Transactions of the Korean Nuclear Society Spring Meeting
Goyang, Korea, October 24-25, 2019

Parallel Monte Carlo Computing for a Failure Analysis on a Batch of Coated Fuel Particles of an HTR

Young Min Kim* and Chang Keun Jo
Korea Atomic Energy Research Institute
111, Daedeok-daero 989beon-gil, Yuseong-gu, Daejeon, 34057, Republic of Korea
* *Corresponding author: nymkim@kaeri.re.kr*

1. Introduction

A large number of coated fuel particles (CFPs) are contained in a fuel element of a high temperature reactor (HTR). The integrity of CFPs greatly affects the characteristics of the fission product migration in a fuel element. It should be evaluated how various mechanical and chemical hazards deteriorate the integrity of the CFPs during the operation of an HTR.

The Monte Carlo method has been used to analyze the pressure-vessel failure of CFPs [1,2]. In the method, a large number of random particles are sampled to account for the statistical variations. The design parameters such as component sizes and material properties are typically sampled from Gaussian statistical distributions, while the coating layer strengths are sampled from Weibull statistical distributions. The Monte Carlo method adapts well to the failure modes of the coating layers of a CFP. It, however, becomes very time-consuming as the number of related statistical parameters increases. A parallel computing is one means to shorten the long computation time of the Monte Carlo method.

This study describes how to implement an MPI (Message Passing Interface) parallel computing [3] into the previously developed Monte Carlo failure analysis method for random CFPs, COPA [4].

2. Implementation of a Parallel Monte Carlo Method

Fig. 1 displays an MPI parallel computing scheme implemented into a COPA module named FAIL which is used to calculate the failure fraction of a batch of CFPs using the Monte Carlo method. The MPI command MPI_COMM_RANK determines the rank of the calling process in the communicator, *myrank*. The other MPI command MPI_COMM_SIZE determines the size of the group associated with a communicator, *nprocs*. The variable *nptl* is the total number of CFPs to be analyzed. In the Monte Carlo method, *nptl* random particles are generated by determining their statistical quantities, as described in Ref. [5]. The failure analysis is applied to *nptl* random CFPs. In the MPI parallel computing, the *nptl* failure calculations are evenly distributed to participating computer processors. Each process has its own particle range, *ista* and *iend*. The stresses in the coating layers of a random particle are solved incrementally from the beginning to the end of irradiation. A CFP coating layer is assumed to break if its calculated maximum tensile stress is greater than its sampled strength. The MPI command MPI_REDUCE reduces failure counts on all processors to a single value.

3. Failure Fraction Calculations

Table I lists the CFP design parameters and coating layer strengths. It was assumed that only the coating layer thicknesses had Gaussian distributions. Their standard deviations were deliberately largely set in order to allow CFP failure to occur prematurely. The CFPs were assumed to be irradiated for 1500 days.

Table I: CFP design parameters and coating layer strengths

Coating layers	IPyC	SiC	OPyC
Density (g/cm^3)	1.9	3.2	1.9
[a]BAF	1.03	1	1.03
Thickness (μm)	40±2	35±1.75	40±2
Weibull modulus	9.5	6	9.5
Median Strength (MPa)	350	770	350

[a] BAF = Bacon Anisotropy Factor

The failure calculations were executed on a CentOS 7 cluster with 9 nodes and 96 processers. The parallel computation software used is MPICH-3.3 [6], a free implementation of MPI standard. Fig. 2 shows the failure fractions of a batch of CFPs calculated using the parallel Monte Carlo method currently being developed. The failure fractions calculated by varying the number of processors are in good agreement.

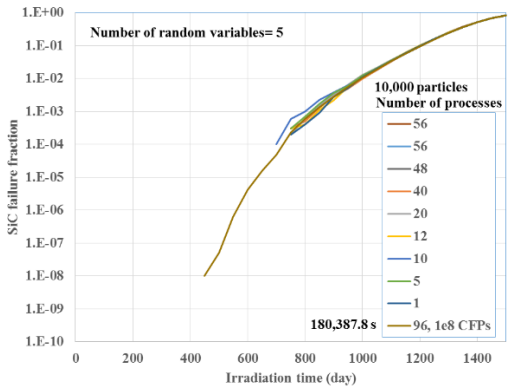

Fig. 2. Failure fractions of a batch of CFPs calculated using a parallel Monte Carlo method.

Fig. 3 shows the wall time for failure calculations for 10000 CFPs. Within a main node with 12 processors, the wall time decreases significantly as the number of

processors increase. For some reason not yet identified, however, the use of sub-node processors no longer reduces computation time.

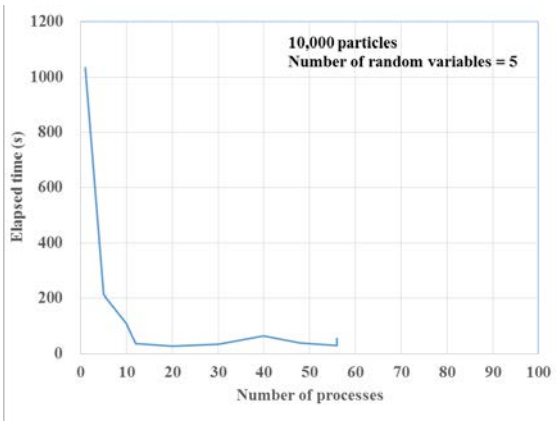

Fig. 3. Elapsed times for the failure calculations of 10000 CFPs.

4. Summary

A MPI parallel Monte Carlo method for the CFP failure fraction analysis has been developed to reduce the computation time. In the method, the CFPs are grouped into as many groups as there are processors. Calculations for each CFP group are assigned sequentially to the processors. The calculated failure fractions were consistent regardless of the number of participating processors. The intra-node wall time is greatly reduced, but not the inter-node wall time. The trouble related to the inter-node wall time will be scrutinized and solved.

ACKNOWLEDGEMENTS

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (No. 2017M2A8A1014757).

REFERENCES

[1] Kovacs, W.J., Bongartz, K., and Goodin, D.T., 1985. High-temperature Gas-cooled Reactor Fuel Pressure Vessel Performance Models. Nuclear Technology 68, pp. 344-354.
[2] Miller, G.K., Maki, J.T., Knudson, D.L., Petti, D.A., 2010. Calculating failure probabilities for TRISO-coated fuel particles using an integral formulation. Journal of Nuclear Materials 399, pp. 154-161.
[3] Message Passing Interface Forum, 2015. MPI: A Message-Passing Interface Standard Version 3.1. https://www.mpi-forum.org/docs/.
[4] Kim, Y.M., Cho, M.S., Lee, Y.W., and Lee, W.J., 2008. Development of a Fuel Performance Analysis Code COPA. In: Proceedings of the 4th International Topical Meeting on High Temperature Reactor Technology, HTR2008, September 28-October 1, 2008, Washington, DC USA.
[5] Kim, Y.M, Jo, C.K., and Kim, E.S., 2019. Random Particle Generation for a Failure Analysis on the Coated Fuel Particles of an HTR. In: Transactions of the Korean Nuclear Society Spring Meeting, Jeju, Korea, May 23-24, 2019.
[6] MPICH. https://www.mpich.org/.

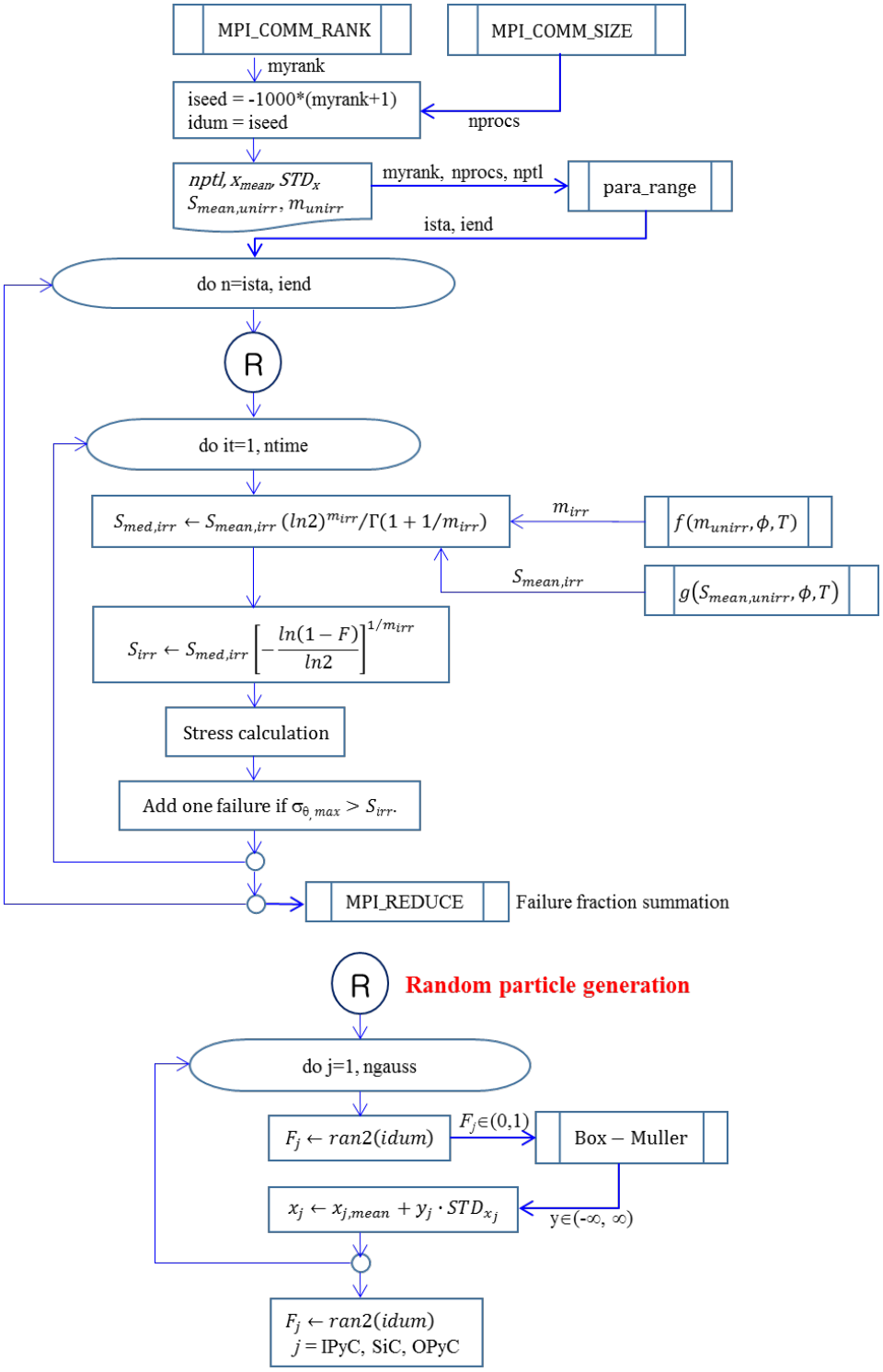

Fig. 1. MPI parallel Monte Carlo computing of the failure fraction of a batch of CFPs.