# Development of fully coupled FRAPTRAN with MARS-KS code system for calculation of fuel behavior during LOCA

Hyochan Kim[1], Changhwan Shin[1], Yongsik Yang[1]
*[1]Nuclear Fuel Safety Research Division, Korea Atomic Energy Research Institute, 111 Daedeok-daero 989beon-gil, Yuseong-gu, Daejeon, 34057, ROK*

Taewan Kim[2]
*[2]Department of Safety Engineering, Incheon National University, 119 Academy-ro, Yeonsu-gu, Incheon, ROK*

ABSTRACT

During a loss of coolant accident (LOCA), fuel behaviors such as ballooning and burst affecting the safety criteria as well as high burnup fuel have different characteristics such as the thermal conductivity, initial oxide thickness and so on. Therefore, fuel models should be required for a safety analysis. In this work, MARS-KS1.4 which is an audit code for a safety analysis in KOREA, has been coupled with FRAPTRAN2.0 to take into account the fuel behavior and fuel properties with respect to burnup. New coupling methodology was proposed to maintain the calculation methodology of each code. FRAPTRAN2.0 was modulized into the so-called S-FRAPTRAN. To verify the coupled code system, a safety analysis was performed using the hypothesis LBLOCA scenario of OPR1000 under BOL/MOL/EOL fuel conditions. The calculation results demonstrate that the fuel behavior affects the safety criteria, such as the PCT (peak cladding temperature) and ECR (Equivalent Cladding Reacted).

## 1. Introduction

The study of fuel behaviour under accidental conditions is a major concern in the safety analysis of pressurized water reactors (PWRs). In particular, the consequences of a loss of coolant accident (LOCA) have to be investigated and quantified in comparison to the related safety criteria already defined, in order to prevent severe core damage that could result from a fuel rod failure, fuel ejection into coolant, loss of core coolability, and fission product release into the primary circuit. Those criteria were established in the 70s on the basis of several experimental programs performed with fresh or low burnup irradiated fuel.
However, since the early 90s, economic concerns have led utilities to consider the increase of the average burnup of the fuel subassemblies and the use of new types of fuel and cladding materials, in view of optimizing the fuel management. At the present time, the increased industrial competition and constraints result in more aggressive conditions for the fuel (higher burnup, higher power, load follow,..) and create incentive conditions for the development of advanced fuel designs with improved performance (new fuel types with additives, cladding material with better resistance to corrosion,…) [1]. These long anticipated developments involved the need for new investigations of irradiated fuel behaviour under reference accidents in order to check the adequacy of the current criteria, evaluate the safety margins, provide new technical bases for modelling, and allow an evolution of these criteria.
Recently, the ECCS (emergency core cooling system) acceptance criteria (10CFR50.46c) will be revised in Korea [2]. In the revised criteria, the safety analysis

code system should take into account the fuel transient behavior. During LOCA, previous researchers have reported that the fuel rod undergoes thermo-mechanical deformation of the cladding (ballooning), exothermic high-temperature oxidation, cladding burst, and FFRD (fuel fragmentation, relocation and dispersion).
Therefore, many researchers have been developing the coupled code system between the thermal hydraulic and fuel for a transient analysis. The U.S. NRC developed the coupled TRACE/FRAPTRAN/DAKODA code system to study the fuel rod behavior and uncertainty during LBLOCA [3]. However, its methodology was limited as one-way coupling. IRSN developed the DRACCAR code system, which is a multi-rod 3D thermo-mechanics code, with mechanical and thermal interactions between rods, coupled with subchannel type two-phase flow codes [4]. EDF developed CATHS-BI, which considers the fuel behavior and its relative thermal hydraulic models (flow blockage, hot rod clad-coolant exchange surface reduction and fuel relocation). The coupled results show that fuel models affect the safety criteria remarkably. In Korea, MARS-KS1.4 was coupled with FRAPTRAN1.4 as well [6]. They applied the DLL (Dynamic-Link Library) method of the FRAPTRAN code, and the time step of the fuel does not match that of the system code owing to a limitation of coupling.
In this work, fully coupled MARS-KS1.4/FRAPTRAN2.0 has been developed to simulate the fuel behavior. The fully coupled code demonstrates that the modulized FRAPTRAN2.0 was implemented into MARS-KS as a subroutine. However, each code calculation methodology should be kept so that a new coupling methodology was proposed. To implement the FRAPTRAN2.0 into MARS-KS1.4, FRAPTRAN2.0 was modulized into the so-called S-FRAPTRAN. To verify the coupled code system, the hypothetical LBLOCA scenario for OPR1000 was simulated with the BOL (begin of life)/MOL (middle of life)/EOL (end of life) fuel conditions.

## 2. Development of MARS-KS/FRAPTRAN coupled code system

### 2.1. MARS-KS 1.4 and FRAPTRAN 2.0

The Fuel Rod Analysis Program Transient (FRAPTRAN) is a Fortran language computer code that calculates the transient performance of light-water reactor fuel rods during reactor transients and hypothetical accidents such as loss-of-coolant accidents, anticipated transients without a scram, and reactivity-initiated accidents. FRAPTRAN calculates the temperature and deformation history of a fuel rod as a function of the time-dependent fuel rod power and coolant boundary conditions. Although FRAPTRAN can be used in “standalone” mode, it is often used in conjunction with, or with input from, other codes. The phenomena modeled by FRAPTRAN include a) heat conduction, b) heat transfer from the cladding to coolant, c) elastic-plastic fuel and cladding deformation, d) cladding oxidation, e) fission gas release, and f) fuel rod gas pressure. FRAPTRAN is programmed for use on Windows-based computers, but the source code may be compiled on any other computer with a Fortran 2008 and newer compiler. The burnup-dependent parameters may be initialized from the FRAPCON steady-state single rod fuel performance code [7]. In 2017, FRAPTRAN 2.0 was released.
The MARS-KS (Multi-dimensional Analysis of Reactor Safety) code has been developed by KAERI for a multi-dimensional and multi-purpose realistic thermal-hydraulic system analysis of light water reactor transients. The backbone of the code has been built by unifying and restructuring the RELAP5/MOD3 and COBRA-TF1 codes [8]. The MARS-KS code has the capability of analyzing a one-dimensional and

three-dimensional thermal-hydraulic system, as well as the fuel responses of light water reactor transients. The thermal hydraulic modeling capability of the MARS code has been continuously improved and extended for an application, not only to light and heavy water reactors but also to research reactors and to many advanced reactor types. Many improved models and capabilities were added to the code, and the latest version of the series is the MARS-KS 1.4. Notable upgrades include 3-dimensional simulation capabilities incorporated into the latest version in order to treat a turbulent mixing model and a conduction model. MARS-KS has been mainly used for regulatory activities by the Korea Institute of Nuclear Safety (KINS).

### 2.2.Coupling methodology

To develop the MARS-KS/FRAPTRAN code system, the coupling methodology should be defined because each code system was already used and validated with their own methodology. As shown in Fig. 1, we proposed a coupling methodology of two codes for a steady state and transient maintaining each calculation flow and I/O (Input/Output) system.
At the beginning of a fully coupled calculation, MARS-KS performs a steady state calculation with an input file. For this calculation, MARS-KS employs its heat structure instead of a fuel rod. We call the first SS (steady state) calculation, which performs a null transient calculation without FRAPCON/FRAPTRAN. Once MARS-KS completes the first SS, it calls S-FRAPTRAN (Simplified-FRAPTRAN), which is modulized FRAPTRAN to be implemented into MARS-KS. For the first calling, S-FRAPTRAN initiates the input variables and stores the FRAPCON result file to apply burnup dependent variables. The S-FRAPTRAN starts the fuel stabilization, which gradually increases the power to stabilize the fuel thermo-mechanical behavior. We call the second SS for fuel stabilization. Once the fuel stabilization is completed, fully coupled MARS-KS/FRAPTRAN is ready to start the transient calculation for LOCA.

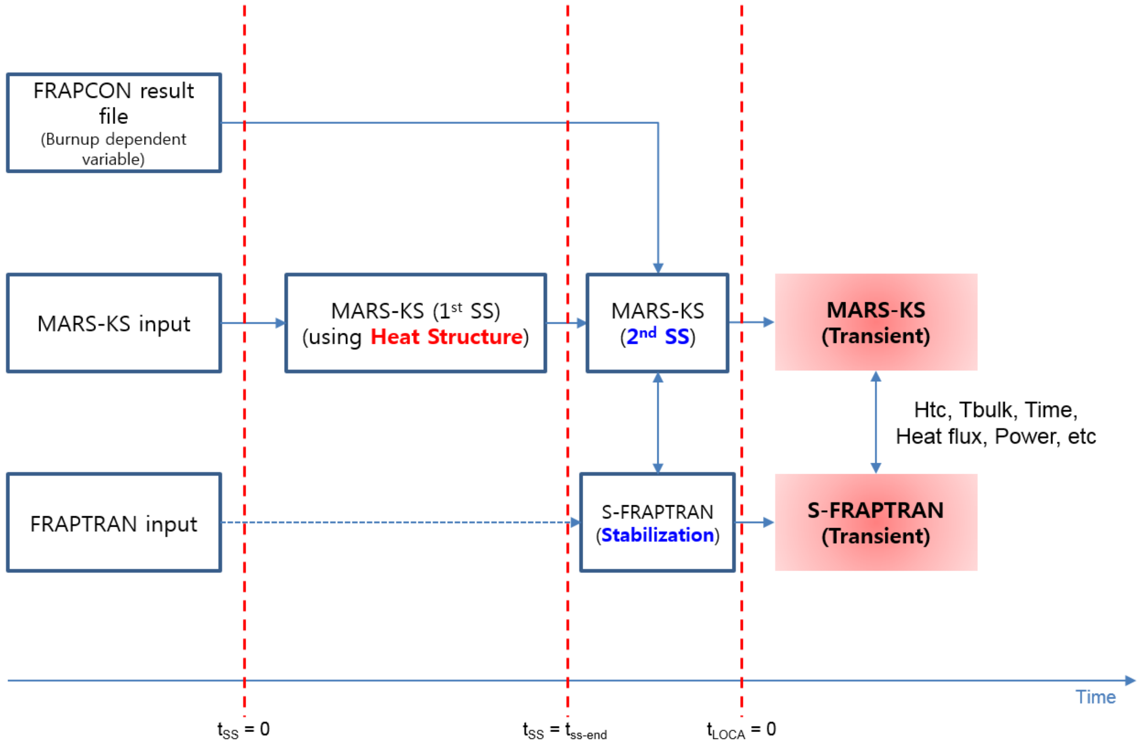



*Fig. 1. Coupling methodology of fully coupled MARS/FRAPTRAN.*

TABLE 2: Coupled variables of MARS/FRAPTRAN code system.

| Calling module | Variable name | Content |
|---|---|---|
| **S-fraptran** | Timeincrement | Size of Time step |
| | Power | Linear Heat Generation Rate (LHGR) |
| | CoolPress | Coolant pressure |
| | Htc | Heat transfer coefficient of cladding surface |
| | Tbulk | Coolant T |
| **MARS-KS** | Outdia | Cladding outer diameter (incl. oxide thickness) |
| | Heatflux | Cladding heat flux |
| | Tsurf | Cladding surface T |

Coupling variables between MARS-KS and FRAPTRAN were defined as shown in Table 2. For the current time step, MARS-KS calculates the time increment, LHGR, coolant pressure, heat transfer coefficient, and coolant temperature. All variables are stored for S-FRAPTRAN calculation. Subsequently, S-FRAPTRAN calculates the deformed cladding diameter, heatflux, and cladding surface temperature. Those variables are also stored for MARS-KS calculation for the next time step.

The fuel module requires power and thermal hydraulic boundary conditions at the surface of the outer cladding to calculate the thermo-mechanical behavior of fuel during a LOCA. In addition, the coolant pressure affects the cladding deformation. The system code requires the outer diameter of the cladding and heatflux considering the radial burnup distribution, gap conductance, and metal water reaction energy. All variables are stored in the module and updated at each time step.

### 2.3. Modulization of FRAPTRAN (S-FRAPTRAN)

The FRAPTRAN2.0 code was modulized as S-FRAPTRAN to implement FRAPTRAN into MARS-KS. Basically, FRAPTRAN2.0 was modernized into Fortran90. Its environment should be identical to MARS-KS V1.4 as Intel Visual Fortran Composer XE2013 Update2.

To couple the variables of two codes, a new module (MARSLINK) was created in S-FRAPTRAN. When the subroutine uses this module, the subroutines are able to access the coupled variables, as shown in Table 3. In addition, new subroutines were added into S-FRAPTRAN to obtain new variables from MARS-KS. Some of the subroutines were eliminated because the thermal hydraulic calculations were carried out in MARS-KS. However, the I/O system was maintained to easily use the I/O file for a fuel code user and MARS-KS code user.

TABLE 3: Additional module and subroutine in S-FRAPTRAN.

| Module | Subroutine | Function |
|---|---|---|
| **MARSLINK** | MARSDT | Time step of MARS-KS was stored |
| | MARShtctbulk | Heat transfer coefficients and bulk temperatures of axial nodes were stored |
| | MARSPKW | LHGR of MARS-KS at each node was stored |

| | | |
|---|---|---|
| | GetPMax | Max LHGR was found for fuel stabilization |
| | MARSCOOLPRS | Coolant pressures of axial nodes were stored |
| | FTRANVAR | Variables calculated by MARS-KS were stored for fuel calculation |
| | checkvari | Debug the coupling variables |

When MARS-KS calls S-FRAPTRAN for the current time step, the fuel calculation begins with the input file. Figure 2 shows flowchart of S-FRAPTRAN, which is identical to that of FRAPTRAN except for the thermal hydraulic calculation. If it is the first call, S-FRAPTRAN initializes the variables with the input file and FRAPCON file to take into account the burnup dependent effects. At the first call, fuel stabilization was carried out and max LHGR among all axial nodes was searched to divide the power step for a steady state calculation. After that, the transient calculation begins. Unlike FRAPTRAN Stand Alone, MARS-KS controls the time increment. Therefore, the time increment from MARS-KS is added for calculation. In addition, LHGR and the coolant pressure of all nodes are given from MARS-KS. Once the fuel calculation ends, the main subroutine of MARS-KS begins.

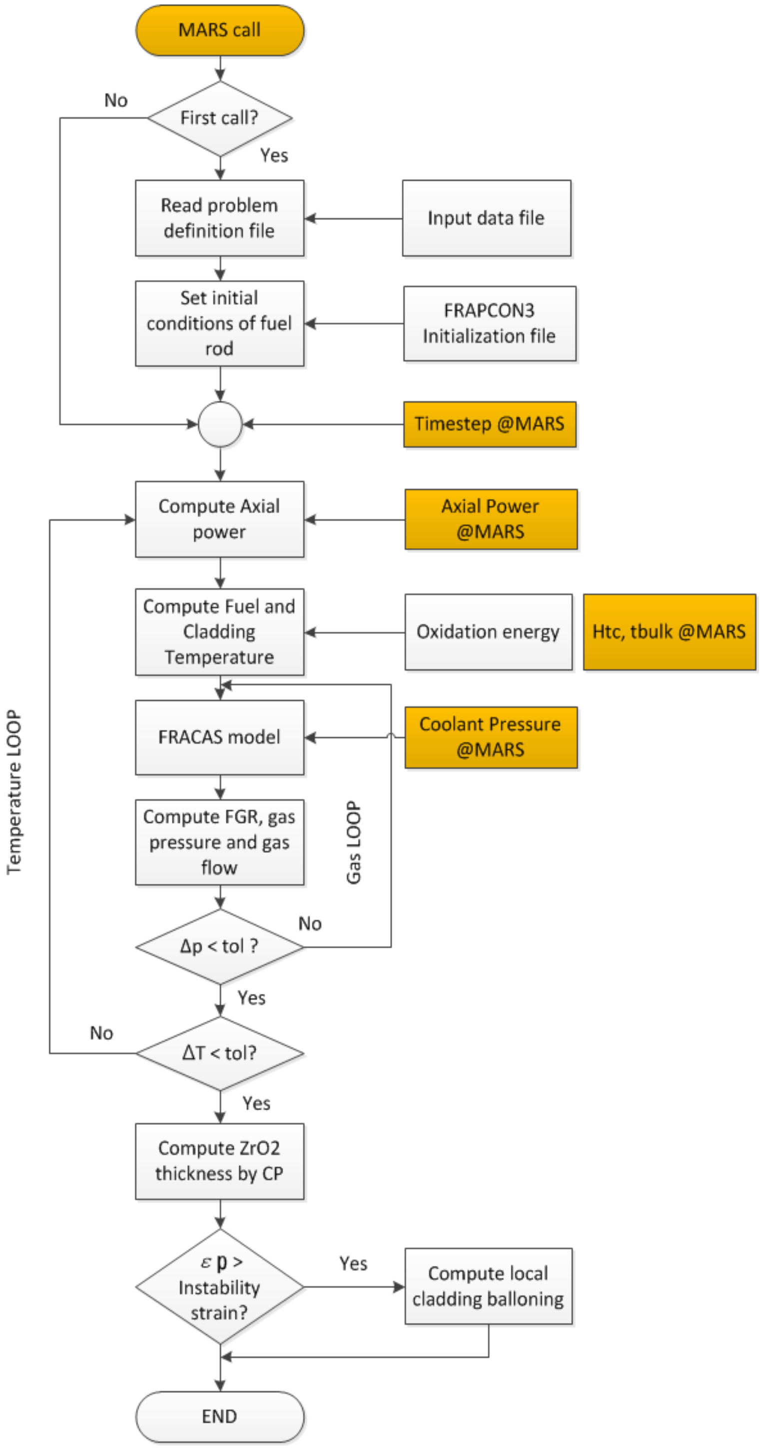



*Fig.2. Flowchart of S-FRAPTRAN.*

## 3. Calculation results by MARS-KS/FRAPTRAN coupled code system

### 3.1. Verification of the coupled variables

Once S-FRAPTRAN was implemented into MARS-KS, a preliminary test was carried out under LBLOCA input conditions. The case was run successfully, and the coupled variables were checked by debugging the subroutine. At each time step, it was verified that coupled variables were stored in memory, and S-FRAPTRAN uses the stored variables at each time step. Figure 3 shows the variables calculated by MARS-KS. In the case of the standalone (S.A.) calculation FRAPTRAN, the heat transfer coefficient and time pairs should be inserted into the input file. In addition, the number of pairs are limited owing to the array size. However, the coupled code overcomes the limitation of the S.A. calculation by the coupling simultaneously. In addition, the axial powers and coolant pressure at the axial nodes are stored at the current time.

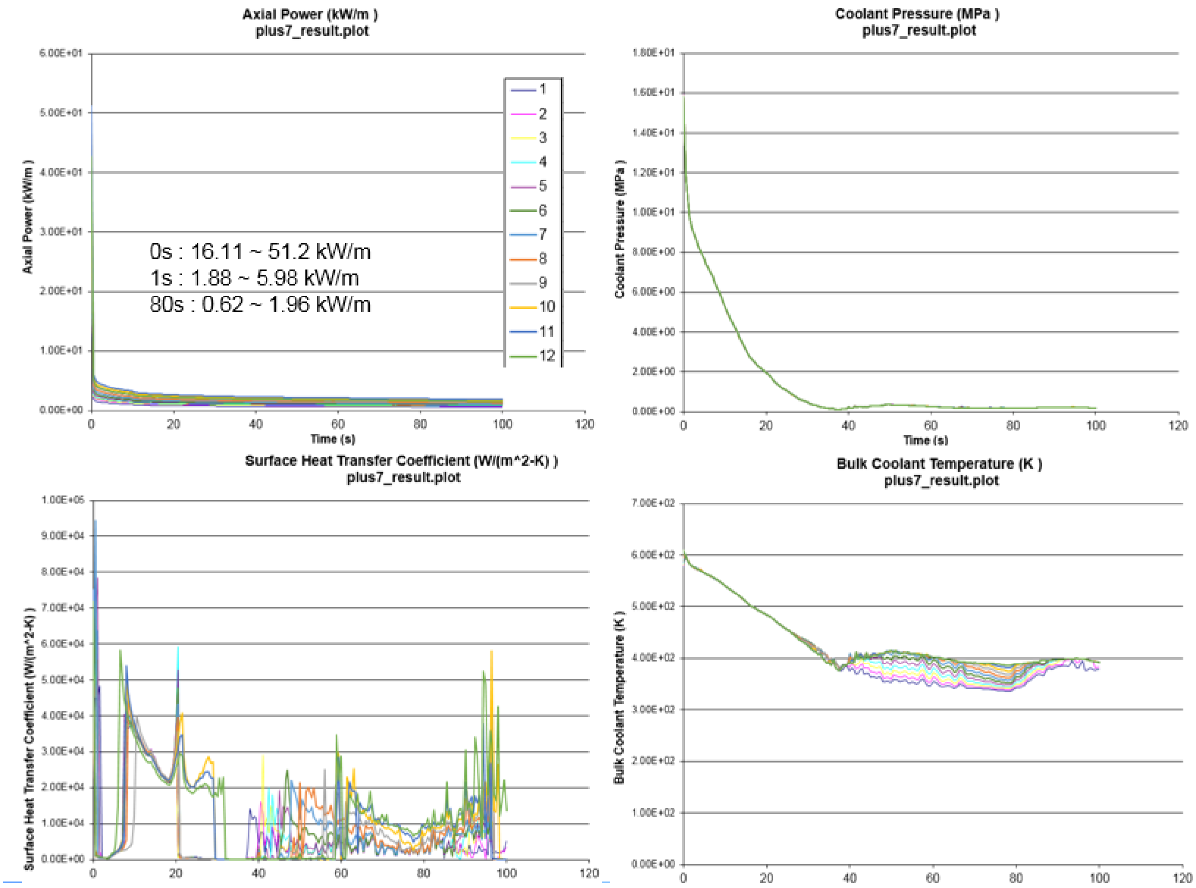



*Fig.3. Variables calculated by MARS-KS for S-FRAPTRAN.*

### 3.2. Calculation results of coupled code

To evaluate the coupled MARS/FRPATRAN code system, it runs the LOCA input deck including the system code and fuel input.
The input decks of MARS-KS are as follows: hypothetical LBLOCA scenario of Units 3/4 of the Hanul plant, non-rezoning option for a reflood, hot pin power, time step size fixed at $10^{-3}$s, and a conservative decay power applied, one core channel and three heat structures (hot pin, hot assembly, average core) were modelled.
The input decks of FRAPTRAN are as follows: the dimensions of the PLUS7$^{TM}$ fuel rod were applied, the cathcart-pawel model for high-temperature oxidation was used, the balon2 model was turned on, the burnup of BOL is 0 MWD/kgU, the burnup of MOL is 35.03 MWD/kgU, and the burnup of EOL is 64.55 MWD/kgU.
To simulate burnup effects of fuel, average rod power (LHGR) for steady state was set as follows; average LHGR of the 1$^{st}$ cycle, that of the 2$^{nd}$ cycle and that of the 3$^{rd}$ cycle are 23.5 kW/m, 21 kw/m and 17 kW/m, respectively. The LHGRs were obtained by neutronics code.
As a result of the coupled code, figure 4 shows the cladding outer temperature of each axial node over time. For this scenario, the peak cladding temperature (PCT) occurs during the blowdown phase. Owing to the power, T/H condition, and metal

water reaction, PCT occurs at axial node 11. The biggest temperature difference among axial nodes is approximately 600 K. Following the blowdown, the refill and reflood begin. The cladding temperature for a reflood state is not higher than that for a blowdown. The coupled code demonstrates the temperature fluctuation of the cladding well owing to complex T/H conditions during a reflood. The hoop strain and gap conductance of the fuel were also analyzed during a LOCA. In conclusion, it was determined that the fully coupled code system is performed correctly.

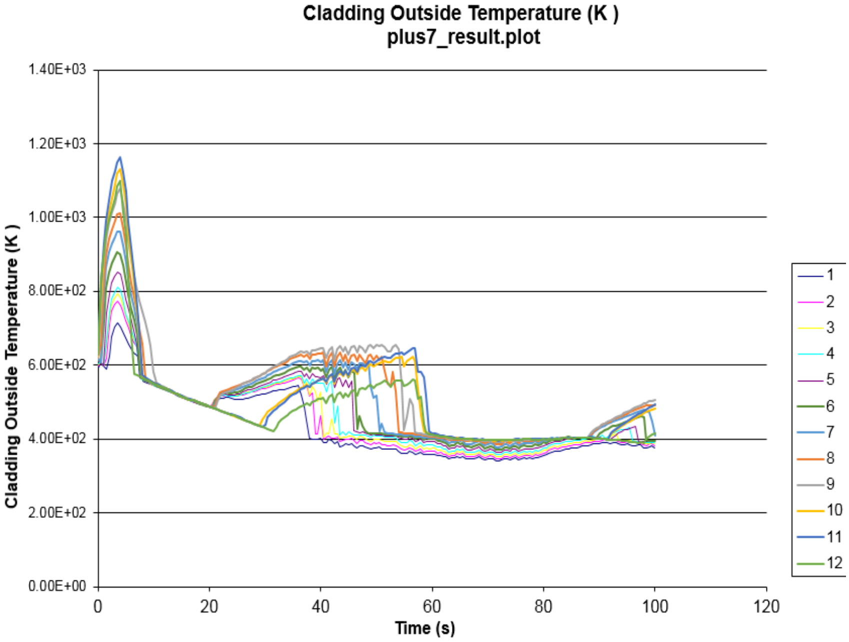
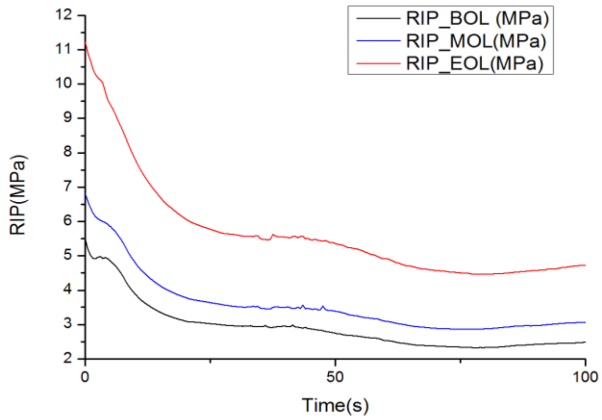


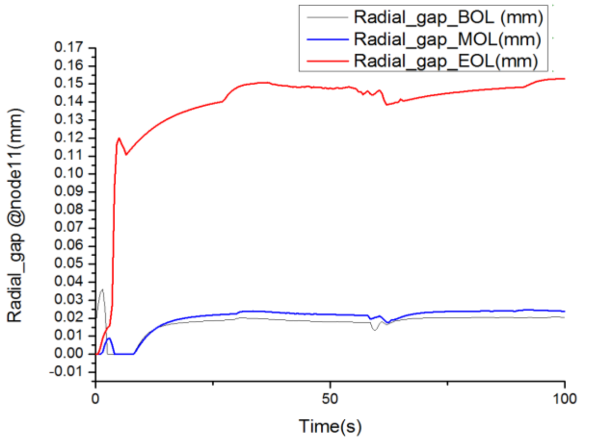

*Fig.4. Calculation results of the fully coupled code with LBLOCA scenario*

To investigate the calculation results with respect to a fuel burnup, the code was calculated along with LBLOCA under BOL/MOL/EOL fuel conditions. As shown in figure 5, the fuel burnup significantly affects the safety criteria (PCT and ECR). Even though input and load conditions of all cases are identical apart from burnup, cladding temperature can rise during reflood phase.

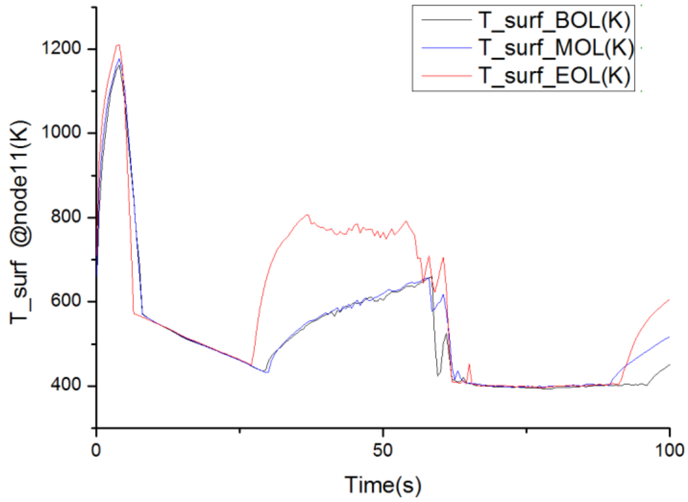
The rod internal pressure (RIP) of the EOL fuel is much higher than those of the others owing to FGR (fission gas release) and void volume change during irradiation. Because a high RIP results in a widening gap between the pellet and cladding during LOCA as shown in figure 5(b), a gap conductance of high burnup fuel becomes much lower than that of BOL. According to gap model, gap conductance is in inverse proportion to the gap width [9]. Eventually, PCT in the reflood phase is much higher than those of the others because the stored energy cannot be removed during a blowdown phase due to a low gap conductance and low thermal conductivity of high burnup fuel. A high PCT can induce high-temperature oxidation of the cladding and its ballooning. Therefore, a safety analysis should take into account the fuel behavior with the fully coupled code system. It was demonstrated that the fully coupled code system is able to simulate the thermal hydraulic behavior and fuel behavior simultaneously.




(a) Rod internal pressure of cases




(b) Radial gap size of cases at node 11




(c) Cladding surface temperature of cases at node 11

*Fig.5. Calculation results with BOL/MOL/EOL fuel conditions*

## 4. Conclusion

The ECCS acceptance criteria will be revised to enhance the reactor safety. To evaluate the revised criteria that incorporate the fuel models, the fully coupled MARS-KS1.4/FRAPTRAN2.0 code system has been developed. We proposed the coupling methodology of two codes because basically the methodology of the two codes should be kept despite the coupling. FRAPTRAN was modulized as S-fraptran to be implemented into MARS-KS. To evaluate the fully coupled code system preliminarily, a hypothetical LOCA input deck was chosen. The coupled results demonstrate that the fully coupled code system is correctly performed.

In the future, the coupled code system will be validated against experimental data to determine whether new models should be developed for a multi-physics simulation.

## 5. References

[1] OECD NEA, Nuclear Fuel Behavior in Loss-of-coolant Accident (LOCA) Conditions: State-of-the-art Report, NEA No. 6846, 2009.

[2] Y.S. Bang et al., Technical Basis for the Revision of ECCS Acceptance Criteria of Domestic PWR Plants, KINS/RR-1686, 2017.

[3] K. Tien, Fuel Rod Behavior and Uncertainty Analysis by FRAPTRAN/TRACE/DAKOTA Code in Maanshan LBLOCA, NUREG/IA-0471, 2016.
[4] G. Repetto, F. Jacq, F. Barr´e, F. Lamare, and J. M. Ricaud, DRACCAR, a new 3D thermal mechanical computer code to simulate LOCA transients on Nuclear Power Plants, in Proceedings of the International Congress on Advances in Nuclear Power Plants (ICAPP ’09), Tokyo, Japan, May 2009.
[5] N. WAECKEL et al, Impact of the new French LOCA limits on PWR LOCA safety analysis, Fuel Safety Research Meeting 2016, Mito, Japan, 2016.
[6] H.C. KIM et al, DEVELOPMENT OF MARS/FRAPTRAN COUPLED CODE AND ITS APPLICATION FOR LOCA ANALYSIS, 2017 Water Reactor Fuel Performance Meeting, Jeju, Korea, 2017.
[7] K.J. Geelhood, W.G. Luscher, J.M. Cuta, I.A. Porter, FRAPTRAN 2.0: A Computer Code for the Transient Analysis of Oxide Fuel Rods, PNNL-19400, Vol.1, Rev.2, 2016.5.
[8] MARS Code manual (Volume I: Code Structure, System Models, Solution Methods), KAERI/TR-2812, 2004.
[9] A.M. Ross and Stoute R.L., Heat transfer coefficient between UO2 and Zircaloy-2, AECL-1552, 1962.