tally} + M_{bum}}{P} \qquad (4)$$

Three major data types are all proportional to the number of burnup cells, as shown in Eq. (5). Here, it is assumed that there are about 150 isotopes in every burnable cell or material for transport simulation. A total of four types of one-group cross section for every isotope in every material should be enabled for the depletion calculation [12].

$$M_{mat} + M_{tally} + M_{bum} \cong N_{burncell}\left(4.8 \times 10^3 + 70 \times 600 + 3.6 \times 10^4\right) \times \text{bytes} \approx 0.08 N_{burncell} \text{MB} \qquad (5)$$

Therefore, the necessary number of domain decomposition can be figured out according to the capacity of computer RAM. For example, to carry out MC burnup calculations on 4-GB RAM computers, assuming that memory size of data not decomposed is no more than 2 GB, the condition and result can be derived as follows [Eqs. (6) and (7)], indicating that no less than 42 domain partitions is adequate for a 1-million-region burnup calculation.

$$\frac{M_{mat} + M_{tally} + M_{burn}}{P} \leq 2\text{GB} \qquad (6)$$

$$P \geq 4.14 N_{burncell} \times 10^{-5} \qquad (7)$$

4.2. *Implementation of coupled domain decomposition*

Coupled domain decomposition is implemented in RMC based on the aforementioned coupling strategy, as shown in Fig. 3. It can be seen that depletion regions are grouped according to domain partition in the transport process.

5. Numerical test of 3D full-core burnup calculation

The Hoogenboom–Martin core is used for 3D full-core burnup calculation tests. Table 3 shows the main calculation parameters used in the calculation; 1.5 million burnup regions are set up with 24 burnup regions (12 axially by 2 radially) for each rod, illustrated by Fig. 4. This full core is decamped into 96 domains with eight pieces radially and 12 slices axially. The partitions in radial planes are performed in a way to make every assembly fully belong to only one domain, while balancing the loads of domains as well as possible.

The "Inspur TS10000" cluster at Tsinghua University, Beijing, China was used to run the 96-domain parallel calculation.

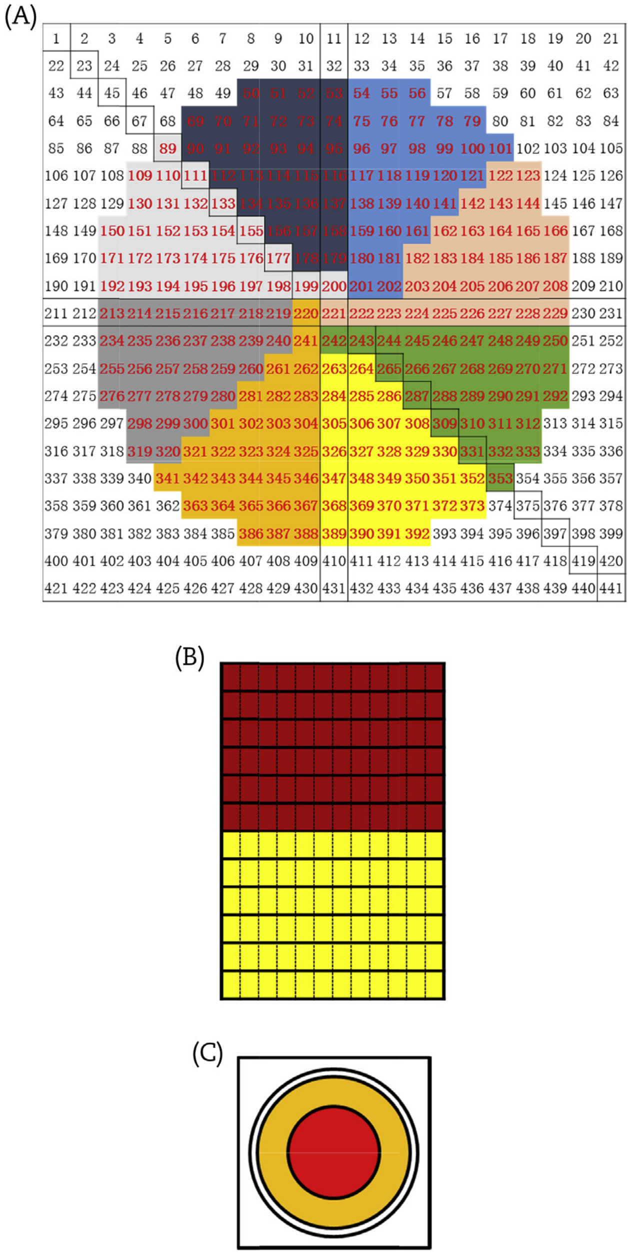

Fig. 4 – Domain and depletion region partition in three-dimensional full-core burnup. (A) Eight domains (8 colors) radially, without cutting each assembly; (B) 12 segments axially; and (C) two rings/rod.

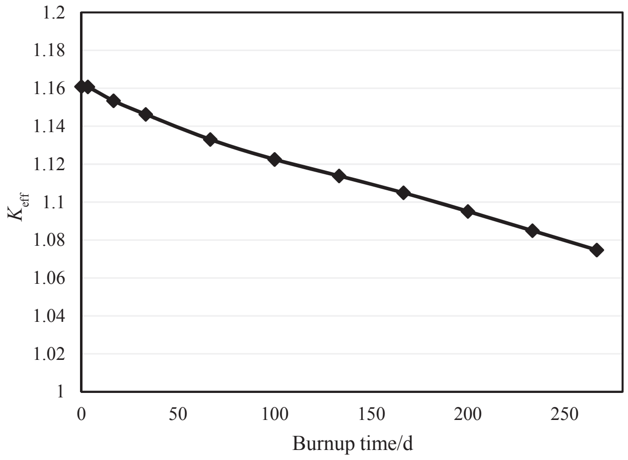

Fig. 5 – Variation of core K-effective (K_{eff}) with burnup.

As estimated in Section 2, the total memory storage consumed in this calculation is up to 127 GB, which exceeds the memory capacity of a normal computer processor. Nevertheless, it is demonstrated that the RMC code integrated with the domain decomposition method is capable of completing the calculation, and memory storage/processor is about 3 GB. Figs. 5–10 show the results of the calculation.

Fig. 5 is the K-effective (K_{eff}) variation along with burnup time. Figs. 6 and 7 are the radial pin-by-pin power distributions of the top layer of the core at 0 MWD/kgU and 5 MWD/kgU burnup points, respectively, whereas Figs. 8 and 9 are radial burnup distributions of the middle of the core, all plotted by jointing eight domains together. Fig. 10 shows the axial power distributions of 12 radially integrated segments at different burnup points.

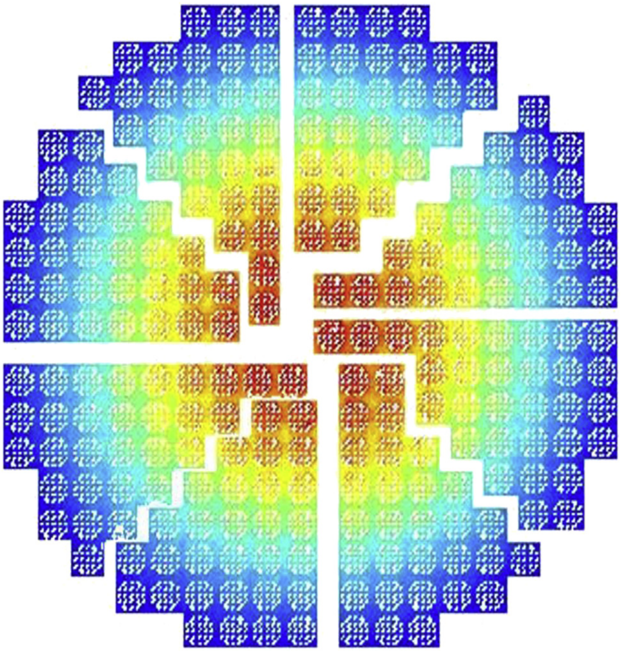

Fig. 6 – Core radial power distribution of Layer 0 (top) at 0 MWD/kgU. d, day.

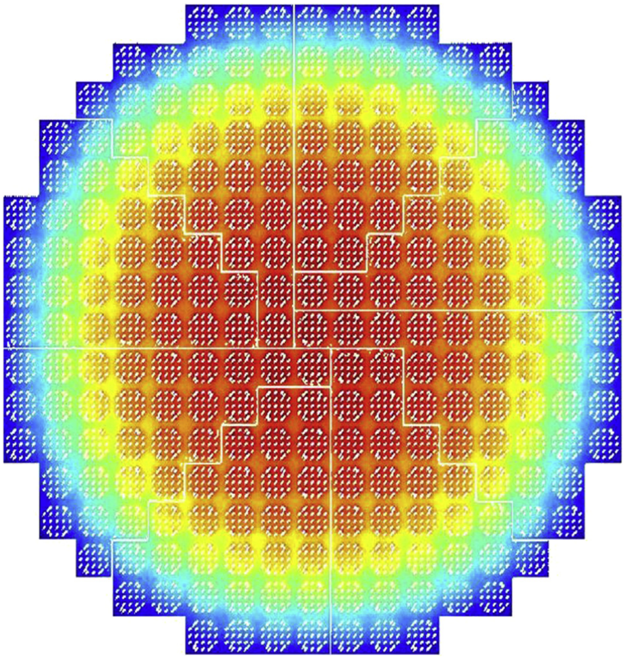

Fig. 8 – Core radial burnup distribution of Layer 6 (middle) at 5 MWD/kgU.

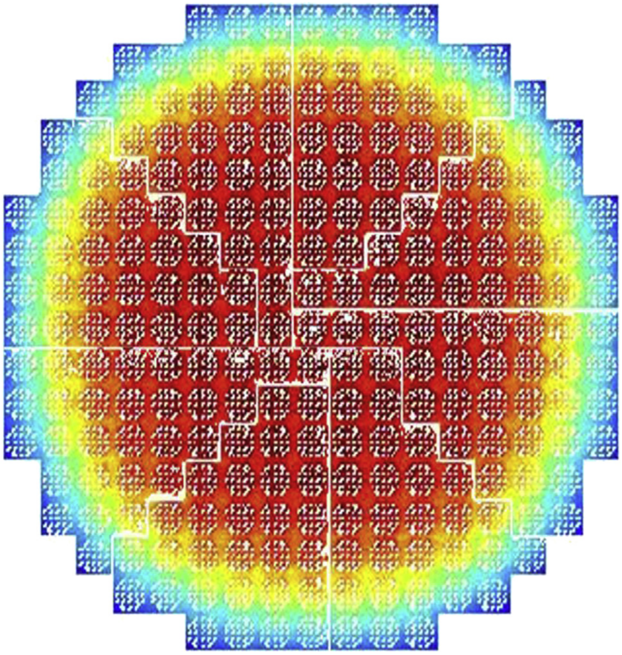

Fig. 7 – Core radial power distribution of Layer 0 (top) at 5 MWD/kgU.

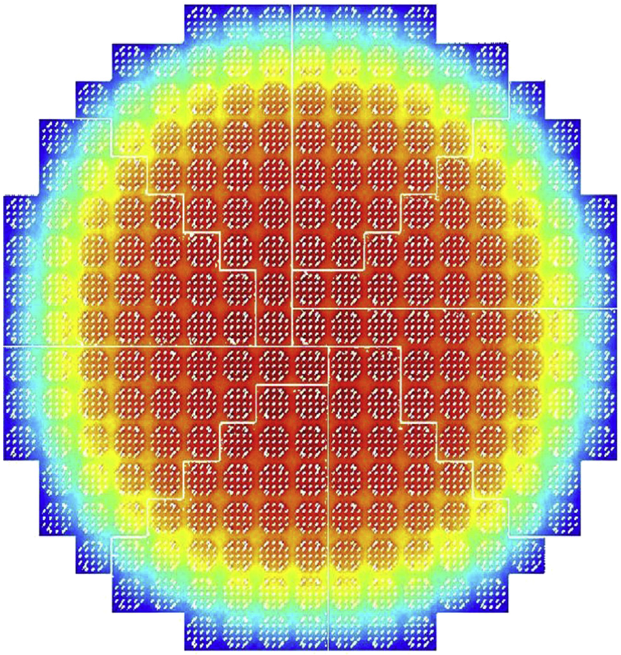

Fig. 9 – Core radial burnup distribution of Layer 6 (middle) at 8 MWD/kgU.

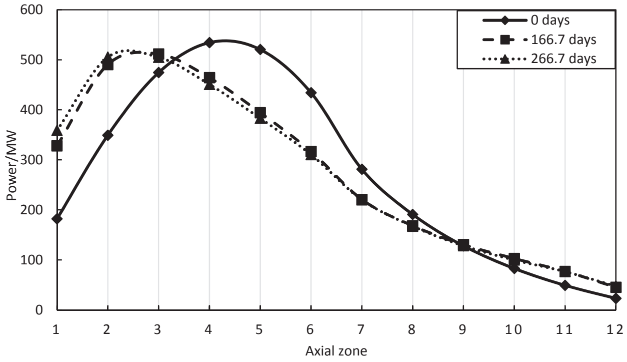

Fig. 10 – Axial power distributions.

6. Discussion

From a qualitative point of view, the results of the whole-core burnup calculation are reasonable. However, it should be noted that the numerical test is mainly performed to verify the capability of the domain decomposition method for extending the scale of MC burnup analysis. To achieve practical application in nuclear engineering, further improvements remain to be done. First, some more features need to be integrated, for example, criticality search function, to maintain the operation state of the core. Second, thermal-hydraulic coupling is required in reactor analysis to account for temperature feedback, which relies on efficient treatment of temperature-dependent nuclear cross sections, that is, on-the-fly Doppler broadening. Progress associated with these issues are under study [13,14].

In addition, the domain decomposition burnup strategy (e.g., the usage of a huge amount of data produced in depletion and the method of domain partition) is still not ideal for engineering use. Only one method of domain partition is used and performances are not tested in this study. Research shows that the pattern of decomposition, which determines source load imbalance between different domains, influences computing performance significantly [15].

In conclusion, the paper deals with the domain decomposition method and its coupling with burnup calculations in the RMC code. The successful running of the Hoogenboom–Martin benchmark with the assumed burnup cases demonstrates the effectiveness of domain decomposition methods for solving memory problems. It is indicated that domain-decomposed MC codes are capable of performing pin-wise full-core burnup calculations with millions of depletion regions. The next steps are to improve the practicability of the domain-decomposed burnup strategy and to integrate it with other latest features, such as thermal–hydraulics feedback, to eventually realize practical engineering application.

Conflicts of interest

All contributing authors declare no conflicts of interest.

Acknowledgments

This research work is partially supported by the Science and Technology on Reactor System Design Technology Key Laboratory , Chengdu, China.

REFERENCES

[1] W.R. Martin, Challenges and prospects for whole-core Monte Carlo analysis, Nucl. Eng. Technol. 44 (2012) 151–160.
[2] F.B. Brown, Recent advances and future prospects for Monte Carlo, Prog. Nucl. Sci. Technol. 2 (2011) 1–4.
[3] P.K. Romano, A.R. Siegel, B. Forget, K. Smith, Data decomposition of Monte Carlo particle transport simulations via tally servers, J. Comput. Phys. 252 (2013) 20–36.
[4] T.A. Brunner, P.S. Brantley, An efficient, robust, domain-decomposition algorithm for particle Monte Carlo, J. Comput. Phys. 228 (2009) 3882–3890.
[5] N. Horelik, A. Siegel, B. Forget, K. Smith, Monte Carlo domain decomposition for robust nuclear reactor analysis, Parallel Comput. 40 (2014) 646–660.
[6] J. Liang, K. Wang, G. Yu, D. She, X. Chai, S. Qiang, D. Yao, Research on tally data decomposition algorithms based on reactor Monte Carlo code RMC, Nucl. Power Eng. 35 (2014) 142–146.
[7] K. Wang, Z. Li, D. She, RMC-A Monte Carlo code for reactor physics analysis, Ann. Nucl. Energy 82 (2015) 121–129.
[8] D. She, J. Liang, K. Wang, B. Forget, 2D full-core Monte Carlo pin-by-pin burnup calculations with the RMC code, Ann. Nucl. Energy 64 (2014) 201–205.
[9] J. Liang, Y. Qiu, K. Wang, X. Chai, S. Qiang, Research on full core burnup calculations based on tally data decomposition using RMC, Nucl. Power Eng. 35 (2014) 231–234.
[10] J.E. Hoogenboom, W.R. Martin, B. Petrovic, The Monte Carlo Performance Benchmark Test: Aims, Specifications and First Results [CD-ROM], Proceedings from International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering (M&C 2011), Rio de Janeiro (Brazil), May 8–12, 2011, American Nuclear Society, 2011.
[11] Message Passing Interface Forum [Internet]. 2015 [cited 2016 Feb 2]. Available from: http://www.mpi-forum.org/index.html.
[12] D. She, Y. Liu, K. Wang, G. Yu, B. Forget, P.K. Romano, K. Smith, Development of burnup methods and capabilities in Monte Carlo code RMC, Ann. Nucl. Energy 51 (2013) 289–294.
[13] L. Li, H. Yuan, K. Wang, Coupling of RMC and CFX for analysis of pebble bed-advanced high temperature reactor core, Nucl. Eng. Des. 250 (2012) 385–391.
[14] B. Forget, S. Xu, K. Smith, Direct Doppler broadening in Monte Carlo simulations using the multipole representation, Ann. Nucl. Energy 64 (2014) 78–85.
[15] A.R. Siegel, K. Smith, P.K. Romano, B. Forget, K. Felker, The effect of load imbalances on the performance of Monte Carlo algorithms in LWR analysis, J. Comput. Phys. 235 (2012) 901–911.