Development of NSSS Simulation Engine for SMART Simulator Using the Best Estimate Code, MARS3.1

K. D. Kim [a*], S. W. Lee [a], Sung-Chul Lee [a], Yong-Suk Suh [a], Jae-Seung Suh [b]
[a]*Korea Atomic Energy Research Institute, 1045 Daedeok-daero, Yuseong-gu, Daejeon, 305-353, Korea*
[b]*SENTECH, 575 Yongsan, Yuseong-gu, Daejeon, 305-500, Korea*
[*]*Corresponding author: kdkim@kaeri.re.kr*

1. Introduction

Limited computational capability and crude thermal-hydraulic modeling in early 1980s forced the use of overly simplified physical models and assumptions for a real-time calculation at the cost of fidelity. Rapid advances in computer technology make it possible to improve the fidelity of the simulator models. These efforts have been made based on RELAP5 [1] in the US, and CATHARE2 [2] in France. The NSSS thermal-hydraulic engines adopted in the most domestic full-scope power plant simulators have been replaced with RELAP5 based engines which were provided by US vendors. Since the technology dependency of the NSSS T/H engine by foreign vendors, it may cause difficulties in maintenance and model improvement.

KAERI has started to develop a realistic NSSS calculation engine based on the best-estimate code MARS 3.1 for the SMART full-scope simulator. Even though we are developing the NSSS calculation engine for SMART [3] simulator, it can be easily extended to light water reactors and GEN-IV reactors, etc.

The verification of the NSSS calculation engine for SMART simulator has been conducted by an integrated test in the simulator environment, Jade 4.0, developed by GSE [4] of Windows 2003. This paper briefly presents our efforts for the NSSS calculation engine for SMART simulator and verification test results of SAT (Site Acceptance Test).

2. Development of the NSSS T/H engine

The NSSS calculation engine for SMART simulator was developed based on MARS 3.1, which is a best estimate code developed by KAERI for various transient analyses of nuclear power plants [5]. In order to implement the MARS code as an NSSS calculation engine for the full-scope simulator, the NSSS calculation module must communicate with the other simulator modules. We developed the interface program for MARS to communicate with others. Figure 1 shows an overall configuration of the SMART simulator system. The MARS code calculates NSSS calculation phenomena based on boundary conditions received from the other simulator modules. The simulator module of MARS and other modules of the simulator communicate through a global vector. Every module in the simulator writes the calculated values in this vector every 1/24 second. MARS module reads the various boundary conditions such as valve opening areas, pressures, reactor power distribution from a global vector and writes all MARS global variables, and some additional variables which are needed in other modules.

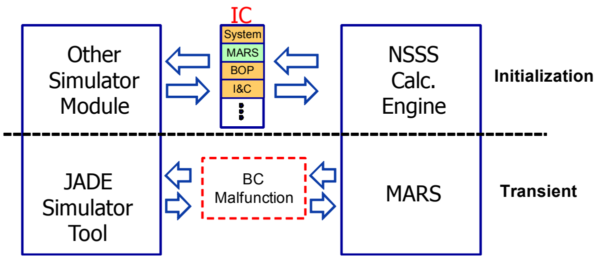

Fig. 1. overall configuration of the SMART simulator system

3. Integration of the NSSS T/H engine in simulator environment

We integrated the NSSS calculation module with other simulator modules developed using the GSE tool. The SAT was performed with a complete set of the full-scope simulator, which includes all hardware and software as shown in Fig. 2.

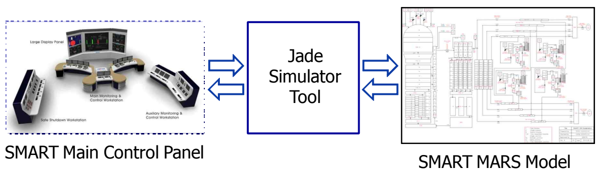

Fig. 2. Schematics of SMART simulator

4. Verification Test of the NSSS T/H engine

The various test simulations have been performed to confirm the real-time calculation capability, robustness and fidelity. While the results must be compared with experiment results and/or the independent calculation results of a best-estimate code, the quantitative analysis have not been performed since MARS 3.1 calculation parts are used without modification. Instead, we confirmed that the same calculation results in simulator environment were obtained with the standalone MARS calculation results. The integrated test with other simulator software and hardware panel has also been performed in the simulator environment. To illustrate the real-time simulation and fidelity, the iFix, which can examine the trends of the transient in a window has been used. In the simulator environment, it rejects the module if that module cannot obtain a solution within 1/24 seconds. That means it runs in real-time if MARS module is not rejected. Figure 3 shows the iFix screens for small break LOCA.

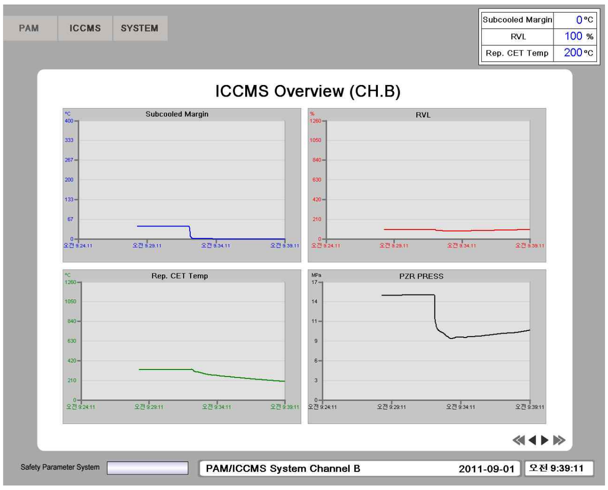

Fig. 3. iFix screen for simulation of small break LOCA

3. Conclusions

In order to examine the applicability of MARS code for the NSSS calculation model of the full-scope simulator, we developed an NSSS simulation module based on a best-estimate code, MARS.

The overall performance in terms of real-time calculation is confirmed without major changes of MARS code. Since the NSSS calculation module is developed based on a best-estimate code, its fidelity is already certified. Since this is the first development of the simulator's NSSS T/H engine based on a domestic best-estimate code, it may not be perfect and may require further efforts for maintenance. However, we confirmed that it is possible to replace the NSSS calculation engine of full-scope simulator with MARS based module.

REFERENCES

[1] RELAP5 code development team, RELAP5/MOD3 code manual, NUREG/CR-5355, U. S. Nuclear Regulatory Commission (1995).
[2] F. Barre et al., The CATHARE code strategy and assessment, Nuclear Engineering and Design, 124, 1990.
[3] KAERI Internal Document, 000-PL403-001, S. Choi, et al., "SMART Design Description" 2010. 1. 29.
[4] SimExec user guide for Windows NT operating system, GSE power systems, Inc.
[5] B. D. Chung, et al., "MARS code manual volume 1: Code Structure, System Models and Solution Methods", KAERI/TR-2812/2004, 2004.