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

Development of AutoCASK Code System for PWR Spent Nuclear Fuel Cask Analysis at UNIST

Wonkyeong Kim [a], Jaerim Jang [a], Bamidele Ebiwonjumi [a], Hyunsuk Lee [a], Matthieu Lemaire [a], Peng Zhang [a], Seung-gu Kang [b], Hanjoo Kim [a], Nhan Nguyen Trong Mai [a], Sooyoung Choi [a], Jinsu Park [a], Jiwon Choe [a], and Deokjung Lee [a*]
[a] *Department of Nuclear Engineering, Ulsan National Institute of Science and Technology, 50 UNIST-gil, Ulsan, 44919, Republic of Korea*
[b] *School of Mechanical Engineering, Pusan National University, 2, Busandaehak-ro 63beon-gil, Geumjeong-gu, Busan, 46241, Korea*
[*] *Corresponding author: deokjung@unist.ac.kr*

1. Introduction

The transportation, storage and disposal of spent nuclear fuel (SNF) have become inevitable concerns ever since the Kori unit-1 nuclear power plant (NPP) was shut down permanently. One of the major factors to address those concerns in the process of back-end nuclear fuel cycle is the accurate prediction of the source term of SNF, that is, the radioactivity, energy, neutron or gamma released per unit time and other characteristics emitted to the surrounding environment in the event of a severe accident in a nuclear reactor or SNF storage facility. The nuclear facilities required for the process should be designed in accordance with guidelines for nuclear protection purposes, including radioactive waste management and environmental radiological safety. It is necessary to accurately estimate the radiation doses to humans and the surrounding environment due to nuclear installations and to ensure that they do not exceed the limits given in the guidelines.

In order to store SNFs into a spent fuel storage cask or transport them safely into an intermediate storage facility using a transportation cask, a safety assessment of those casks loaded with SNFs should be carried out. A safety assessment is a periodic evaluation in terms of the criticality, the surface dose and the surface temperature of the cask loaded with SNFs. The evaluation should be carried out with the exact isotopic inventory of SNFs, and the evaluation should ensure that such criteria meet regulatory standards.

To perform the evaluation, the isotopic inventory of SNF must first be determined. However, obtaining all the SNF isotopic inventories through measurements would represent massive work and cannot be done realistically, thus a code system that predicts SNF isotopic inventory based on the operation history is required. Then, analysis of the SNF casks in terms of criticality and shielding (radiation dose) should be performed with detailed isotopic inventory and geometric representation.

For the purpose of the evaluation, the integrated and user-friendly code system, AutoCASK (Automated Cask Analysis Supporting toolKit), is developed by the Computational Reactor Physics and Experiment Laboratory (CORE lab.) in Ulsan National Institute of Science and Technology (UNIST).

2. Description of AutoCASK code system

In this section, the different codes integrated in the AutoCASK code system and the role of each code are described. The AutoCASK code system includes STREAM/RAST-K2.0 (ST/R2), STREAM-SNF, SNF-DB, MCS and AutoLOADER codes as shown in Fig. 1.

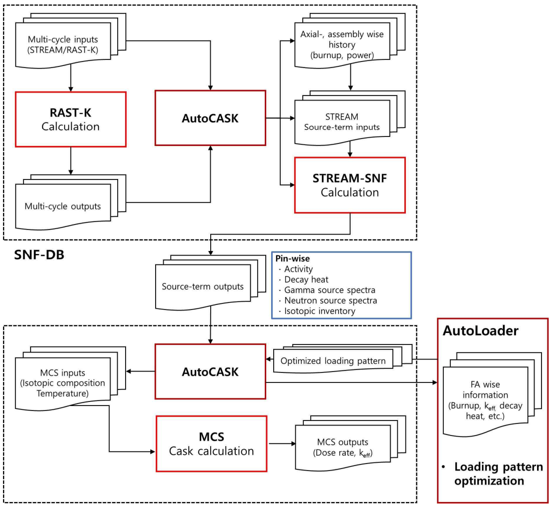

Fig. 1. Flowchart of AutoCASK code system.

Supporting Python scripts to process inputs/outputs and automatically format the data between the codes contained in the AutoCASK code system have been written and are listed in Table I. “autocask.py” is the main Python script that controls the full calculation flow of the AutoCASK code system according to the “input” file given by users. The “input” file includes flag options to turn on or off selected stages of the AutoCASK calculation flow. It also includes the required file directory, the loading pattern in cask, and the cooling time.

Table I: Python codes supporting AutoCASK code system

File	Description
autocask.py	Main code
autocask_h.py	Header file
input	AutoCASK input
snfinpgen.py	ST/R2 output processing code

2.1 Stage 1: Core follow calculation with STREAM/RAST-K2.0

The first step to evaluate SNF isotopic inventory based on the operation history is to perform core follow calculations. In AutoCASK, conventional two-step method is adopted for core follow calculations with the STREAM/RAST-K2.0 (ST/R2) code system [1] developed by CORE in UNIST.

The neutron transport analysis code, STREAM (Steady state and Transient REactor Analysis code with Method of Characteristics), can perform LWR core calculations with either the direct transport analysis method (2-D) or the two-step method with RAST-K 2.0. One of the advanced features of STREAM is the pin-based pointwise energy slowing down method (PSM) [2] to tackle resonance treatment with high accuracy and performance.

The three-dimensional two-group nodal diffusion PWR analysis code, RAST-K2.0, has been developed to perform in-core fuel management, core design, load follow simulation and transient analysis in neutronics.

Solver and features of ST/R2 have been validated against commercial reactor cores (OPR-1000, three-loop Westinghouse reactor core, and APR-1400) and verified against other conventional code systems [1].

The output of ST/R2 core follow calculation provides node-wise history information as follow for each cycle, burnup step, fuel assembly and axial node:

- Power [W], *P*
- Fuel temperature [K], T_f
- Moderator temperature [K], T_m
- Boron concentration [ppm], *C*
- Burnup [MWD/kgU], *B*

2.2 Stage 2: Fuel assembly wise source term calculation based on operation history with STREAM-SNF

The output of ST/R2 core follow calculation is input to STREAM-SNF source term calculation module for discharged fuel assemblies.

A SNF characterization module has been implemented in STREAM to calculate the following source terms from pin-wise/assembly-wise SNF isotopic inventories generated by STREAM depletion module: radioactivity, gamma power, decay heat, source spectra for neutron and gamma [3]. The capability that predicts isotopic compositions in STREAM has been validated for PWR SNF [4].

To generate STREAM-SNF inputs, it is necessary to know when the fuel assemblies loaded in each operation cycle are discharged and how many times the fuel assemblies have been burned. AutoCASK automatically scans all given ST/R2 outputs to find out this information and generates STREAM-SNF inputs accordingly for discharged fuel assemblies with node wise history information, as can be seen in Fig. 2.

In this second stage, the following information is output:

- <Title>_SNF_FA.hist file
  : Information of discharged fuel assemblies (discharged position, volume, burnup, infinite multiplication factor)
- STREAM-SNF inputs
  : NPP (title), CY (cycle), A (fuel assembly ID), H (axial node index)

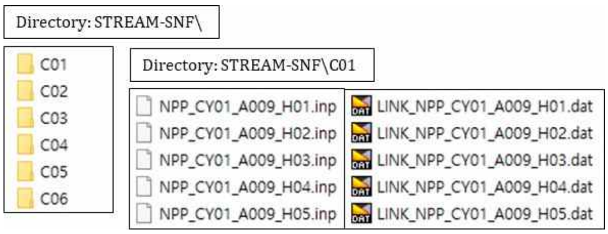

Fig. 2. Example of automatic generation of STREAM-SNF inputs based on operation history.

2.3 Stage 3: STREAM-SNF source term calculation and post-processing to establish the SNF database (SNF-DB)

Using the inputs generated at the end of stage 2, STREAM-SNF calculations generate two types of file as shown in Fig. 3. The first file is an output file in ASCII format that includes basic geometry, material, calculation options and burnup information. The second file is a SNF file in binary format that includes fuel pin-wise source term data.

STREAM-SNF calculations are performed for a two-dimensional fuel assembly model with reflective boundary conditions. As one approximation, the code system currently disregards actual leakage conditions for the source term calculation.

By default, the cooling interval is split in 24 steps from the discharge date to 200 years after the discharge date. These cooling intervals can be modified by the user.

The output of SNF file provides pin-wise source term data as follows for each pin and axial node:

- Gram density [g/cm^3]
- Number density [#/barn-cm]
- Activity [Bq]
- Decay heat [W]
- Gamma power [W]
- (α, n) source [n/s]
- Spontaneous neutron source [n/s]
- Delayed neutron source [n/s]
- Neutron source spectra [n/s]
- Gamma source spectra [n/s]

AutoCASK supports interpolation functions to generate data anew for any cooling time from a given discharge date and simulate the isotopic inventory changes according to the exponential decay law.

For one fuel assembly, given any cooling time, AutoCASK post-processes SNF files to generate two files: a SRC file that contains source term information for all the pins and an ASRC file that contains merged source term information for one fuel assembly. For Westinghouse 14x14 fuel assembly type with 24 axial nodes in the nodal calculation, the sizes of the data files constituting the SNF database (SNF-DB) are calculated in Table II.

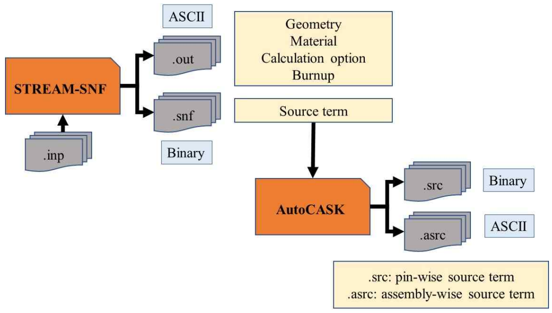

Fig. 3. Flowchart of STREAM-SNF source term calculation and post-processing for the establishment of SNF-DB.

Table II: Sizes of data files constituting the SNF-DB for one discharged fuel assembly of type Westinghouse 14x14 assemblies with 24 axial nodes

File	Data configuration	Data size
SNF	24 steps * (14 * 14 – 17) fuel pins * 1 axial node	~350 Mb
SRC	1 step * (14 * 14 – 17) fuel pins * 24 axial nodes	~350 Mb
ASRC	1 step * (14 * 14 – 17) fuel pins * 1 axial node	~130 Kb[1)]

1) ASCII format

2.4 Stage 4: Automated MCS input generation with optimized loading pattern by AutoLOADER

The Monte Carlo code MCS is under development at UNIST CORE since 2013. MCS is a 3D continuous-energy neutron and photon physics code for particle transport based on the Monte Carlo method. MCS allows for 3 kinds of calculation: neutron criticality runs for criticality safety analysis, neutron fixed-source runs and photon fixed-source runs for shielding problems. MCS neutron transport kernel is experimentally validated against ~300 benchmarks of the International Criticality Safety Benchmark Experimental Problem (ICSBEP) database [5, 6]. MCS photon transport capability [7] is verified against the Monte Carlo codes MCNP6.1 [8] and SERPENT2.1.29 [9].

AutoLOADER is a module developed by Pusan University that performs SNF loading pattern optimization [10]. Acceptable loading patterns must satisfy several safety criteria regarding the reactivity, radiation dose and integrity of the cask, thus setting boundary conditions on the loaded SNF, such as maximum acceptable burnup or minimum cooling time. Depending on the SNF type and the cask model, the safety margin can be increased (and so the utilization of the cask improved) by means of loading pattern optimization.

Variable information for the optimization provided by AutoCASK to AutoLOADER deals with the cooling period, burnup, infinite multiplication factor, radioactivity, decay heat, gamma power, neutron source and gamma source for each SNF. The optimized loading pattern output by AutoLOADER can be used directly as input for AutoCASK.

In the SRC file from stage 3, the pin- and axial- wise number density information of fuel assemblies for a given cooling time is used as the material input for Monte Carlo criticality and shielding simulations. AutoCASK converts the number density files into a file format that MCS can read based on the optimized loading pattern from AutoLOADER. Then, based on the cask input created by the user, AutoCASK can automatically generate MCS inputs for various SNF loading patterns as shown in Fig. 4.

Based on SNF-DB and AutoLOADER, AutoCASK provides water-filled and air-filled cask MCS inputs for the criticality calculation, the neutron fixed-source run, and the photon fixed-source run. The criticality calculation determines the effective neutron multiplication factor of the loaded cask whereas the neutron and photon fixed-source runs calculate the neutron and photon flux and dose as illustrated in Fig. 5. The photon and neutron dose rates presented in Fig. 5 are compared with those of MCNP6, showing good agreements with each other within three standard deviations [12].

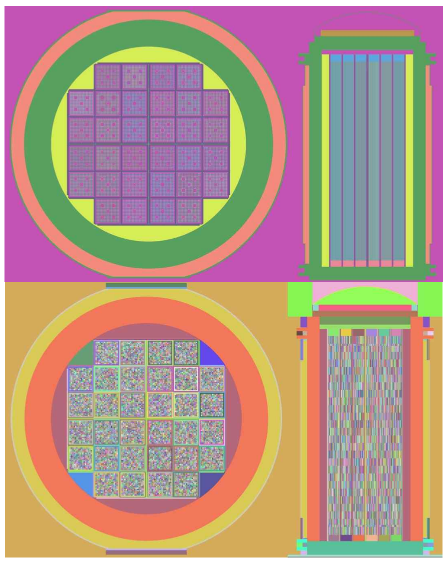

Fig. 4. Material and index configuration for MCS TN-32 cask input by AutoCASK code system.

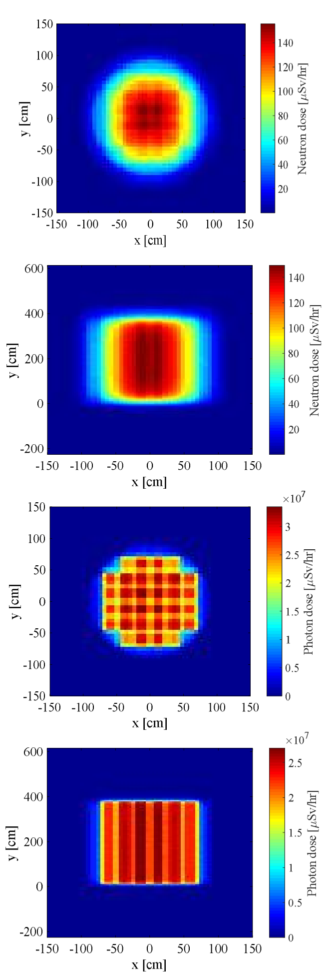

Fig. 5. MCS results of neutron and photon dose calculation for TN-32 cask loaded with SNFs.

3. Conclusions

The spent nuclear fuel analysis code system, AutoCASK, developed at UNIST, has been introduced in this paper. AutoCASK performs core follow calculations based on an operation history or a design report to track the irradiation history of discharged fuel assemblies. The irradiation history (power, burnup, fuel temperature, moderator temperature, and boron concentration) for SNFs during the core follow calculations can be simulated with any number of axial nodes in the RAST-K2.0 nodal calculation. Specific Python scripts of AutoCASK help retrieve quickly the discharge time, position and operation cycles of given SNF assemblies.

AutoCASK produces STREAM-SNF inputs reflecting the irradiation history of each SNF with axial discretization. Subsequent STREAM-SNF calculations produce the source term results, and AutoCASK processes them to provide the source term result after a cooling time specified by the user.

AutoCASK provides automated MCS input generation for a given loading pattern for both criticality and shielding analysis of cask loaded with specific SNF. In addition, AutoLOADER can determine optimized SNF loading pattern based on source term, burnup, and multiplication factor of SNFs.

Finally, MCS performs the criticality calculation and the fixed-source shielding calculation (neutron and photon transport) for the cask loaded with SNF.

Acknowledgments

This research was supported by the project (L18S015000) by Korea Hydro & Nuclear Power Co. Ltd..

REFERENCES

[1] J. Choe, S. Choi, P. Zhang, J. Park, W. Kim, H. C. Shin, H. S. Lee, J. Jung, D. Lee, Verification and validation of STREAM/RAST-K for PWR analysis, Nuclear Engineering and Technology, vol.51, pp.356-368, 2019.

[2] S. Choi, C. Lee, D. Lee, Resonance Treatment using Pin-Based Pointwise Energy Slowing-Down Method, Journal of Computational Physics, vol.330, pp.134-155, 2017.

[3] B. Ebiwonjumi, S. Choi, M. Lemaire, D. Lee, Verification and Validation of Radiation Source Term Capabilities in STREAM, Annals of Nuclear Energy, vol.124, pp.80-87, 2019.

[4] B. Ebiwonjumi, S. Choi, M. Lemaire, D. Lee, H. C. Shin, Validation of Lattice Physics Code STREAM for Predicting Pressurized Water Reactor Spent Nuclear Fuel Isotopic Inventory, Annals of Nuclear Energy, vol.120, pp.431-449, 2018.

[5] J. Jang, W. Kim, S. Jeong, E. Jeong, J. Park, M. Lemaire, H. Lee, Y. Jo, P. Zhang, D. Lee, Validation of UNIST Monte Carlo Code MCS for Criticality Safety Analysis of PWR Spent Fuel Pool and Storage Cask, Annals of Nuclear Energy, vol.114, pp.495-509, 2018.

[6] International Handbook of Evaluated Criticality Safety Benchmark Experiments NEA/NCS/DOC(95)03/IV, September 2014 Edition.

[7] M. Lemaire, H. Lee, B. Ebiwonjumi, C. Kong, W. Kim, Y. Jo, J. Park, D. Lee, Verification of Photon Transport Capability of UNIST Monte Carlo Code MCS, Computer Physics Communications, vol.231, pp.1-18, 2018.

[8] D. Pelowitz, MCNP6 user's manual. Los Alamos National Laboratory; 2013. (LACP-13-00634).

[9] J. Leppänen, M. Pusa, T. Viitanen, V. Valtavirta, T. Kaltiaisenaho, The Serpent Monte Carlo code: Status, development and applications in 2013, Annals of Nuclear Energy, vol.82, pp.142-150, 2015.

[10] S. Kang, H. C. Shin, H. C. Lee, A Comparison of Cooling Schedules in Spent Nuclear Fuel Grouping Optimization by Simulated Annealing for Dry Cask Loading, Transactions of the Korean Nuclear Society Spring Meeting, May 23-24, 2019, Jeju, Korea.

[11] U.S.NRC, TN-32 Dry Storage Cask System Safety Evaluation Report. https://www.nrc.gov/docs/ML0036/ML003696918.pdf

[12] N. N. T. Mai, P. Zhang, M. Lemaire, B. Ebiwonjumi, W. Kim, H. Lee, D. Lee, Extension of Monte Carlo Code MCS to Spent Fuel Cask Shielding Analysis, International Journal of Energy Research, under review (2019).