Journal of KOSSE. (2022. 12) Vol. 18, No. 2 pp. 75-93
DOI: https://doi.org/10.14248/JKOSSE.2022.18.2.075
www.kosse.or.kr

ISSN (print) : 1738-480X
ISSN (online) : 2288-3592

A SE Approach for Real-Time NPP Response Prediction under CEA Withdrawal Accident Conditions

Felix Isuwa Wapachi[1)], Aya Diab[1),2)*]

1) Nuclear Engineering Department, KEPCO International Nuclear Graduate School, Ulsan, South Korea,

2) Mechanical Power Engineering Department, Faculty of Engineering, Ain Shams University, Cairo, Egypt.

Abstract : Machine learning (ML) data-driven meta-model is proposed as a surrogate model to reduce the excessive computational cost of the physics-based model and facilitate the real-time prediction of a nuclear power plant's transient response. To forecast the transient response three machine learning (ML) meta-models based on recurrent neural networks (RNNs); specifically, Long Short Term Memory (LSTM), Gated Recurrent Unit (GRU), and a sequence combination of Convolutional Neural Network (CNN) and LSTM are developed. The chosen accident scenario is a control element assembly withdrawal at power concurrent with the Loss Of Offsite Power (LOOP). The transient response was obtained using the best estimate thermal hydraulics code, MARS-KS, and cross-validated against the Design and control document (DCD). DAKOTA software is loosely coupled with MARS-KS code via a python interface to perform the Best Estimate Plus Uncertainty Quantification (BEPU) analysis and generate a time series database of the system response to train, test and validate the ML meta-models. Key uncertain parameters identified as required by the CASU methodology were propagated using the non-parametric Monte-Carlo (MC) random propagation and Latin Hypercube Sampling technique until a statistically significant database (181 samples) as required by Wilk's fifth order is achieved with 95% probability and 95% confidence level. The three ML RNN models were built and optimized with the help of the Talos tool and demonstrated excellent performance in forecasting the most probable NPP transient response. This research was guided by the Systems Engineering (SE) approach for the systematic and efficient planning and execution of the research.

Key Words : Recurrent Neural Network (RNN), Long Short Term Memory (LSTM), Gated Recurrent Unit (GRU), Convolutional Neural Network (CNN), Machine Learning (ML), Best Estimate Plus Uncertainty (BEPU)

Received : October 11, 2022 / **Revised** : December 19, 2022 / **Accepted** : December 21, 2022

* 교신저자 : Aya Diab / KEPCO International Nuclear Graduate School / aya.diab@kings.ac.kr

1. Introduction

Machine Learning (ML) has been gaining popularity in many fields, including applications to nuclear industry in recent years. This is due to current advances in computing power, data science, and artificial intelligence at large. In the event of an accident at a nuclear power plant (NPP), machine learning can provide crucial support for maintaining safety and minimizing human error. ML meta-models can be harnessed to expedite the decision-making process during an NPP accident condition.

Artificial intelligence encompasses machine learning algorithms, which aim at imitating the human learning process by way of adaptation to relevant data features.[1] Several ML implementation tools can be used, for instance, Support Vector Machine (SVM), Decision Tree (DT), Multilayer Perceptron (MLP) random forest, K- Nearest Neighbor (KNN), Logistic Regression (LR), or Neural Network (NN), These approaches learn to reveal the hidden features within a dataset through continuous mathematical operations to characterize and generalize what has been learned to predict similar but unseen dataset. To accomplish this, independent variables (inputs) are linked to dependent variables (outputs) via weights and biases which are continuously updated until the model adapts to the dataset features.

In recent years, a growing number of nuclear accident scenarios are being predicted using machine learning. For instance, a comparative study of a neural-based LSTM with deep neural networks (DNN) for predicting a loss of coolant accidents was achieved with reasonable accuracy.[2] Similarly[3], performed a real-time prediction of the NPP response following operator actions; while[4], evaluated three machine learning models, Support Vector Machines (SVM), Decision Trees (DT), and Multilayer Perceptron (MLP) for nuclear reactor accident prediction using a dataset that reflects the sensor status taking into account parameter uncertainty caused by sensor failure during an accident. In a recent study[5], used DNN to identify the window of time during which the FLEX strategy will succeed for NPP under extended station blacked out.

To train, test, and validate the ML meta-model a substantial amount of time series database is required. The database can be acquired through the development of an uncertainty quantification framework to perform the best estimate plus uncertainty quantification (BEPU) analysis by coupling DAKOTA statistical software with the best estimate thermal hydraulics system code, MARS-KS.

The best estimate plus uncertainty quantification (BEPU) methodology has been applied to analyze reactivity-initiated accidents (RIA) by [6],[7],[8], loss coolant accident (LOCA) by [2],[9],[10],[11], and more recently, station blackout (SBO) by [12], [5].

The BEPU methodology starts with the Phenomena Identification and Ranking Table (PIRT) established by [13],[14] for Reactivity Initiated Accident (RIA). Next, key uncertain parameters are derived and propagated using the non-parametric Monte-Carlo approach using Wilks' fifth-order statistics and the Latin Hypercube Sampling

(LHS) technique to generate a statistically significant database of the thermal-hydraulic NPP response.

Systems Engineering involves advanced techniques that help simplify the handling and management of complex engineering systems to systematically and efficiently achieve a set of goals within the defined time frame. The Systems Engineering approach was applied to develop a framework for a multi-physics load following simulation of the Korean APR1400 nuclear power plant by [15]. Furthermore, [16] explored the Systems Engineering approach to predict the success window of FLEX strategy under extended Station Black Out (SBO) using artificial intelligence. Likewise[17],[18], conceptualized a simplified Field Programmable Gate Arrays (FPGA) based on the Core Power Calculation System (CPCS) and Functional Complexity Reduction Concept (FCRC) of the Man-Machine Interface System (MMIS) for Innovative SMRs using the Systems Engineering approach.

1.1 Accident Scenario

The chosen accident scenario is a Control Element Assembly (CEA) withdrawal concurrent with Loss of Offsite Power (LOOP) and the power plant of choice is APR1400. The presumed transient of uncontrolled CEAs withdrawal may occur as a result of a single failure in the Control Element Drive Mechanism Control system (CEDMCS), reactor regulating system (RRS), or with regards to operator error concurrent with the Loss of Offsite Power (LOOP)[19]. Operating at nominal power condition, the reactor undergoes uncontrolled withdrawal at the speed of 76.2 cm per minute with an equivalent reactivity insertion rate of 0.315 × 10−4 Δρ/s which in turn induces an increase in the core power and heat flux with a corresponding increase in the Reactor Coolant System (RCS) temperature and pressure. It is important to also note that the reactor at the above set conditions will experience asymmetrical distribution of core power, leading to intense thermal stress in the region of CEA withdrawal and consequently, the specified acceptable fuel design limits (SAFDL) on departure from nucleate boiling ratio (DNBR) and fuel centerline melt temperatures might be approached which will eventually lead to the reactor protection system (RPS) signaling on Variable Overpower (VOP), Low DNBR, High Local Power Density (HLPD) and or High Pressurizer Pressure (HPP) and hence reactor trip.

1.2 Objective

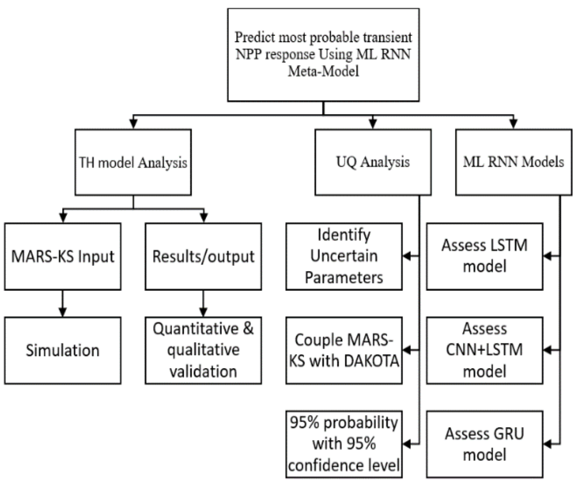
In this research, we intend to develop a machine learning meta-model based on time series data to predict the transient response of an NPP undergoing a reactivity-initiated accident. This research demonstrates that machine learning techniques can be effective in supporting nuclear power plant operators and will be extended in the future to provide prudent solutions/decisions in case of more severe accident conditions; most importantly, at an affordable cost. The research objectives can be summarized as follows:

1) Use Best Estimate Plus Uncertainty Quantification BEPU analysis to quantify the uncertainty for CEA withdrawal at power to generate a sufficient database

for the machine learning meta-model training, testing, and validation,

2) Develop a robust RNN time series machine learning algorithm to predict real-time key safety features of a nuclear power plant (NPP) under a CEA withdrawal accident scenario,
3) Assess the applicability of the deep learning technique to forecast time-series events for a nuclear power plant under accident condition,
4) Estimate the potential of deep learning techniques to reduce the high computational cost of the physics-based model, and
5) evaluate the potential of deep learning techniques to expedite the decision-making process for plant operators in the event of nuclear power plant accident condition.

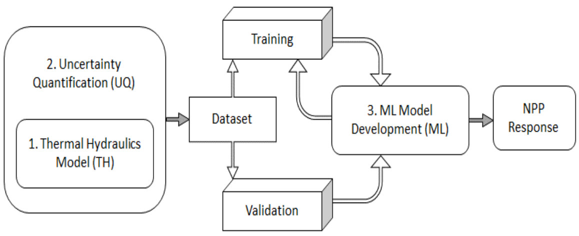
In line with the research objectives above, the overall research framework was achieved using the Systems Engineering (SE) approach to plan and manage from conceptualization, implementation, and validation of the ML meta-model to predict the Nuclear Power Plant (NPP) transient response under reactivity-initiated accident conditions. Figure 1 depicts the whole research objective hierarchy.



[Figure 1] Objective Hierarchy

2. Methodology

This section discusses the methodology applied to achieve the set objectives that were outlined in section 1 with the ultimate goal of predicting the NNP response for the accident scenario under consideration. The work consists of developing three main building blocks: thermal-hydraulic model, uncertainty quantification framework, and machine learning model. Figure 2 illustrates the overall research methodology.



[Figure 2] Research Framework

2.1 Thermal Hydraulic Model

For the thermal-hydraulic simulation, the best estimate thermal-hydraulic system code, MARS-KS version 1.4, was used. The input deck was developed to reflect the key systems and components of APR1400 model and the nodalization was verified for steady-state behavior. Figure 3 depicts the APR1400 nodalization used for the current simulation. TH model was modified to reflect the initial and boundary conditions as well as the kinetics parameters relevant to the CEA withdrawal accident scenario at full power. For the chosen accident scenario, the transient was modeled using appropriate control logic and trips to reflect exactly the assumption from the design and control document (DCD) Chapter 15 for cross-validation of the transient response. Table 1 depicts the conservative assumptions used for cross-validation against the DCD.

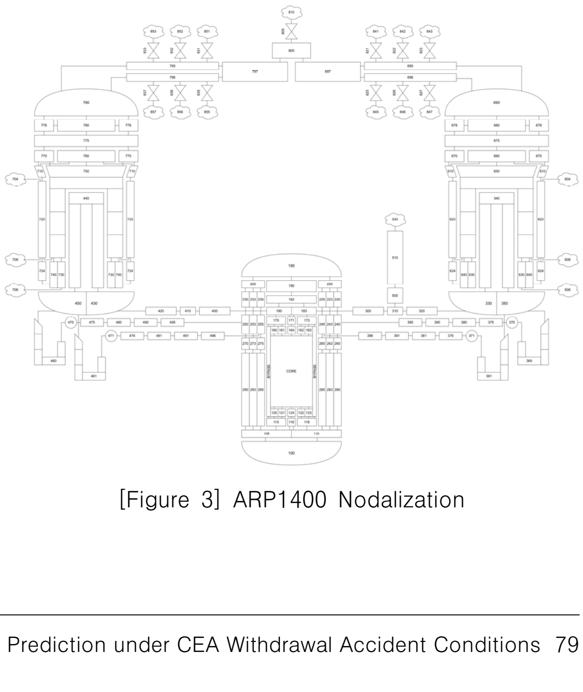
2.1.1 APR1400 Nodalization

The thermal-hydraulic model development necessitates building a system nodalization for APR1400 key systems and components relevant to the selected accident scenario as shown in Figure 3. The primary side consists of the reactor pressure vessel (RPV), a pressurizer (PRZ), two loops with four cold legs (CLs), and two hot legs (HLs) connected to the steam generators. The secondary side includes a detailed representation of the two steam generators (SGs), four main steam lines together with associated valves (MSSV, MSIV, and ADV).

From either side of the loop, the primary coolant flows from the RPV through the SG u-tube section via a single HL, where the heat is transferred to the secondary feed water, and then back to the RPV via the two CLs. Each cold leg hosts a single reactor coolant pump (RCP) that forces the flow of coolant in the primary circuit. On one of the hot legs, the PRZ is connected to compensate for pressure drop or build-up in the primary system. Four Pilot Operated Safety Relief Valves (POSRV) are connected to the pressurizer to protect the primary side against over-pressurization. The reactor core is represented using an average channel by lumping 240 fuel assemblies and a hot channel representing the hottest fuel assembly. Both the average channel and the hot channel is discretized using 20 vertical nodes. The turbine is modeled as a boundary condition. The safety injection system (SIS) is modeled to represent the emergency core cooling system (ECCS) of the APR1400.

<Table 1> Conservative Assumptions for CEA Withdrawal

Parameter	DCD
Core power, MWt	4062.66
Core inlet coolant temperature, °C	287.8
Core mass flow rate, 106 kg/hr	69.64
Pressurizer pressure, kg/cm^2	163.5
Integrated radial peaking factor	1.49
Initial core minimum DNBR	1.72
Steam generator pressure, kg/cm^2	68.26
Moderator temperature coefficient	Most positive
Fuel temperature coefficient	Least negative
CEA worth on trip, %Δρ	−8.0
Reactivity addition rate, 10−4Δρ/sec	0.315
CEA withdrawal speed, cm/min	76.2

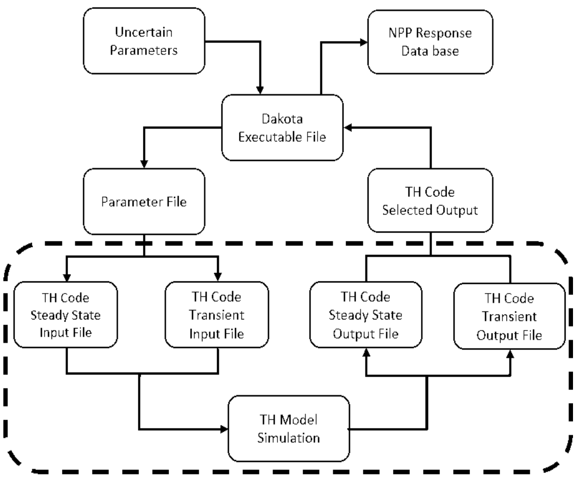


[Figure 3] ARP1400 Nodalization

2.2 UQ Framework

An uncertainty quantification (UQ) framework was developed by coupling the best estimate system code, MARS-KS, and the statistical tool, DAKOTA, via a python interface. DAKOTA is an open-source statistical tool developed by the Sandia National Laboratory and can be used for optimization, sensitivity analysis, and uncertainty quantification.[20] The coupling of MARS-KS and DAKOTA permits the propagation of the uncertain parameters using the non-parametric Monte Carlo random propagation technique based on fifth-order Wilks' statistics. It is important to note that in DAKOTA the Latin Hypercube Sampling (LHS) was used given its computational efficiency compared to the simple random sampling (SRS) technique.

The expected system responses are obtained as a database for the ML model. These selected responses consist of DNBR, RCS pressure, core flow, core power, peak cladding temperature, core inlet temperature, heat transfer coefficient, and heat flux. Figure 4 below depicts the UQ framework.



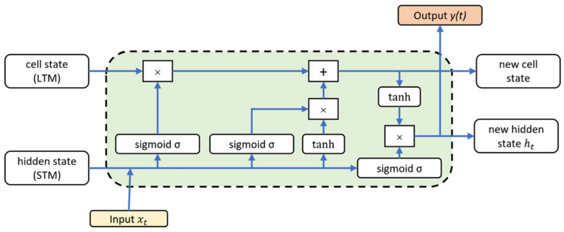
[Figure 4] UQ Framework

2.3 ML RNN Meta-Models

Recurrent Neural Networks (RNNs) are capable of predicting sequence data. This is done using previous information from the independent variables (inputs) to find characteristic trends and hence predict future dependent variables (output). As previously mentioned, three different RNN models were developed and their performances were evaluated: the RNN includes Long Short Term Memory (LSTM), Gated Recurrent Unit (GRU), and a series combination of Convolutional Neural Network (CNN) and LSTM.

2.3.1 Long Short-Term Memory (LSTM)

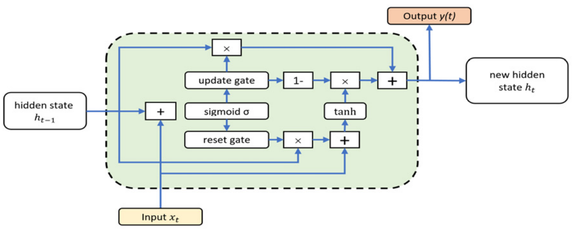
LSTM neural networks are a specialized type of RNN, designed to adapt to long-term dependency using its ability to memorize trends, as well as to discard any unrelated information that does not contribute to the prediction or forecasting (Hochreiter and Schmidhuber, 1997). As shown in Figure 5, LSTM comprises three different types of gates in each of its internal unit cells that help navigate the information dynamics within a given cell unit. These gates include: the forget gate (f_t), the input gate (i_t) and output gate (o_t). Each gate is controlled by a specific activation function (usually a sigmoid and tanh function) that normalizes the outputs in a range from 0 to 1 and −1 to 1, respectively. The gates interact with the cell state, which acts much like an information carrier that accepts relevant information and discards irrelevant information.



[Figure 5] LSTM Cell Unit

2.3.2 Gatted Recurrent Unit (GRU)

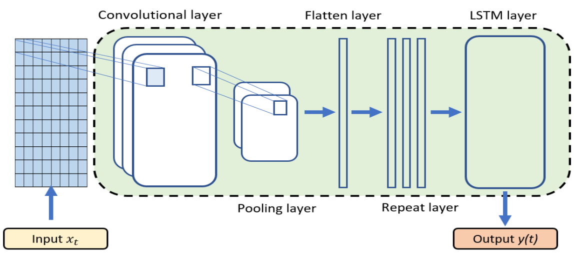
Developed in 2014 by [22], GRU has a similar structure to LSTM, except for a difference in the operation and associated gates of its unit cells. Each GRU unit cell consists of an update gate and a reset gate activated by the sigmoid function as depicted in Figure 6.



[Figure 6] GRU Cell Unit

2.3.2 Convolutional Neural Network (CNN)+LSTM

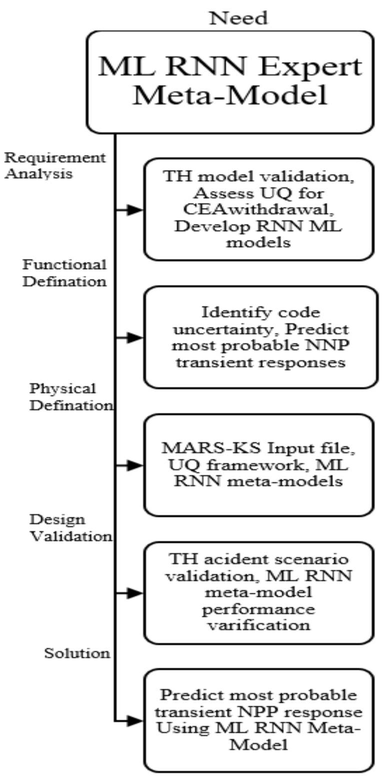
One of the most commonly used neural networks for deep learning is the Convolutional Neural Network (CNN). CNN works with shared-weight convolution kernels that slide along input features and provide equivariant representations known as feature maps. CNN has been applied in image recognition[23]; as well as facial recognition[24], with accuracy as high as 97%.[25] Time series forecasting has generally been performed using recurrent neural networks, but a recent study suggests that when combined with convolutional neural networks the performance is enhanced.[26] A number of researchers have therefore adopted the combination of CNN and LSTM for better time-series predictions; for instance, Ke and Chan, (2021) developed a multilayer CARU framework to obtain probability distribution for paragraph-based sentiment analysis; Li, (2022) predicted wind speed of an unmanned sailboat based on a hybrid (CNN + LSTM) neural network; similarly, Kortli et al., (2022) developed a deep embedded hybrid (CNN + LSTM) network for lane detection on NVIDIA. This research explores as an alternative a series combination of CNN and LSTM in comparison to GRU and LSTM. A schematic of a hybrid network combining CNN with LSTM in series is depicted in Figure 6.



[Figure 7] CNN+LSTM

3. System Engineering Method

Systems Engineering (SE) as an interdisciplinary approach will help allows for the actualization of effective systems planning, development, and implementation of this research objectives as outlined in section 1. SE encompasses the entirety of the system lifecycle from realization, utilization, evolution, and disposal with specified procedures to achieve all lifecycle processes of the entire system ISO/IEC/IEEE/15288. A SE approach is used in this study to plan and manage the entire system lifecycle's stages to develop an ML RNN meta-model algorithm for the prediction of key transient NPP responses, namely: DNBR, RCS pressure, core flow, and core power for NPP under reactivity-initiated accident scenarios specifically CEA withdrawal at power concurrent with the loss of offsite power.



[Figure 8] System Engineering Method

To ensure that all requirements are met within reasonable limits, verification and validation tests are conducted at every developmental phase. This is aided by the simplicity of the V-Model system engineering lifecycle type, which is a representation of the systems development lifecycle that summarises the main steps to be taken with corresponding deliverables within the validation framework.

It is important to note, that Kossiakoff's four fundamental concepts which include majorly, (i) defining the system requirement, (ii) performing functional analysis, (iii) physical definition, and (iv) performance validation; are assumed in this research. Figure 8 illustrates the system engineering method employed in this research.

3.1 Work Breakdown Structure (WBS)

Having defined the Systems Engineering method and the objective hierarchy as shown in Figures 1 and 7, respectively, the work breakdown structure for this research entails:

Qualitative cross-validation of APR1400 TH model simulation for CEA withdrawal at full power against design and control document (DCD) taking into account the conservative assumptions

Set TH model input to nominal operating initial and boundary condition for UQ taking into account the nominal assumptions

Identify uncertain parameters in relation to reactivity-initiated accident conditions from the literature

Couple DAKOTA with MARS-KS to perform UQ. This generates enough NPP transient responses as a database for the RNN ML model training, testing, and validation

Selecting the most probable NPP transient responses using Wilk's fifth order method to guarantee 95% probability and 95% confidence level

Develop three RNN ML meta-models and

compare their performance based on specified metrics of evaluation (MSE, MAE, RMSE, R2)

Predict the most probable transient NPP responses using the developed ML models

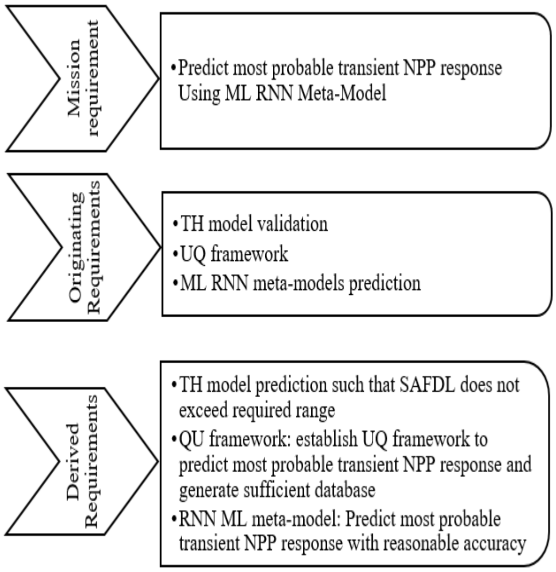
Compare prediction against the true value of the NPP transient responses

3.2 Requirement Analysis

Requirement analysis entails defining the project objectives and or needs with regards to client use, conditions, and distinguished system attributes to decide prerequisites for framework capabilities which is to predict NPP transient response under the CEA withdrawal accident condition using three different ML RNN meta−models. Requirements Analysis serves to: distil clients objectives and requirements, set expected objectives and improve them into requirement, profer measures to checkmate effectiveness based on defined functional and performance requirement.

Figure 9 depicts the decomposed requirements analysis by categorizing it into mission, originating, and derived requirements. Starting from the mission requirement, the Systems Engineering method deployed should be able to help develop a systematic approach to build and optimize three different ML RNN meta−models to predict NPP transient response for a CEA withdrawal accident scenario with acceptable accuracy monitored within the validation framework.

Secondly, the originating requirement allows for the above mission statement to be accomplished by utilizing the appropriate TH model coupled with the DAKOTA tool to develop a UQ framework using Wilk's fifth



[Figure 9] Requirement Analysis

order to guarantee the 95% probability with 95% confidence level, a criterion commonly required by the [27] for BEPU analyses. Part of the originating requirement is the development of the ML RNN meta−models to predict the NPP transient response using the database generated from the UQ framework taking into account specified testing and validation methods to achieve the desired mission statement.

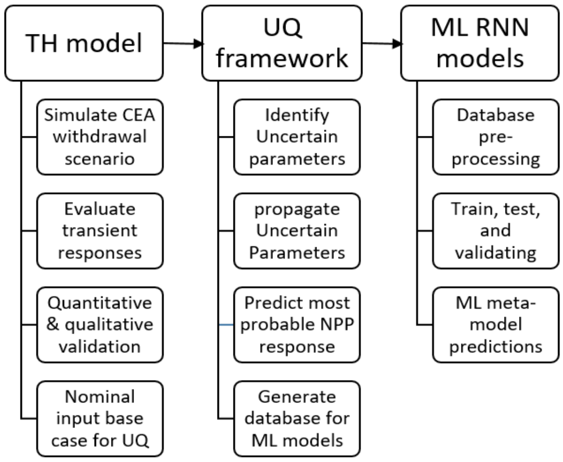
Lastly, the derived requirements entail the prediction of the NPP transient responses by the ML RNN meta−models such that the SAFDL does not exceed the accepted range. This is done by training, testing, and validating the ML RNN meta−models for distinct generalization to the database at hand.

3.3 System Architecture

This is the early stage of the system design process. In this step, major components of the system are identified, along with their interconnections. An architectural description of the modules is produced by this design process. MARS-KS and DAKOTA are codes used to provide a one-way communication interface through the help of Python programming language to loosely coupled and provide for the UQ framework. MARS-KS is a best-estimate Multi-dimensional Analysis of Reactor Safety code developed by the Korean Atomic Energy Research Institute (KAERI) and used for the simulation of thermal-hydraulic (TH) NPP transient response while Dakota is an open-source statistical tool developed by the Sandia National Laboratory that can be used for optimization, sensitivity analysis, and uncertainty quantification Adams et al. (2020). This process generates a big enough dataset to train the ML RNN meta-models. Figure 10 depicts the system's architecture.

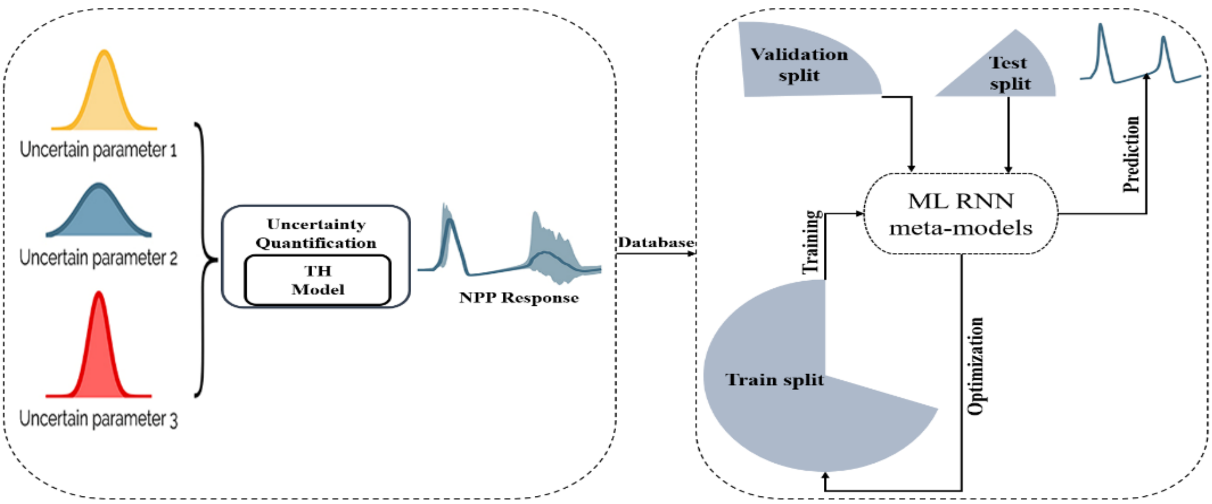
3.4 Functional Architecture

The functional architecture portrays the fundamental capabilities, first for the TH model to predict the specified Figure of Merit (FoM) for the CEA withdrawal accident condition. The TH model function to maintain the capability by making sure the SAFDL is kept within the specified acceptable range and cross-validated against the DCD. This is done by modeling appropriate initial and boundary conditions taking into account physical components, structures, and model parameters.



[Figure 11] Functional Architecture

The UQ framework is developed with the help of the DAKOTA tool, which functions to ascertain the most probable system response



[Figure 10] System Architecture

and generate a sufficient database for the ML model's training, testing, and validation. To propagate the uncertainty nominal input for APR1400 CEA withdrawal scenario at power is considered together with defined uncertain parameters in the literature.

The ML RNN meta-models functional architecture help predict with excellent certainty the most probable NPP transient responses. Here, three different ML RNN models are built and optimized with the help of the Talos tool. Talos is an open-source tool that functions to optimize and build ML models as considered in this research.

The functional architecture of the ML RNN meta-models basically explains the training, testing, validating, and deployment of the ML RNN meta-models. 181 samples were generated using Wilk's fifth-order statistics. 161 samples were used as a training dataset with a 20% cut for validation and 15% for testing the models. Nine features are passed to the model for multivariate time series prediction: core power, core flow, heat flux, critical heat flux, peak cladding temperature, RCS pressure, inlet temperature, DNBR, and time. The four key NPP most probable transient responses (DNBR, RCS pressure, core flow, and core power) are selected to be forecasted by the ML RNN meta-models.

Next, the functional architecture of the ML RNN meta-models entails processing the dataset. The preprocessing is done by normalization or scaling to minimize the bias as a result of too-big or too-small data range by specifying a finite range of 0 to 1 using the MinMax scaler function. Similarly, the ML RNN meta-models which consist of specified optimum hidden layers, neurons per layer, activation function, learning rates, and other specified hyperparameters are trained, optimized validated, and then deployed to predict the most probable transient responses. Figure 10 depicts the functional architecture TH model, UQ framework, and ML RNN meta-model.

3.5 Physical Architecture

The architectural representation of the fundamental system structure and components which are used for the CEA withdrawal accident scenario are considered. Physical architecture is usually understood in relation to functional architecture. Table 1 depicts the major systems and components of APR1400 model for a CEA withdrawal accident condition. The detailed representation of systems and components is described in the ARP1400 model nodalization section 2.1.1 and Figure 3.

For the UQ framework, the physical structure consists of the DAKOTA interface coupled with the MARS-KS system code. Figure 4 depicts a simplified representation of the physical coupling of the UQ framework.

The ML RNN physical architecture representing the three different models entails the simplified unit cells as depicted in Figures 5, 6, and 7 for LSTM GRU and CNN+LSTM respectively. The LSTM unit cell is fundamentally defined by the input forget and while the GRU consists of majorly the update and resets gate. Distinct to CNN+LSTM is the convolutional neural network majorly characterized by the convolutional and pooling layers.

All three ML RNN meta-model architectures are obtained after many iterations to obtain a computationally efficient structure that can learn the relevant data features and generalize them on an unseen dataset. This can be achieved with the help of Talos tool which helps optimize the hyperparameters such as the number of neurons, number of hidden layers, activation functions, etc.

<Table 1> APR1400 Systems and Components

Primary Systems
Reactor Pressure Vessel (RPV)
Four Reactor coolant Pumps (RCPs)
Pressurizer
Two(2) Hot Legs(HL) and Four Cool Legs (CL)
Pilot Operated Safety Relieve Valves (POSRV)
Four Safety Injection Tanks (SITs)
Secondary Systems
Two (2) Steam Generators (SGs)
Four Main Steam lines
Main Steam Safety Valves (MSSVs)
Main Steam Isolation Valves (MSIVs)
Turbine Isolation Valve (TIV)
Atmospheric Dump Valves (ADVs)

3.6 Model Development Phase

The model development phase entails the development of the three interrelated modules required to predict the NPP transient responses for the CEA withdrawal accident condition. As specified in the system lifecycle type (V-Model of Figure 12), three modules development will consist of the development of the TH model, UQ framework, and ML RNN meta-models.

The TH model shall accurately simulate NPP transient responses for CEA withdrawal accident conditions with qualitative and quantitive in comparison to the NPP response reported in DCD. While the UQ framework shall generate a sufficient database for the ML RNN meta-models as well as predict the most probable transient NPPP responses. Similarly, the ML RNN meta-model shall be developed to predict accurately the most probable transient response as well as generalize time series features over all time steps for a CEA withdrawal accident condition.

3.7 System Implementation Phase

During the system implementation phase, the system requirements developed in the early life cycle phase are taken into account and used to construct system elements in line with architectural design and system analysis results. The three modules outlined in the module development section shall be implemented sequentially as depicted in the system architecture outlined in Figure 10. Having cross-validated the TH model against the DCD, the UQ framework is developed by coupling the TH system code MARS-KS, and the DAKOTA statistical tool via a python interface. The identified uncertain parameters are propagated by coupling the DAKOTA software with the MARS-KS system code to meet the system requirements. The UQ framework then generates a statistically representative database of 181 samples as required by Wilks' fifth order. This database is then preprocessed through normalization after which it is split into training, testing and validation dataset to be used for the development, optimization, and validation of the ML RNN meta-model with the help of the

Talos tool.

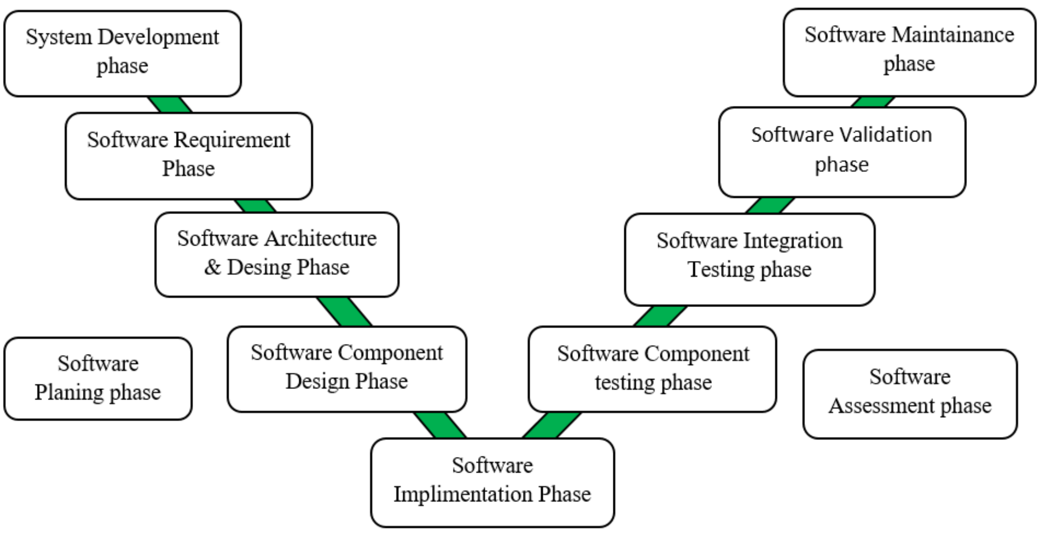
The system element for the ML RNN meta–model will consist of the Talos tool, python programming language, a google collaboratory interface, modules (Keras, TensorFlow, SkitLearn), and a dataset from the UQ framework. The developed ML models are trained and validated on the training and validation dataset. The ML RNN meta–model can now predict with reasonable certainty the NPP transient responses under the CEA withdrawal condition given the appropriate combination of multivariate input to meet the stakeholder–identified requirements. A summary of the system implementation phase is depicted in Figure 12 in terms of the V–Model developed over the system lifecycle.

3.8 System Verification and Validation

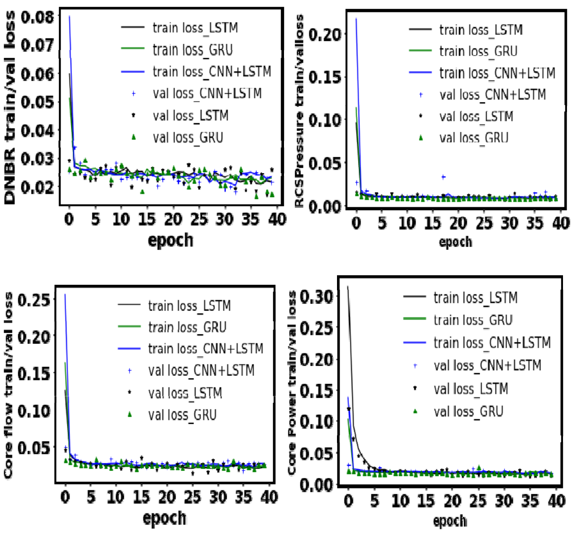
System verification and validation confirm the system's fidelity to fulfill the specified design requirements by monitoring the correctness of each system element. Here, the scope includes a comparison of system elements outlined in the implementation phase against architectural design and requirements.

Several methods can be used to verify the developed TH model, UQ framework, and ML RNN meta–models, these methods include testing, analysis, sampling, demonstration, inspection, etc. This research focuses on the testing method of verification. The verification process shall take place at both the unit and integrated model levels. Unit tests shall verify that all models perform well independently and provide the expected output. The integrating testing takes into account the overall performance in connection to all three models to reflect the entire requirement statement.

The system validation test is carried out in relation to the mission statement which is to predict NPP transient response using ML RNN meta–models. Firstly, common phenomena associated with ML model training consist majorly of over–fitting, under–fitting, and good–fitting. This usually makes the ML RNN meta–model to overestimate, under–estimate, and or have a good fitting respectively. During



[Figure 12] V–Model (Base on IEC62279 assumed from [28]) the training process, it is important to evaluate the model performance by monitoring the ML model's loss function for both training and validation datasets over all the time steps as shown in figure 13. This is to give an idea about the ML model's ability to learn data features over all time steps without over-fitting or under-fitting. The green, blue and black makers compare the validation loss function corresponding to the training loss function. The closer the training and validation loss the better the ML RNN meta-model to predict the NPP transient response on an entirely new dataset usually referred to as a test dataset.



[Figure 13] Traning and Validation Losses

Having monitored the learning curve averaged over all time steps for the training data set of the ML model as depicted in Figure 12, it is paramount to test the ML models to determine their performance to an entire set of new datasets (test data). For an ideal model, the prediction and the true value of the dependent variable should be equal. In reality, this is never the case, and usually, the disparity is quantified to assess the model's fidelity. Five metrics of evaluation were used in this research to quantify the disparity, namely: R^2, MSE, RMSE, MAE, and accuracy computed using the following equations:

$$R^2 = 1 - \frac{\sum_{i=0}^{n}(y_{true} - y_{pred})^2}{\sum_{i=0}^{n}(y_{true})^2} -------- (1)$$

$$MSE = \frac{1}{N}\sum_{i=0}^{N}(y_{true} - y_{pred})^2 --------(2)$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=0}^{N}(y_{true} - y_{pred})^2} ------(3)$$

$$MAE = \frac{1}{N}\sum_{i=0}^{N}|y_{true} - y_{pred}| --------(4)$$

$$Accuracy = \frac{1}{N}\sum_{i=0}^{N} 1 - \frac{|y_{true} - y_{pred}|}{y_{true}} ----(5)$$

where y_{pred} and y_{true} represent the predicted and true values of the dependent variables, respectively; N represents the number of samples in a dataset.

3.8.1 System Unit Testing

The TH model is quantitatively and qualitatively tested against the TH simulation results reported in the DCD. It is expected to qualitatively conform strongly with the DCD's reported trend of NPP response for a CEA withdrawal accident condition, however qualitatively there might have some discrepancies due to differences in simulation tools used.

The UQ quantification framework is expected to predict the most probable transient response as well as generate a sufficient database for the ML RR meta-models

The three ML RNN meta-model performance is individually evaluated. This is to test the ML model's fidelity to perform on an entirely new dataset. the ML model's predictions of the NPP transient response are compared with the actual most probable transient. All ML models shall be tested and evaluated separately using evaluation metrics. The output of the summary of the ML model's evaluation metrics averaged over all time steps based on the test data set is listed in Table 2 below.

<Table 2> ML Evaluation Metrics

	ML Models	MSE	RMSE	MAE	R^2	Acc. (%)
DNBR	LSTM	0.0269	0.1640	0.0505	0.9966	92.524
	CNN+ LSTM	0.0268	0.1639	0.0841	0.9731	91.537
	GRU	0.0399	0.1997	0.0836	0.9950	91.210
Pressure	LSTM	0.0092	0.0444	0.0960	0.9907	87.736
	CNN+ LSTM	0.0080	0.0896	0.0233	0.9919	94.737
	GRU	0.0080	0.0899	0.0304	0.9919	94.512
Core Flow	LSTM	0.0159	0.1262	0.0513	0.9840	91.832
	CNN+ LSTM	0.0234	0.1532	0.0486	0.9764	91.427
	GRU	0.0610	0.2470	0.0918	0.9389	86.486
Core Power	LSTM	0.0187	0.1367	0.0682	0.9812	92.321
	CNN+ LSTM	0.0144	0.1203	0.0272	0.9855	96.884
	GRU	0.0161	0.1272	0.0363	0.9838	95.833

3.8.2 System Integration Testing

From the system lifecycle in Figure 12, the integration testing will consist of the nominal TH model input coupled with the DAKOTA tool to propagate the uncertainty. This is to predict the most probable transient system responses and as well serves as a tool to generate a sufficient database for the ML RNN meta-models. The ML RNN meta-models shall predict with reasonable accuracy as depicted in Table 2 in predicting the series event for the CEA withdrawal accident scenario. As is with the unit testing, to verify the ML RNN meta-model performance, the prediction shall be monitored and evaluated using the specified metrics as indicated in equations 1, 2, 3, 4, 5, and 6 averaged over all time steps. Table 2 indicates how well the ML RNN models perform.

3.8.3 System Testing

Here full system implementation testing is conducted. The system architecture defines the entire integral system with the sole aim of predicting the NPP transient responses under reactivity-initiated accident conditions. In the entire system testing and validation, the ML RNN meta-models are monitored to verify for generalization of the ML models on the unknown dataset. This is to avoid common challenges faced by ML models of gradient explosion and or overfitting and under-fitting.

3.8.4 System Acceptance Testing

This phase entails confirmation of the mission requirement as depicted in Figure 9, which fundamentally emphasizes the ML RNN meta-model's ability to predict the NPP transient responses for the CEA withdrawal scenario with high accuracy and generalization as depicted by the accuracy and R2 in Table 2. Here, the ML models were considered acceptable if the predictions are matching the actual outputs in the testing subset of the dataset with an average accuracy above 90% as shown in Table 2.

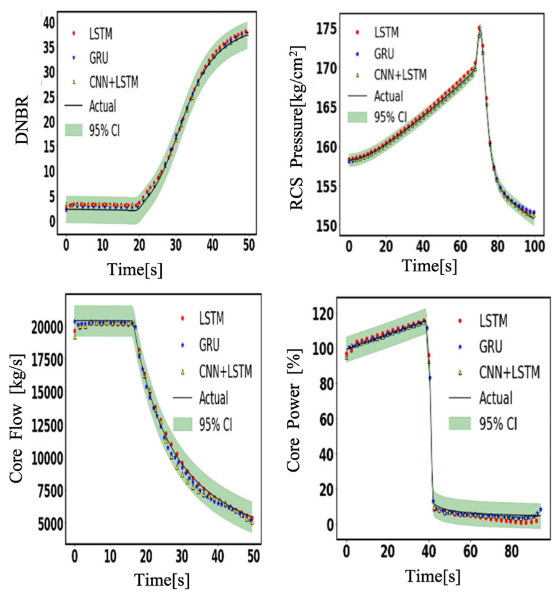
Verification and validation of the various developmental stages of the ML RNN meta-models with the aid of the V-Model as shown in Figure 12 was conducted to ensure compliance with all stated system requirements. The ML RNN meta-model is then deployed to predict the NPP transient responses using the generated database from the UQ framework.

3.9 Results and Discussion

Once the ML models are trained and validated, they are then tested to predict the most probable NPP transient response using the test dataset which is entirely new to the RNN ML models. This is to ensure compliance with all stated system requirements. The testing dataset consists of the input dimension of (4030,10,9) for all the ML models which include the most probable NPP transient response as predicted by the UQ method. Figure 14 depicts the test prediction for all three RNN ML models. Red, blue, and yellow markers represent the predicted curves for LSTM, GRU, and a combination of CNN and LSTM, respectively.

The ML model's prediction depicts with reasonable accuracy the general trend of the key parameters of the NPP response as shown in Figure 14 although the LSTM model underestimates the initial core power and flow rate but predicts with reasonable accuracy the later behavior and especially the peak values for both the core power and flow rate which from the safety perspective qualifies the model since it can capture correctly the limiting NPP transient response. While GRU better predicts the NPP initial transient response but tends to underpredict the peak values of core power and flow rate as well as later transient response in the core. The overall accuracy of prediction averaged over all time steps the GRU and LSTM have shown comparable performance with an accuracy of 90% and 91%, respectively. The model combining CNN and LSTM outperformed both the GRU and LSTM with an accuracy of 94% as shown in Table 2. The computational efficiency of the hybrid model (CNN+LSTM) may be attributed to the handling of compacted data which leads to a faster processing time of 1.75s per iteration step as opposed to 6.75s/step and 2s/step for the GRU and LSTM, respectively.

It is important to also note that a 95% confidence interval for the ML model's



[Figure 14] ML RNN Meta-Model Predictions prediction was estimated to depict the fidelity of the ML model's predictions with 5% uncertainty shown as the light green band around the most probable predicted NPP transient responses. The outcomes show the precision of the ML model's predictions of the four most probable NPP transient responses. The majority of the forecasts fall inside the assessed 95% certainty band of the actual value.

3.10 Conclusion

In this work, the SE approach has been adopted to plan and manage this research as defined by the mission requirement complemented by the objective hierarchy. At each stage of the project development, the V-model ensured that the requirements were met to achieve the sole aim of predicting the NPP transient responses for CEA withdrawal accident condition using three different machine learning (ML) algorithms based on recurrent neural networks (RNNs). For the ML RNN meta-models, the UQ framework which consists of coupling the TH model with DAKOTA software was used to predict the most probable transient NPP response as well as generate the required database as required by Wilk's fifth order.

Although the developmental process of the ML RNN meta-models using the Systems Engineering approach is systematic, the process might take longer due to some bottleneck, but once developed, the RNN ML meta-models can then be used to predict the NPP transient response much faster than the conventional physic base model. This may be highly beneficial in expediting prudent solutions/decisions under more severe accident condition.

Acknowledgments

This research was supported by the 2022 Research Fund of the KEPCO International Nuclear Graduate School (KINGS), Republic of Korea.

References

1. B. Mehlig, Machine learning with neural networks. 2021. doi: 10.1017/9781108860604.
2. M. I. Radaideh, C. Pigg, T. Kozlowski, Y. Deng, and A. Qu, "Neural-based time series forecasting of loss of coolant accidents in nuclear power plants," Expert Syst. Appl., vol. 160, p. 113699, Dec. 2020, doi: 10.1016/j.eswa.2020.113699.
3. J. Bae, G. Kim, and S. J. Lee, "Real-time prediction of nuclear power plant parameter trends following operator actions," Expert Syst. Appl., vol. 186, p. 115848, Dec. 2021, doi: 10.1016/j.eswa.2021.115848.
4. S. Racheal, Y. Liu, and A. Ayodeji, "Evaluation of optimized machine learning models for nuclear reactor accident prediction," Prog. Nucl. Energy, vol. 149, p. 104263, Jul. 2022, doi: 10.1016/j.pnucene.2022.104263.
5. S. O. Alketbi and A. Diab, "Identification of Flex Strategy Success Window for Extended SBO using BEPU and AI," p. 4, 2020.
6. A. Dokhane, A. Vasiliev, M. Hursin, D. Rochman, and H. Ferroukhi, "A critical study

on best methodology to perform UQ for RIA transients and application to SPERT-III experiments," Nucl. Eng. Technol., vol. 54, no. 5, pp. 1804-1812, May 2022, doi: 10.1016/j.net.2021.10.042.
7. J. P. Gorton and N. R. Brown, "Defining the performance envelope of reactivity-initiated accidents in a high-temperature gas-cooled reactor," Nucl. Eng. Des., vol. 370, p. 110865, Dec. 2020, doi: 10.1016/j.nucengdes.2020.110865.
8. O. Marchand, J. Zhang, and M. Cherubini, "Uncertainty and sensitivity analysis in reactivity-initiated accident fuel modeling: synthesis of organisation for economic co-operation and development (OECD)/ nuclear energy agency (NEA) benchmark on reactivity-initiated accident codes phase-II," Nucl. Eng. Technol., vol. 50, no. 2, pp. 280-291, Mar. 2018, doi: 10.1016/ j.net.2017.12.007.
9. C. Queral, J. Montero-Mayorga, J. Gonzalez-Cadelo, and G. Jimenez, "AP1000® Large-Break LOCA BEPU analysis with TRACE code," Ann. Nucl. Energy, vol. 85, pp. 576-589, Nov. 2015, doi: 10.1016/ j.anucene.2015.06.011.
10. P. Mazgaj et al., "Demonstration of the E-BEPU Methodology for SL-LOCA in a Gen-III PWR Reactor," Reliab. Eng. Syst. Saf., p. 108707, Jul. 2022, doi: 10.1016/ j.ress.2022.108707.
11. W. Chen, Q. Xiong, D. Wu, S. Ding, L. Qian, and Q. Wu, "Uncertainty analysis of HPR-1000 LOCA with probabilistic and deterministic methods," Prog. Nucl. Energy, vol. 146, p. 104174, Apr. 2022, doi: 10.1016/j.pnucene.2022.104174.
12. A. Ghione, B. Noel, P. Vinai, and C. Demazi?re, "Uncertainty and sensitivity analysis for the simulation of a station blackout scenario in the Jules Horowitz Reactor," Ann. Nucl. Energy, vol. 104, pp. 28-41, Jun. 2017, doi: 10.1016/j.anucene. 2017.02.008.
13. O. Marchand, J. Zhang, and M. Cherubini, "Uncertainty and sensitivity analysis in reactivity-initiated accident fuel modeling: synthesis of organisation for economic co-operation and development (OECD)/ nuclear energy agency (NEA) benchmark on reactivity-initiated accident codes phase-II," Nucl. Eng. Technol., vol. 50, no. 2, pp. 280-291, Mar. 2018, doi: 10.1016/j.net.2017.12.007.
14. J. Zhang, J. Segurado, and C. Schneidesch, "Towards an Industrial Application of Statistical Uncertainty Analysis Methods to Multi-physical Modelling and Safety Analyses," p. 19, 2011.
15. A. E. R. Mahmoud and A. Diab, "A Systems Engineering Approach to Multi-Physics Load Follow Simulation of the Korean APR1400 Nuclear Power Plant," J. Korean Soc. Syst. Eng., vol. 16, no. 2, pp. 1-15, Dec. 2020, doi: 10.14248/JKOSSE.2020.16. 2.001.
16. S. O. Alketbi and A. Diab, "A Systems Engineering Approach to Predict the Success Window of FLEX Strategy under Extended SBO Using Artificial Intelligence," J. Korean Soc. Syst. Eng., vol. 16, no. 2, pp. 97-109, Dec. 2020, doi: 10.14248/ JKOSSE.2020.16.2.097.
17. P. K. Gyan and J. C. Jung, "Development of a Functional Complexity Reduction Concept

of MMIS for Innovative SMRs," J. Korean Soc. Syst. Eng., vol. 17, no. 2, pp. 69-81, Dec. 2021, doi: 10.14248/JKOSSE.2021.17.2.069.

18. F. Randiki and J. C. Jung, "Concept Development of a Simplified FPGA based CPCS for Optimizing the Operating Margin for I-SMRs," J. Korean Soc. Syst. Eng., vol. 17, no. 2, pp. 49-60, Dec. 2021, doi: 10.14248/JKOSSE.2021.17.2.049.
19. "APR-1400 Design Control Document Tier 2." Aug. 2018.
20. B. M. Adams et al., "Dakota, A Multilevel Parallel Object-Oriented Framework for Design Optimization, Parameter Estimation, Uncertainty Quanti?cation, and Sensitivity Analysis: Version 6.0 User's Manual," p. 331, 2014.
21. S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," Neural Comput., vol. 9, no. 8, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.
22. K. Cho, B. van Merrienboer, D. Bahdanau, and Y. Bengio, "On the Properties of Neural Machine Translation: Encoder-Decoder Approaches." arXiv, Oct. 07, 2014. Accessed: Sep. 10, 2022. [Online]. Available: http://arxiv.org/abs/1409.1259
23. D. Ciresan, U. Meier, and J. Schmidhuber, "Multi-column deep neural networks for image classification," in 2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, Jun. 2012, pp. 3642-3649. doi: 10.1109/CVPR.2012.6248110.
24. S. Lawrence, C. L. Giles, Ah Chung Tsoi, and A. D. Back, "Face recognition: a convolutional neural-network approach," IEEE Trans. Neural Netw., vol. 8, no. 1, pp. 98-113, Jan. 1997, doi: 10.1109/72.554195.
25. M. Matsugu, K. Mori, Y. Mitari, and Y. Kaneda, "Subject independent facial expression recognition with robust face detection using a convolutional neural network," Neural Netw., vol. 16, no. 5-6, pp. 555-559, Jun. 2003, doi: 10.1016/S0893-6080(03)00115-1.
26. S. Bai, J. Z. Kolter, and V. Koltun, "Deep Equilibrium Models," p. 12.
27. US NRC, "Best-Estimate Calculations of Emergency Core Cooling System Performance, Regulatory Guide 1.157.," p. 20, 1989.
28. International Electrotechnical Commission (2015) IEC 62279:2015 Railway applications Communications, signaling and processing systems - Software for railway control and protection systems.