Proceedings of the 33rd European Safety and Reliability Conference (ESREL 2023)
Edited by Mário P. Brito, Terje Aven, Piero Baraldi, Marko Čepin and Enrico Zio

doi: 10.3850/978-981-18-8071-1_P247-cd

Development of Post-Processing Methods for Dynamic Event Analyzer, DICE

Yuntae Gwak

Department of Nuclear Engineering, Kyung Hee University, 1732 Deogyeong-daero, Giheung-gu, Yongin-si, Gyeonggi-do 17104, Republic of Korea. E-mail: gwakyuntae@gmail.com

Gyunyoung Heo*

Department of Nuclear Engineering, Kyung Hee University, 1732 Deogyeong-daero, Giheung-gu, Yongin-si, Gyeonggi-do 17104, Republic of Korea. E-mail: gheo@khu.ac.kr

*Corresponding author

DDET (Discrete Dynamic Event Tree) method or MCET (Monte Carlo Event Tree) are methods where the reliability of components and operator's interventions are contextually reflected in safety analysis over time. DDET and MCET methods uses both deterministic analyses using physical simulations and probabilistic state changes, where the stochastic features could be used to find undetermined scenarios that are not discussed in conventional deterministic or probabilistic analysis. DICE™ (Dynamic Integrated Consequence Evaluation) is a tool on the basis of DDET and MCET method. In the MCET method, the distribution of scenarios could be examined because stochastic considerations including recovery of the components and operator's action are involved. In order to utilize this method, it is necessary to perform some post-processing for user's convenience. This study demonstrated the results of post-processing techniques for scenarios generated through DICE™ with an example of LBLOCA (Large Break Loss of Coolant Accident).

Keywords: Post-processing, MCET, DICE™, LBLOCA.

1. Introduction

DICE™ (Dynamic Integrated Consequence Evaluation) is a software that handle both of DDET (Discrete Dynamic Event Tree) method and MCET (Monte Carlo Event Tree) method for analyze dynamic events happening in a nuclear power plant. By conducting simulations of events using these two methods, it is expected to draw insights from the results revealed by contextual interaction between a physical model and a stochastic behavior of components or human operators. When simulating an accident situation using the MCET method, the variable results will be obtained for repeated simulations, which means it is necessary to be able to view data on when and which equipment or system operated in each simulation, as well as data on how physical variables changed over time. DICE have shown outcomes as a dynamic event analyzer, but, due to this necessity, a post-processing module was necessitated to generate information about the events each simulation has gone through, changes in physical variables, and the outcomes of all simulations for specific physical variables. In other words, the output contains information on when and what events the simulation went through, how many equipment or systems operated during each event, and if operator intervention occurred, when it happened. It also includes the state of the reactor at the end of the simulation and how the physical variables of the reactor changed during the simulation. In this study, the post-processing module was used to analyze the results of calculations using MCET method of DICE for the LBLOCA (Large Break Loss of Coolant Accident).

2. DICE™

In the conventional PSA, analysts determine the event junctions in advance based on thermal-hydraulic calculations, making it difficult to incorporate time-series accidents and operator interventions into the analysis. To address this difficulty, DICE is being developed, which can select either the Discrete Dynamic Event Tree (DDET) or Monte Carlo Event Tree (MCET) methods with identical input sets. Both generally share the algorithms, but there are some distinctions originated from the method how to assign probability distributions at branching points. The DDET method analyses the scenario by dividing the simulation into time steps and checking the branch rules to generate a branch if the branch rules are satisfied. The MCET method is similar to the DDET method in that it divides the process into time steps and checks whether the branching rule is satisfied. However, when generating branches, instead of generating all possible branches, it selects only one possible branch and conducts repeated simulations to analyze a scenario. For this purpose, DICE has a scheduler, physical module, diagnosis module, and reliability module. The physical module uses MARS-KS or MELCOR to calculate the thermal-hydraulic variables of the plant during the time step, and the calculated variables are transmitted to the diagnosis module through the scheduler. The diagnosis module checks whether the transmitted variable values satisfy the branching rules for automatic or manual action of the plant's system and equipment, and if the branch rules are satisfied, generates a branch, and allocates the calculation result of the physical module to the branch. In addition, the reliability module judges the failure and recovery of the system and equipment, taking into account the failure rate of the system and equipment that satisfy the branching rules. When a branch is generated through this process, information about the branch is transmitted back to the scheduler, and the calculation of the branch is resumed through the physical module according to the transmitted information. In other words, DICE can divide the time step, and information exchange between the physical and diagnostic modules occurs in each time step of the simulation. This allows DICE to reflect the state of the system (component failures or recoveries, operator actions, etc.) at each time step, enabling the derivation of various scenarios.

3. Post-processing module

The ultimate goal of the post-processing module is to group similar scenarios together. To achieve this, it processes the changes in monitoring variables obtained through scenario analysis in a way that is easy to understand. By graphically representing the values of the monitoring variables and comparing them, the module aims to identify scenarios that exhibit similar patterns. It is programmed based on the Python language and have three functions as shown in Fig.1.

The first function of the post-processing module is to facilitate the monitoring of the variable changes for each iteration. Since DICE uses MARS-KS or MELCOR as its physical module. DICE stops their calculation and creates a restart file when the branching rule is satisfied, it stores the calculation results up to the point where the restart file is created in, for instance, rstplt files. Therefore, to check the overall result of the monitoring variables that have gone through each iteration, the result values stored for each branch of the iteration must be combined after all calculations are completed. To do this, the post-processing module is used to combine the results stored for each branching into a single file. This file contains the results of various monitoring variables that are saved over time.

The second function of the post-processing module is to allow the user to investigate the results of a specified monitoring variable for every iteration and generate a file containing all the results of a specified monitoring variable. Since the system and equipment may vary during each iteration of the simulation, this feature allows the user to observe how the monitoring variable changes accordingly. Additionally, to make it easier to visualize these changes, there is a feature that displays the results in a graph. Furthermore, there is a functionality that displays the distribution of the monitoring variable's results at a specific point in time. This enables grouping of similar iterations based on the monitoring variable's results at that specific point in time.

The third function of the post-processing module is to display the event sequence. The event sequence shows which branch was followed when the iteration satisfied the branching rule. In this function, information is provided about how many devices were operational at the branching point, when the operator's intervention was taken, and finally, the information on the final result of that iteration. In an event sequence file, the iteration number, the time when the devices operated, the type and number of devices that operated, and the last state of the reactor (U for Unknown, D for core Damage, N for Normal) are recorded. By checking the event sequence with the results of the monitoring variables obtained through the first or second function, it is easy to determine how the progression of the accident changes depending on which devices operated at what point in time.

4. Results

For the analysis of an LBLOCA, the MCET method of DICE was used with an iteration count of 30. The typical PWR was used as the reference model, which has three SIT (Safety Injection Tanks) and three LPSI (Low Pressure Safety Injection) pumps. The diameter of the cold leg break was set to 20 cm for the LBLOCA implementation. The branching points were set to the start of SIT injection and the start of LPSI pump operation. The criterion for starting SIT injection was set to when the pressure of the three accumulators fell below 4.3 MPa, and the criterion for starting LPSI pump operation was set to when the pressure of the pressurizer fell below 12.5 MPa and the pressure of the three steam lines fell below 4.1 MPa. Since the iteration count is 30, if the actual failure rates of the SITs and LPSI pumps are used for analysis, events where these devices fail occur very rarely. Therefore, the failure rates of these devices were increased by a factor of 100 arbitrarily for the event analysis.

Using the post-processing module, the results of the monitoring variables for iteration 30 were analyzed. The first function was used to create a single file containing the results of monitoring variables for each iteration, such as the temperature, pressure and flow rate of the hot leg and cold leg, water level of vessel, and so on. When using the second function, among the monitoring variables, PCT (Peak Cladding Temperature) was selected, and the changes in PCT and the distribution of iterations based on PCT at a specific point were examined. Under

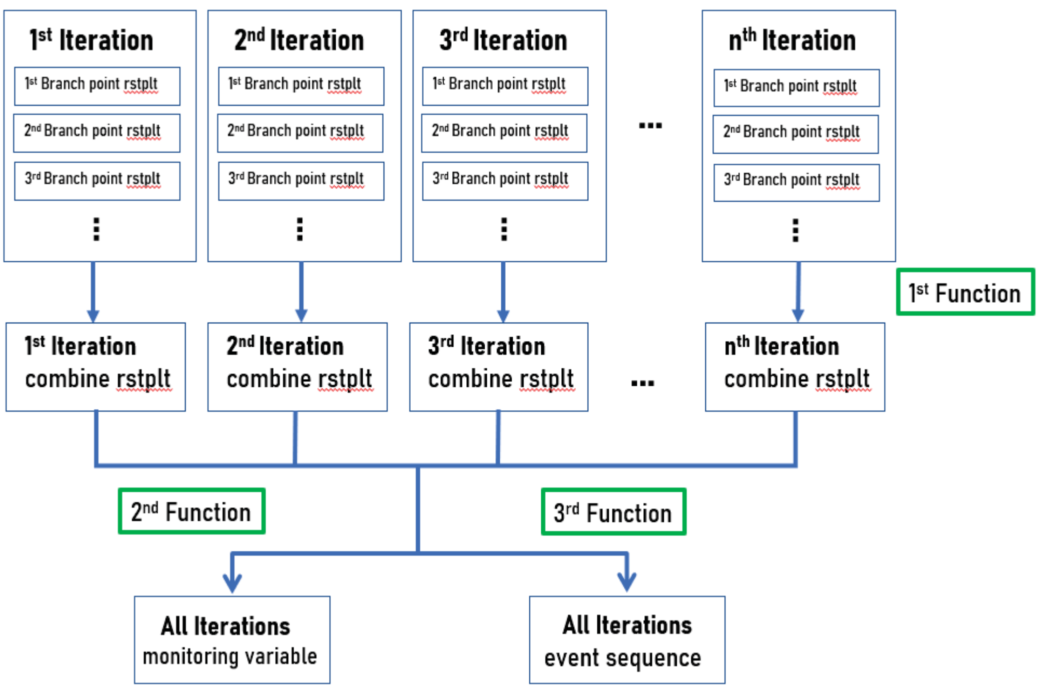

Fig. 1. Diagram of Post-processing Module

normal conditions, the PCT showed 658 ℃. However, when LBLOCA occurs, the temperature of PCT increases sharply for all iterations. Afterward, the SIT and LPSI pumps start operating, and depending on the number of operating SIT and LPSI pumps, the PCT varies as shown in Fig. 2. For example, in the 10th iteration (I10), all three SIT pumps were operational, but none of the LPSI pumps were operating. As a result, during the initial phase of SIT operation, the PCT did not increase significantly, but afterwards, it exhibited a sharp increase. On the other hand, in the 17th iteration (I17), none of the SIT pumps were operational, resulting in a rapid increase in PCT during the initial phase. However, after all three LPSI pumps started operating, there was a sharp decrease in PCT. To group iterations that exhibit similar patterns, it is necessary to examine the distribution of iterations at specific time points. Fig. 3. represents the distribution of iterations at a 50-second interval, where it can be observed that the iterations are divided into three distinct groups. By examining the distribution of iterations at multiple time points and consistently observing them being classified into the same group, it is possible to group the iterations. Similar results can be obtained by applying the same approach as mentioned above, where the final state of the reactor in Fig. 4. is divided into either "U" or "D".

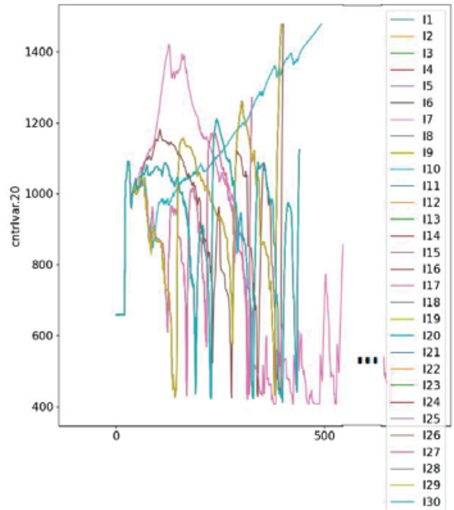

Fig. 2. PCT variation depending on the operating number of the SIT and LPSI pumps under LBLOCA

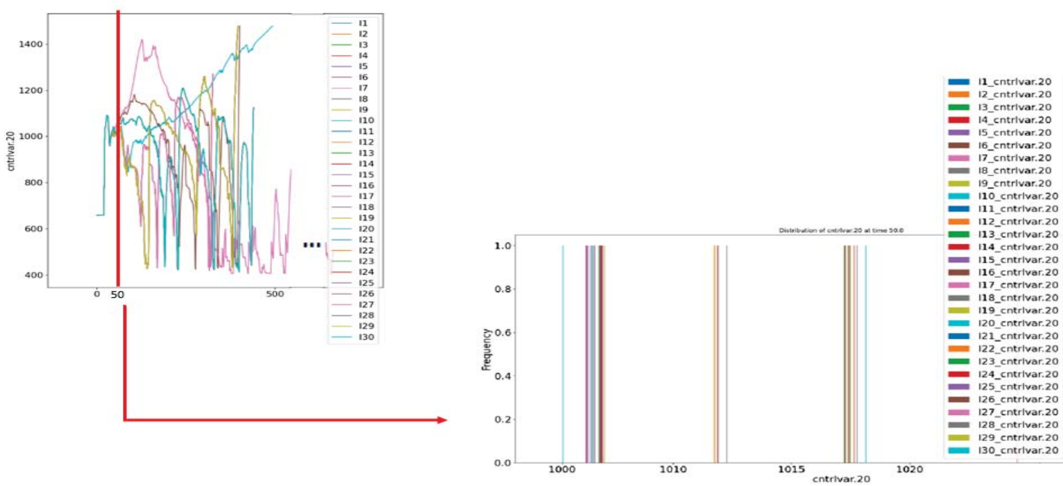

Fig. 3. Distribution of iterations according to PCT at 50 seconds

I1	23.2725	A_2_4	32.7301	A_1_3	U
I2	23.2725	A_2_4	32.7301	A_1_3	U
I3	23.2725	A_2_2	32.7301	A_1_2	U
I4	23.2725	A_2_3	32.7301	A_1_4	D
I5	23.2725	A_2_4	32.7301	A_1_4	U
I6	23.2725	A_2_4	32.7301	A_1_3	U
I7	23.2725	A_2_3	32.7301	A_1_4	D
I8	23.2725	A_2_4	32.7301	A_1_3	U
I9	23.2725	A_2_4	32.7301	A_1_3	U
I10	23.2725	A_2_1	32.7301	A_1_4	D
I11	23.2725	A_2_4	32.7301	A_1_4	U
I12	23.2725	A_2_3	32.7301	A_1_2	U
I13	23.2725	A_2_4	32.7301	A_1_4	U
I14	23.2725	A_2_4	32.7301	A_1_3	U
I15	23.2725	A_2_3	32.7301	A_1_4	D
I16	23.2725	A_2_4	32.7301	A_1_2	D
I17	23.2725	A_2_4		E	
I18	23.2725	A_2_4	32.7301	A_1_3	U
I19	23.2725	A_2_4		E	
I20	23.2725	A_2_2	32.7301	A_1_4	U
I21	23.2725	A_2_4	32.7301	A_1_4	U
I22	23.2725	A_2_4	32.7301	A_1_4	U
I23	23.2725	A_2_3	32.7301	A_1_4	D
I24	23.2725	A_2_3	32.7301	A_1_4	D
I25	23.2725	A_2_3	32.7301	A_1_4	D
I26	23.2725	A_2_4		E	
I27	23.2725	A_2_4	32.7301	A_1_4	U
I28	23.2725	A_2_3	32.7301	A_1_2	U
I29	23.2725	A_2_3	32.7301	A_1_4	D
I30	23.2725	A_2_4	32.7301	A_1_3	U

Fig. 4. Event sequence for 30 iterations

5. Conclusions

Through the post-processing module, it was possible to process the results of the monitoring variables calculated through the MCET method of DICE and group them simply through iterations. In the case of LBLOCA, since only SIT and LPSI pumps are used to mitigate the accident without operator intervention, the event sequence was not diverse. Therefore, when grouping with the post-processing module, only two large groups were formed. In future research, we plan to analyze various accident scenarios involving different operator interventions and attempt to group them using algorithms such as a mean-shift method when grouping is not clearly defined.

Acknowledgement

This work was supported by the Nuclear Safety Research Program through the Korea Foundation Of Nuclear Safety(KoFONS) using the financial resource granted by the Nuclear Safety and Security Commission(NSSC) of the Republic of Korea. (No. 2103081).

References

Aldemir, T. (2018). *Advanced concepts in nuclear energy risk assessment and management (Vol. 1).* World Scientific.

Heo, G., Baek, S., Kwon, D., Kim, H., Park, J. (2021). *Recent research towards integrated deterministic-probabilistic safety assessment in Korea*. Nuclear Engineering and Technology, 53(11), 3465-3473.

Baek, S., Heo, G., Kim, T., & Kim, J. (2021). *Numerical Verification of DICE (Dynamic Integrated Consequence Evaluation) for Integrated Safety Assessment*. In 31st European Safety and Reliability Conference, ESREL 2021 (pp. 2385-2390).

Osborn, D. M., Aldemir, T., Denning, R., & Mandelli, D. (2013). *Seamless Level 2/Level 3 dynamic probabilistic risk assessment clustering*. Proc. Int. Topl. Mtg. Probabilitistic Safety Assessment and Analysis (PSA 2013), 22-27.