ABSTRACT

LEE, JOOMYUNG. Development of the Machine Learning-based Safety Significant Factor Inference Model for Advanced Diagnosis System (Under the direction of Dr. Nam Dinh).

Diagnosis is the process of identifying the nature of abnormal occurrences to provide the information for predicting the consequences in the operation of the management and control system. The Nearly Autonomous Management and Control (NAMAC) system is an integrated system that delivers recommendations to the operators to make a decision through diagnosis, prognosis, and strategy assessment based on the Artificial Intelligence (AI) technical support for accident management. The AI-guided diagnosis, the primary stage in the NAMAC system, aims to identify the plant damage states by using the Machine Learning (ML) algorithm.

In this dissertation, the ML-based Safety Significant Factor Inference Model (SSFIM) is developed to infer the safety significant factor (SSF) by training the Recurrent Neural Network (RNN), used to deal with sequential data, during transient in Experimental Breeder Reactor II (EBR-II). In the research, the fuel centerline (FCL) and cladding temperatures (CL), directly risk-related variables, are specifically designated as the SSF, a surrogate index used to represent the physical damage states during a transient. To generate the training dataset for the ML algorithm, GOTHIC simulates the transient resulting from loss-of-flow (LOF) accident owing to the pump operation status in a simplified EBR-II reactor. The objective of the ML-based SSFIM is accurately inferring the SSF in real time from the measured physical variables, even though the measurement data includes the random noise.

There are several technical challenges in development of the ML-based model: data insufficiency problems, algorithm adequacy, and training process issues. Some sub-assumptions under the data generation process are established to cover data insufficiency issues. Potential challenges from algorithm adequacy and training process problems are solved by the confirmation of hypotheses through the first case study. Also, three major capabilities of the ML model are identified to be used in the AI-guided diagnosis model.

Firstly, by using the RNN, the ML-based SSFIM is well trained to infer the FCL temperature within 1% accuracy, even if there is 5% random noise in measurement. Secondly, the adapted ML-based SSFIM is able to map the temperature distribution in the reactor system from one mass flow rate data and become the faster-than-real-time inference model by predicting the one second later SSF from the current measurement. Lastly, the ML-based SSFIM shows good

inference ability through the extrapolation test in which it covers ±20% of pump operation status even though the transient scenarios are out of range of the training dataset, however, there is a limitation that model performance becomes poor when the extrapolated range in testing is far from the training dataset. The SSFIM is completed as a knowledge-based data-drive model since the ML model is developed by not only the AI techniques but also the thermal-hydraulic computational code result.

The proposed developmental workflow and developed robust SSFIM contribute to the design of the AI-guided diagnosis model in the NAMAC system. By using the ML algorithm, plant damage states can be inferred from the measured physical variable in the reactor. The physical variable's change range in the normalized scale is significant to select the input feature for the ML model. In this work, the mass flow rate at pump #1 is the most powerful input variable because of reverse flow. To improve the SSFIM's model performance, there are several suggestions: 1) reduction of the sensor noise, 2) subdivision of the physical variables in the normalization process to train the model, 3) head event study to generate a large amount of data, and 4) identification of the component's operation status. Then, the ML-based SSFIM is expected to be utilized to develop the advanced AI-guided diagnosis model in various ways.

Development of the Machine Learning-based Safety Significant Factor Inference Model for Advanced Diagnosis System

by
Joomyung Lee

A dissertation submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Nuclear Engineering

Raleigh, North Carolina
2020

APPROVED BY:

_____	_____
Dr. Nam Dinh Committee Chair	Dr. Kostadin Ivanov
_____	_____
Dr. Maria Avramova	Dr. Igor Bolotnov
_____	
Dr. Abhinav Gupta	

DEDICATION

To my family.

BIOGRAPHY

Joomyung Lee was born in Seoul, Republic of Korea, in 1984. He was brought up with love and care from his family. He studied Nuclear Engineering at Hanyang University for both a bachelor's degree in 2011 and master's degree in 2014. His specialized field in M.S. degree was Risk-Informed Emergency Preparedness based on level 3 PRA.

After completing his master's degree, he decided to keep studying for a Ph. D course in the United States. He started his doctoral study in Nuclear Engineering at North Carolina State University in Fall 2014, under the supervision of Professor Nam Dinh. His research focuses on developing the Machine Learning Model to diagnose the plant damage states during operational states or accident conditions to design the AI-guided Diagnosis Model for contribution to the development of the Nearly Autonomous Control and Management system.

ACKNOWLEDGMENTS

Foremost, I would like to express my special thanks of gratitude to my advisor Prof. Nam T. Dinh for giving me the opportunity to do research and providing invaluable guidance throughout this research. His guidance to machine learning sphere was a great opportunity for me. Also, his patience, vision, sincerity, immense knowledge and motivation have deeply inspired me. It was a privilege and honor to work and study under his teachings. He is my great mentor.

I cannot express enough thanks to my committee for their continued support and review my thesis: Prof. Kostadin Ivanov, Prof. Maria Avramova, Prof. Igor Bolotnov, and Prof. Abhinav Gupta. They encourage me with insightful comments to move forward.

This research was performed with support from the project funded by Advanced Research Projects Agency-Energy (ARPA-E). Without the financial support, my completion of this thesis could not have been accomplished. My sincere thanks also go to Dr. Pascal Rouxelin and Dr. Paolo Balestra, for offering me transient data of the simplified EBR-II model. The system simulation used GOTHIC whose license is provided by Zachry Nuclear Engineering, Inc. GOTHIC incorporates technology developed for the electric power industry under the sponsorship of EPRI, the Electric Power Research Institute. Without their work for the NAMAC project, the ML-based SSFIM would not be developed. Next, I want to thank my colleagues, Linyu Lin, Paridhi Athe, Yangmo Zhu, Han Bao, Anil Gurgen, Longcong Wang, Botros Hanna, Chih-Wei Chang, Yeojin Kim, and many others. I spent valuable time with them during graduation school life.

Finally, to my caring, loving, and supportive wife, Jihyun Lee: my deepest gratitude. Without her, my life would have been unimaginable. Melodie, my lovely daughter, I love you. I am extremely grateful to my parents for their love, prayers, caring and sacrifices for educating and preparing me for my future. Also, I express my thanks to my parents in law for their valuable support.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ACRONYMS

ADAM	Accident Diagnostic, Analysis and Management
AI	Artificial Intelligence
ANN	Artificial Neural Network
AOO	Anticipated Operational Occurrences
ATOM	Autonomous Transportable On-demand reactor Module
BDBA	Beyond Design Basis Accident
CAWRLRL	Cosine Annealing Warm Restart Learning Rates
CDF	Core Damage Frequency
CL	Cladding
CLR	Cyclical Learning Rates
CNN	Convolution Neural Network
CPAG	Control Parameter Assessment Guidelines
DBA	Design Basis Accident
DBN	Dynamic Bayesian Network
DMS	Diagnosis Module Storage
DQN	Deep Q-Network
DRS	Data Record Storage
DT	Digital Twin
DT-D	Digital Twin – Diagnosis
EAAG	EOP/SAG Action Assessment Guidelines
EBR	Experimental Breeder Reactor
EEG	Electroencephalogram
EOP	Emergency Operation Procedures
ERO	Emergency Response Organization
FCL	Fuel Centerline
FLB	Feedwater Line Break
FNN	Feedforward Neural Network
FSAG	Function Status Assessment Guidelines
FSDDM	False Sensory Data Detection Model
GAIA	Genuinely Autonomous Intelligence for ATOM
GRU	Gated Recurrent Unit
HPLP	High Pressure Lower Plenum
I&C	Instrumentation and Control
IHX	Intermediate Heat Exchanger
LOF	Loss of Flow
LPLP	Low Pressure Lower Plenum
LSTM	Long Short-term Memory
LWR	Light Water Reactor

M/G	Motor Generator
MAPE	Mean Absolute Percentage Error
MCR	Main Control Room
ML	Machine Learning
MSE	Mean Square Error
NAMAC	Nearly Autonomous Management and Control
NRC	Nuclear Regulatory Commission
NTTF	Near-Term Task Force
PAFO	Passive and Autonomous Frequency control Operation
PCA	Principal Component Analysis
PCC	Pearson Correlation Coefficient
PRA	Probabilistic Risk Assessment
PSAG	Plant Status Assessment Guidelines
PWR	Pressurized Water Reactor
RISARD	Risk-informed Severe Accident Risk Diagnosis
RL	Reinforcement Learning
RLRP	Reduced Learning Rates on Plateau
RMSE	Root Mean Square Error
RNN	Recurrent Neural Network
RPV	Reactor Pressure Vessel
SAMEX	Severe Accident Management Support Expert
SAMG	Severe Accident Management Guidelines
SBDS	Symptom Based Diagnostic System
SGTR	Steam Generator Tube Rupture
SHRT	Shutdown Heat Removal Test
SIS	Structure, System, or Components' Information Storage
SLR	Steam Line Rupture
SMR	Small Modular Reactor
SOARCA	State-of-Art Reactor Consequence Analysis
SSC	Structure, System, or Components
SSF	Safety Significant Factor
SSFIM	Safety Significant Factor Inference Model
TSC	Technical Support Center
UP	Upper Plenum

NOMENCLATURE

Machine Learning and Model Development Field

α	Learning Rate
γ	Discount Factor
σ	System Noise Percentage
A_t	Actual Value
b	Bias
c_t	Final Memory Cell in LSTM
$\widetilde{c_t}$	New Memory Cell in LSTM
f_t	Forget Gate in LSTM
h_t	Hidden State in LSTM Activation in GRU
$\widetilde{h_t}$	Candidate Activation in GRU
i_t	Input Gate in LSTM
I_t	Inferred Value
J	Objective Function or Cost Function
k_t	Training Termination Inducing Number
k_u	Patience Number
MP	Model Performance
o_t	Output Gate in LSTM
r_t	Reset Gate in GRU
r_{xy}	Pearson Correlation Coefficient between x and y
VL	Validation Loss
w_i	Synaptic Weight
z_t	Update Gate in GRU

Thermal-hydraulic Field

α_l	Doner Cell Phase Fraction
ρ	Density
μ_l	Dynamic Viscosity
c_p	Specific Heat
cn	Specific Conductor
A	Surface Area
D_h	Thermal Hydraulic Diameter
D_{cn}^e	Diffusion Coefficient
H_{conv_l}	Liquid Convective Heat Transfer Coefficient
h	Heat Transfer Coefficient
k	Thermal Conductivity
L	Characteristic Length
Nu	Nusselt Number

Pr	Prandtl Number
Q_{cond}	Condensation Heat Flux
Q_{conv}	Convective Heat Flux
Q_{rad}	Radiant Heat Flux
Q''^{n}_{conv}	Convective Heat Rate
Re	Reynolds Number
s^{e}_{cn}	Energy Source per unit wall area
T_{cn}	Temperature at Surface of Specific Conductor
T_{w}	Temperature at Wall
T_{l}	Liquid Temperature
u_{l}	Liquid Velocity
V_{cn}	Volume of the Conductor

CHAPTER 1. INTRODUCTION

1.1.Backgrounds

To operate a nuclear power plant safely, risk, which is caused by various accident scenarios, should be managed, to abide by the 'better safe than sorry' principle. Risk is used as an index to evaluate nuclear safety. It is satisfied by both societal acceptances to design and operating in public criteria [1]. Risk management entails a decision-making process to resolve the problems with risk assessment and allow the operator to decide based on their understanding of the situation. In the risk management point of view, decision-making indicates the response strategies against not only accident conditions but also operational states.

The diagnosis requires precise measurement and a reliable process, as it is the primary stage in predicting consequences. Generally, the diagnosis includes not only the capture of unnatural signals in the system but also the interpretation of the problem, which is revealed by the symptoms. An operator must understand the overall system operation status to manage and control the nuclear power plant.

This research focuses on the development of a Machine Learning (ML) based inference model by using advanced techniques during transient to provide decent information for the Artificial Intelligence (AI) guided diagnosis model. Currently, a variety of state-of-the-art computer science techniques are offered for the development of the management and control system from the engineering perspective. This thesis suggests a real-time inference model, which aims to diagnose the plant damage states from multiple physical phenomena in the reactor system by utilizing an ML algorithm.

1.1.1. Nearly Autonomous Management and Control System

Nearly autonomous management and control system (NAMAC) refers to a system that delivers recommendations by AI technical supports to operators for not only the safe operation but also accident management [2]. The NAMAC system is not a fully autonomous system but a nearly autonomous system because the system performs diagnosis and prognosis and provide a recommendation to the operator; however, the inherent right or authority to make any decision is still held by the operator. As we can see in figure 1, the system consists of three stages: 1) AI-

guided diagnosis to identify the plant operational/damage states, 2) simulation-based prognosis to forecast the consequences, and 3) evaluation of strategic recommendations to suggest an action to the operator. The system utilizes simulation-informed data-driven training, AI-guided inference, faster-than-real-time simulation, knowledge-based operating procedures, and severe accident management guidelines to aid in the establishment of the risk-informed decision-making process for normal operation, anticipated operational occurrences (AOO), Design Basis Accident (DBA), and Beyond Design Basis Accident (BDBA).

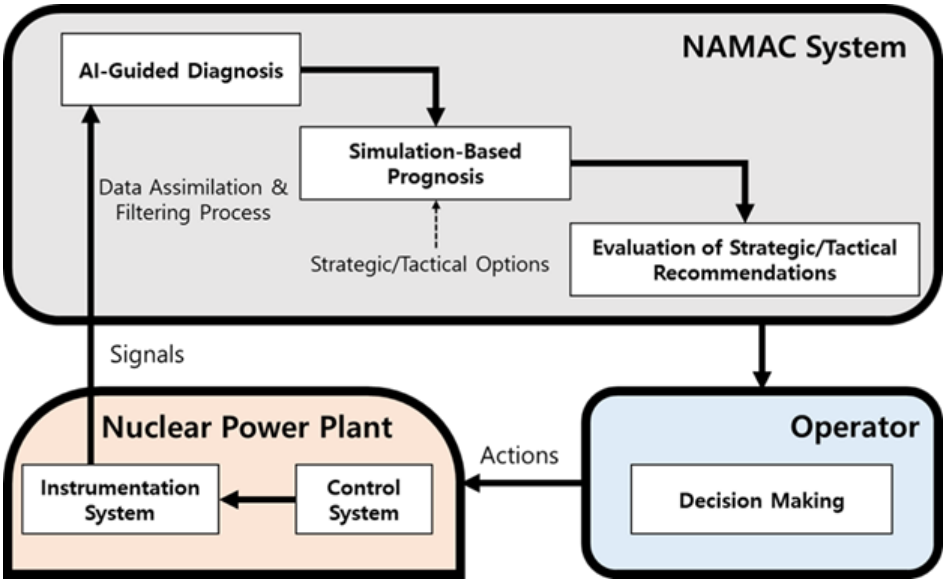


*Figure 1.*NAMAC System Conceptual Architecture

The NAMAC system is expected to bring a transition from an operator-centric plant control system to an AI-guided control system in the risk management regime. There are two main advantages: 1) reducing human error and 2) recommending optimized action. The NAMAC system performs rational analyses to avoid the human error which may be caused by an operator's high stress owing to the accident initiation and progression. By using a number of simulation data for the ML training, a suggested strategic recommendation would be the most appropriate and timely response needed to accomplish the intended goal. The process leads to close the gap between risk-informed decision making for a real plant and risk assessment by "unnecessary conservatism", which is based on predetermined conservative assumptions from traditional safety analysis [3].

1.1.2. Research Scope

For the optimized decision, an improved risk-informed decision-making process should be derived from a rapid and reliable diagnosis process. Through specifying the research scope, the utility and applicability of the model are examined. The research scope is divided into diagnosis scope and accident scope which should determine what (and how) to diagnose and what data to use. This section demonstrates a safety significant factor (SSF) and transient as a diagnosis scope and an accident scope, respectively.

- Safety Significant Factor

The research focuses on the inference of plant damage states based on measurable physical symptoms in the reactor system. The ML-based diagnosis model is used to infer the physical damage states of the reactor in real-time. In order to identify the physical damage states, selecting the obvious variable which represents the plant damage is required.

The SSF is introduced to indicate the physical damage states because of a transient or accident. It is used as a surrogate index to represent the potential risk. Since the risk can be identified when the fuel is damaged, another index is required to describe the plant damage states in transient or DBA.

One of the requirements of the SSF is representativity. Since safety is not observable, the risk is used to analyze the safety of the reactor operation. From a probabilistic risk assessment (PRA) perspective, one of the representative risk indices is core damage frequency (CDF), which is the quantified likelihood of fuel damage. However, the core damage cannot be estimated as a numerical value in real-time, hence, an SSF, which is a kind of physical variable that represents the condition of a structure or safety barrier directly related to the integrity of the fuel element, is suggested.

The other requirement of the SSF is obviosity. For example, radioactive material must be detected in the reactor after core damage, likewise, if containment fails, radiation will be released to the environment. In the Light Water Reactor (LWR), the cladding would be locally damaged owing to Zr oxidation when the temperature exceeds 1000 [K], moreover, above 3100 [K] of the fuel temperature, the core would collapse and be relocated by UO2 melting [4]. Figure 2 illustrates the core degradation process with temperature in the LWR. Such physical variables, which obviously indicate plant damage states, are referred to as SSF. Thus, not only the consequence

variables but also risk-related variables could be selected as the SSF. The SSF application to this study is described in the research assumption section.

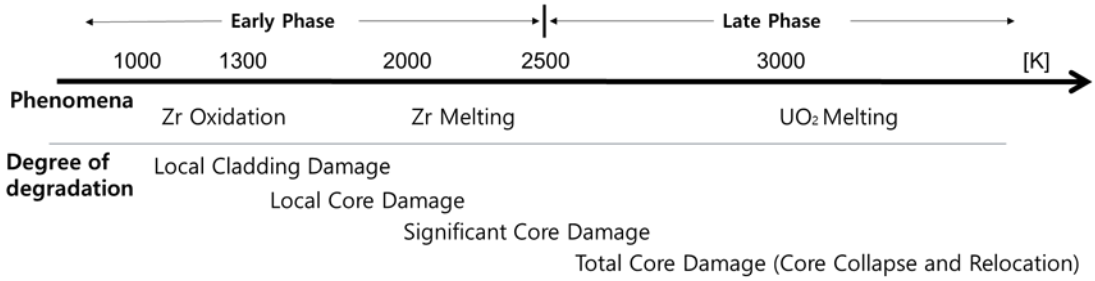


Figure 2. Core Degradation Process in the Light Water Reactor

- Transient

In the research, a transient is the target operation status used to diagnose the plant damage states. As illustrated in table 1, a nuclear power plant state is divided into two states: operational states, which consist of normal operation and AOO, and accident conditions DBA and BDBA, including severe accident [5]. Transient is also known as AOO which does not induce any significant challenge but might be the initial stage needed to evolve to DBA or BDBA. Although a severe critical effect, such as fuel damage, is not considered in the transient, the abnormal condition has a much higher frequency of occurrence (10-2~1/reactor year) than accident conditions [6].

Table 1. Plant State Description, Adapted from 'Considerations on the Application of the IAEA Safety Requirements for the Design of Nuclear Power Plants (p. 13),' by IAEA, 2016, Copyright 2016 by the IAEA

<table>
<tr><th colspan="2">Operational States</th><th colspan="3">Accident Conditions</th></tr>
<tr><td rowspan="2">Normal Operation</td><td rowspan="2">Anticipated Operational Occurrences (Transient)</td><td></td><td colspan="2">Beyond Design Basis Accident</td></tr>
<tr><td>Design Basis Accident</td><td>a</td><td>Severe Accident</td></tr>
</table>

[a] Beyond design basis accidents without significant core degradation

Since any visible critical effects related to risk indices owing to safety challenges are not apparent in the operational states, in order to infer the physical damage, a proper SSF should be determined in the transient realm. In the nuclear reactor system management and control system, a risk indicates the fuel damage or radiation release, which are directly related to risk. For the

severe accident conditions, core damage or fission product release amount would be used as the SSF from the PRA perspective, whereas, the variable that represents core integrity should be chosen as the SSF in both DBA conditions and operational states in which the core is not damaged yet. According to the PRA report, core damage is caused by fuel melting with cladding damage or fuel pin failure due to cladding breach by eutectic attack. Therefore, fuel centerline (FCL) temperature and cladding (CL) temperature would be appropriate SSF.

1.1.3. Research Objective

The main objective is to develop the ML-based SSF inference model (SSFIM) for the AI-guided diagnosis model in the NAMAC system. To support the main objective, there are three goals:

- ✓ development of the ML-based model to infer the plant damage state,
- ✓ identification of the input dataset to infer the plant damage state, and
- ✓ extension from the given partial information to unknown information by using the ML-based inference model.

The multiple case studies will be implemented according to the component requirements, which correspond to the specific objectives. Component requirements that imply the essential features to be taken into account for model development determine the expected functions of the model, inputs, and outputs in an operational workflow, and simulation data in accordance with the research scope. Further details are in chapter IV.

1.2.Motivation

As both the current data and established knowledge are necessary to identify a problem, the measured symptom data is also indispensably evaluated in the diagnostic process to determine the quality of the data sets. The diagnostic procedure is a process used to figure out the reason or SSF based on observed physical symptoms or measured physical variables in the plant. If the input data is not true or intact, even though a well-trained model is prepared to get a correct output, the output from the model must be wrong; in computer science, there is an old phrase “Garbage In, Garbage Out”. Therefore, not only developing the model for the best performance but also evaluating the input datasets for meaningful information extraction are preconditions for getting a reasonable output value.

- Black Swan Theory

Classification of the diagnosis problem refers to the Black Swan Theory and Rumsfeld's remark in which the problem is perceived based on awareness and established knowledge. Black Swan Theory introduces the concept that an event is often inappropriately rationalized by hindsight after the fact as if it could be expected. Taleb used the following metaphor to explain the theory: "People believed that all swans are white until a black swan is spotted in Australia. It obviously shows a limitation to our knowledge from experience or observation." [7]. The theory warns us not to blindly believe our knowledge which is established by the experience so far. Also, Donald Rumsfeld, who was the United States Secretary of Defense, has remarked about recognizing a problem based on awareness and knowledge in 2002 [8]. Table 2 shows how the problem is categorized into four cases: known knowns, known unknowns, unknown knowns, and unknown unknowns [9]. Both theories emphasize the necessity of prudential evaluation of existing knowledge and observation for the problem recognition before finding its corresponding solution.

Table 2. Rumsfeld's Matrix

Knowledge / Awareness	Known	Unknown
Known	Known knowns (Recognize the problem and utilize the solution)	Known unknowns (Recognize the problem and examine the solution)
Unknown	Unknown knowns (Expose the knowledge to practice)	Unknown unknowns (Need more experience)

Likewise, before diagnosing the plant damage states from the physical variable signals, the incoming data sets must be evaluated to assure whether the signal dataset is intact and knowledgeably consistent for the diagnostic procedures. The drawback in a signal dataset is caused by sensor malfunction with false signals or missing signals. Firstly, if one or more missing signal resides in the incoming dataset, the intactness of the dataset is broken. A signal dataset is composed of measured physical variables, such as temperature, pressure, and mass flow rate, by installed

devices in the reactor or the plant and the intactness of the dataset is determined by the number of signals. When any of the signals are not able to be received by sensor failure while physical symptoms are sequentially observed, the signal dataset is regarded as damaged. Secondly, false signals in the dataset must give rise to knowledge-based or physics-based inconsistency between signals. Even in transient or accident progression, physical symptoms should retain the consistency of each other in a confined volume. For instance, from a knowledgeably reasonable perspective point of view, the core outlet temperature surely increases when the core inlet mass flow rate decreases. Since one of the reasons for a wrong diagnosis result is using an input dataset that includes false signals, the false signals should be eliminated in a diagnostic procedure.

According to the incoming dataset analysis, the initial step of the diagnostic procedure is influenced. Table 3 describes the solving process with problem classification. In the first case, if a signal dataset is intact and signals are knowledgeably consistent within a dataset, using a proper logical diagram is a straightforward solution under the obvious conditions. In the second case, if there are some missing signals but the rest of the signals are correct, diagnosis is implemented based only on given information. Thirdly, in the case of diagnosis from an intact dataset including some false signals, reasonable physical variable data should be extracted as input for the diagnostic procedure. Lastly, when some signals are absent in a dataset and the given signals are not knowledgeably consistent, that is the most complicated case, a plausible scenario would be presumed.

Table 3. Problem Classification and Solving Process in Diagnostic Procedure

Consistency / Intactness	Signals are knowledgeably consistent (Incoming signals are correct)	Signals are knowledgeably inconsistent (False signals are included)
Signal dataset is intact (No Missing Signals)	Use logical diagram	Diagnosis with extracted reasonable information
Signal dataset is damaged (Some Missing Signals)	Diagnosis with given information	Presume plausible scenario

1.3. Dissertation overview and outline

The current chapter demonstrates the background, research scope, objectives, and motivations of the research. The rest of the dissertation is structured as follows.

Chapter 2 describes the overview of the AI-guided diagnosis by introducing the ML algorithm, digital twin, and NAMAC diagnosis structure.

Chapter 3 depicts the developmental workflow with technical components. The workflow shows how the ML-based SSFIM is developed, what technical components are used, what potential challenges should be considered, and what assumptions are necessary to cover the challenges. The technical components include data generation, ML components, and training & testing process.

Chapter 4 includes case study objectives formulated to specify the research objectives and component requirements to achieve the specified objectives.

Chapter 5 provides the case study results. There are four case studies: 1) building the ML algorithm for developing the ML-based SSFIM, 2) inferring the SSF by using only a few measured physical variables, 3) extending the diagnosis range by using the ML-based inference model, and 4) Identifying the inference ability of the ML-based inference model.

Chapter 6 analyzes the ML-based SSFIM's model performance based on case study results. It demonstrates the notable findings and capabilities of the ML-based inference model.

Chapter 7 demonstrates the conclusion with contributions and recommended future tasks for advanced AI-guided diagnosis model.

1.4. Glossary

Diagnosis

Diagnosis is the process of identifying the nature of abnormal occurrences or the distinctive characterization in the system to analyze the problem based on the symptoms. An operator who fully understands the overall system operation status diagnoses the complicated system of the nuclear power plant.

AI-guided Diagnosis Model

An AI-guided diagnosis model indicates that diagnosis model based on ML algorithm analyzes data or information instead of a human operator. AI technology is able to reduce stress which is one of the major causes of human error because computer models are unaffected by psychological effects and stress in analyzing the accident phenomena during abnormal operation status or accident conditions.

Risk-informed Decision Making

Risk-informed decision making is defined as decision-making that is implemented not solely by risk but by a variety of other considerations while taking account of inevitable uncertainties. Since a decision is derived by integration of technical and nontechnical elements from human judgement, risk, which is technically assessed by an engineering approach, is used as an input in the decision-making process.

Safety Significant Factor

The SSF is a physical variable used to represent the physical damage states caused by transient or accident progression. The fuel matrix and cladding tubes protect the fuel from the core damage by the accident conditions. Since the condition of the safety barriers depends on their temperature, the SSF is able to indicate the barriers' states to represent the plant state. It is useful to denote the core conditions in the transient and DBA realm. As mentioned above, the SSF must be chosen by considering the representativity and obviosity.

Physical Symptoms

In the nuclear power plant, the number of physical and chemical reactions are revealed through physical symptoms which are used to describe the plant states. Physical symptoms in an abnormal operation, in which obvious evidence are detected and used to analyze the accident conditions, are a result of the aggregation of a change in the physical variables, for instance, a temperature increase, depressurization, or over-pressurization.

Validation

Generally, validation refers to identification for ensuring that the result or requirement satisfies the expectation at the end of the process. It is usually used with a "verification," which indicates the process that identifies the established requirements are met in testing at a specific stage during development, from the validation and verification (V&V) perspective. However, in the ML field, the validation process is a procedure to evaluate the model performance from the validation dataset in the training process. Therefore, from the macroscopic development perspective, the validation of the ML model is equivalent to the verification process in developing the AI-guided diagnosis model.

Knowledge-based, Rule-based, Logic-based, and Skill-based

Human behavior in decision-making is carried out by behavior mode: knowledge, logic, rule, and skill. The modes are classified by conscious level which is determined by how much practice to execute the actions [10, 11].

As figure 3 illustrates below, a 'knowledge-based' is a rudimentary area for technical support used to make a decision. A 'logic-based' concept is used in the procedures or instructions that are logically determined by relevant knowledge. A 'rule-based' concept is enacted by accumulated knowledge with substantial and strong logic, and the 'skill-based' concept is completed through having a lot of experience. The knowledge is established by a variety of experiences over a long period of time, and logics or rules suggest some effective methods for learning the skills. The 'skill-based' concept, which is accomplished by professional and sufficient training and learning, allows the operator to decide an optimized action immediately when the operator recognizes an abnormal situation. Hence, the decision-making model developed by using a 'skill-based' approach improves the operator's technical expertise and ability to take appropriate and efficient action.

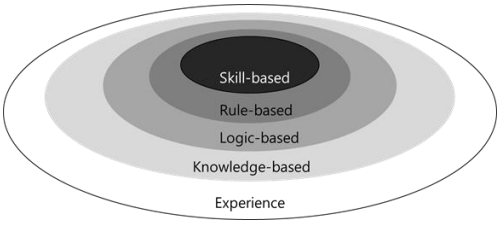


Figure 3. Knowledge, Logic, Rule, and Skill-based Areas

CHAPTER 2. AI-GUIDED DIAGNOSIS MODEL

This chapter demonstrates the technical backgrounds of the ML-based SSFIM. In section 2.1, ML is introduced with the concept of Artificial Neural Network (ANN), and deep learning. In section 2.2, AI-guided diagnosis with a digital twin, which is a part of the NAMAC system, is introduced based on the operational workflow. Lastly, previous studies by using the ML algorithm for the diagnosis field are introduced to compare with the ML-based SSFIM.

2.1. Machine Learning

ML is a broad concept of techniques that provide various support for identifying and solving problems by learning from data. ML, which is a sub-discipline of AI studies, analyzes the data by using an algorithm, learns from the analysis, and predicts or decides based on the learning. The objective is not directly coding the specific guidelines for decision making but becoming proficient at methods used to solve the problem by learning through an algorithm with a large amount of data. In other words, general computer programming produces outputs from inputs through an established program, whereas an ML creates the program itself from inputs and outputs through an efficient algorithm. The evolution of AI is realized through the development of an advanced algorithm instead of improving the capabilities of the computer program.

Generally, ML is divided into three categories: supervised learning, unsupervised learning, and reinforcement learning. The subfields are classified according to the characteristics of information that are the subject of the algorithm [12]. Figure 4 shows the types of ML.

- Supervised learning is a learning method in which labeled data are provided in training datasets.
  - ✓ The objective of supervised learning is to establish a rule that ascertains the relationship between input and output data.
  - ✓ After training, test sets, which are not labeled datasets, are used to evaluate the prediction ability of the algorithm.
  - ✓ Generally, supervised learning applies to classification tasks, in which it is used to categorize data depending on patterns by predicting discrete values, and regression problems, in which it is used to predict continuous numbers of value.

- Unsupervised learning is a learning method used to recognize hidden patterns from unlabeled datasets.
  - ✓ Unsupervised learning aims to discern the hidden features or structure needed to sort input data.
  - ✓ Since the outputs are not judged right from wrong, it is difficult to evaluate the training process and completeness. However, especially in the text/image processing sphere, unsupervised learning is a promising technique to analyze the visual data.
  - ✓ The clustering algorithm is a representative application of unsupervised learning.
- Reinforcement learning is a learning method in which 'rewards' from 'actions' motivates the 'agent' to learn based on the trial and error approach.
  - ✓ The goal of reinforcement learning is to acquire an optimized solution when the obtained reward is a maximum value in a specific 'environment'.
  - ✓ In contrast to supervised learning or unsupervised learning, the reinforcement learning algorithm includes collecting data from dynamic environments. The learning is proceeded by the agent's action which determines the variable rewards in a given state.
  - ✓ Deep Q-Network (DQN) and policy gradient are state-of-the-art algorithm for reinforcement learning.

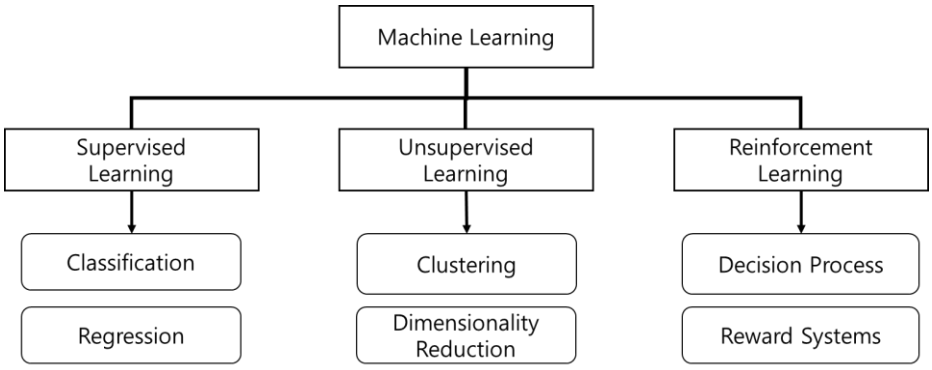


Figure 4. Types of Machine Learning

Supervised learning is an appropriate method for predicting the results that consider a relationship between cause and effect. The supervised learning model produces outputs that respond to unknown input data in the same realm or about the same subject. The ML is assessed by representation, which indicates data analysis to learn patterns or features, and generalization,

which infers the unknown data process. Hence, the quality of the ML system is determined by the suitability and effectiveness of the algorithm, and various and generalized training datasets that are completed through data processing.

2.1.1. Artificial Neural Network

ANN is one of the most used computational models in the ML field. The ANN model is inspired by the human brain, so the model constitution is analogous to aggregation of each artificial neuron, which is also called perceptron [13]. Figure 5 shows a single-layer perceptron that is comprised of an input vector (x_i), bias (b), a synaptic weight (w_i), an activation function (f), and output. Neuron of input indicates a feature of the input data in which influential characteristics are reflected through weight in each edge. The activation function changes input signal into output signal according to characteristics of the function. Through a simple mathematical term, a perceptron is described by the following equation:

$$output = f\left(\sum_{i=1}^{n} w_i x_i + b\right) \tag{1}$$

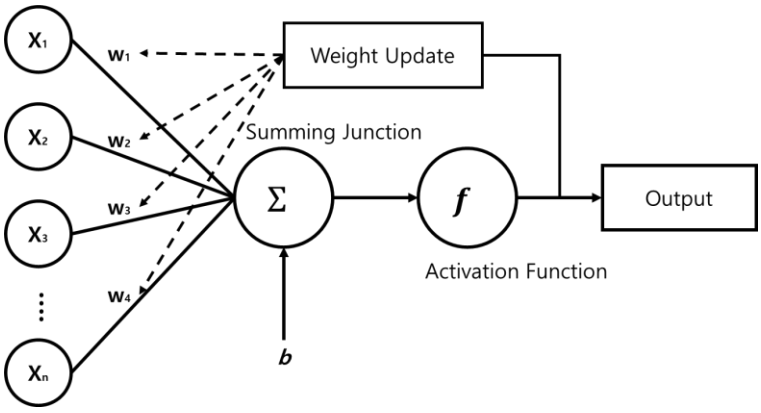


Figure 5. Single Layer Perceptron with One Output

Feedforward Neural Network (FNN), in which information flows forward from the input layer to the output layer, is shown in figure 6. The hidden layer, which is required to solve linearly inseparable problems, is a bridge between the input layer and the output layer for filtering and transforming input data. The user must define the proper amount of the number of nodes and layers to avoid underfitting and overfitting the problem. Supervised learning in neural network follows

these steps to make a ML model: 1) setting up the training datasets that contain input and output data, 2) training a model by training sets, 3) calculating the error between data and output, and 4) optimizing weights by gradient method with backpropagation. The ANN learning aims to seek a weight that minimizes the difference between correct values and calculated outputs. In the regression task, the Mean Square Error (MSE) method is utilized to compute the loss. MSE is calculated by the equation, where n is the number of samples, y_i is output value from the neural network, and $\bar{y}_i$ is labeled value in the training dataset.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \bar{y}_i)^2 \quad (2)$$

With the MSE calculation, which is also called as an objective function (J), a gradient descent algorithm is used to find a minimum value of weight in the network. Gradient descent equation is as follows, when learning rate (α) is given:

$$w = w - \alpha\nabla_w J(w)$$

The learning rate is an important hyper-parameter to train the model. Details are demonstrated in Chapter III.

As the ANN sphere is developed rapidly, a number of high techniques and algorithms are introduced to upgrade the process to model the neural network.

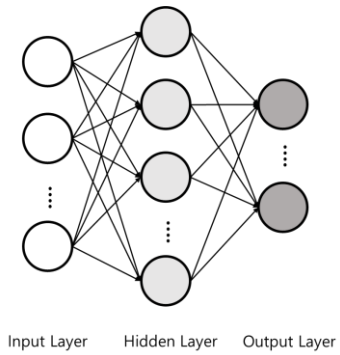


Figure 6. A Single Hidden-layer Feedforward Neural Network

2.1.2. Deep Neural Network

ANN, which has two or more hidden layers, is called the Deep Neural Network (DNN). Figure 7 describes the DNN structure. Due to different features that are learned in each layer while input data go through the hidden layers, DNN is able to extract the complicated features better than

shallow ANN [14]. For instance, as figure 8 shows, simple and detailed features (e.g., horizontal line, vertical line, or diagonal line) are learned in a lower layer and complex and abstractive features (e.g., facial features, car shapes, or airplanes) are learned in a higher layer [15]. Through these processes, in which a number of parameters are involved for using the non-linear function, DNN comprehends high-level datasets. The DNN development has led to a drastic improvement in object recognition, speech recognition, and game field [16-18]. A detailed description of the DNN components can be found in Appendix A.

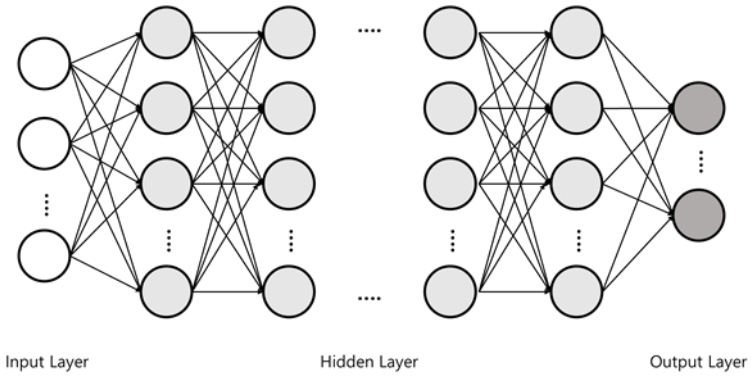


Figure 7. Deep Neural Network

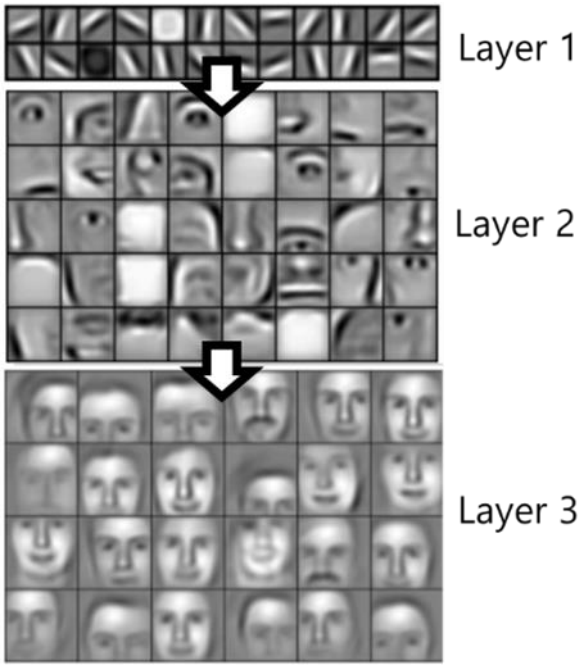


Figure 8. Hierarchical Object-Part Representations in the Convolutional Deep Belief Networks, Reprinted from "Convolutional Deep Belief Networks for Scalable Unsupervised Learning of Hierarchical Representation" by H. Lee et al, 2009

2.1.3. Recurrent Neural Network

The RNN is a network in which compositions of units are connected to each other by recursive relationships in a hidden layer. RNN takes not only input data but also previous information into account to deal with time-series data for calculating output. The network recognizes a dynamic pattern and extracts information by performing the same task for every element of a sequence through a feedback loop network form [19]. RNN is a type of deep learning which has been recently applied for language modeling, machine translation, and stock price prediction [20-22]. Figure 2-5 shows that RNN contains a feedback loop, which allows the information to go through the same hidden states alike the multiple-same layers. The process enables the hidden layer to memorize all the information that is learned from both the input data and previous states [14]. The following are explanations and attributes of the computation process of RNN:

- x_t is the input at time step t.
- S_t is the hidden state at time step t. The equation describes the hidden state, in which the parameters (U, V, W), input, and the previous state are variables in the activation function (f).

$$S_t = f(Ux_t + WS_{t-1}) \tag{3}$$

- Y_t is the output at time step t.

$$Y_t = VS_t \tag{4}$$

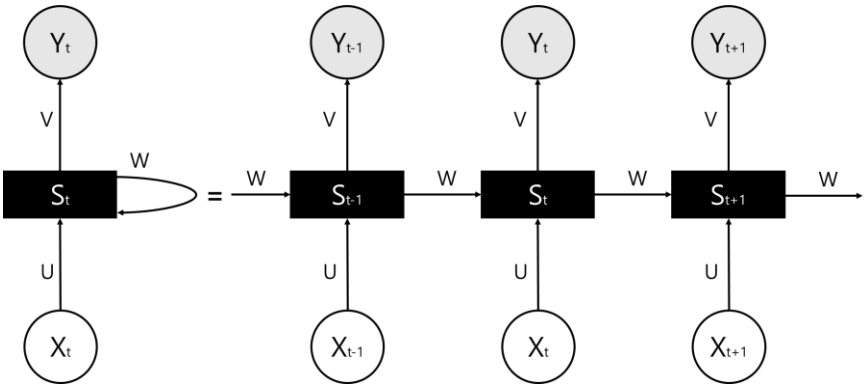


Figure 9. Recurrent Neural Network Architecture

A detailed description of the RNN and advanced modules can be found in Appendix A.

2.2. AI-Guided Diagnosis Model

2.2.1. Digital Twin

Digital twin is a knowledge-based virtual environment in which all of the data and related information about the actual system with various analysis models are prepared to operate, control, and manage the real one. Digital Twin Diagnosis (DT-D) class is a constituent element of digital twin hub to support diagnostic procedure in the management and control system.

DT-D class supports the diagnosis engine during transient or accident progression by providing related information and diagnosis modules. Figure 10 shows the DT-D class which contains three types of storage: Data Record Storage (DRS), Structure, system, or components Information Storage (SIS), and Diagnosis Module Storage (DMS). Digital twin continually saves the sequential physical measurement dataset in the DRS to convey the operation history to the diagnosis engine. The SIS stores the manufactures' information, such as a pump model, valve type, measuring device model, and so on, and system/structure related information. The DMS possesses a variety of computational diagnosis modules to cover the full range of operation conditions, from normal operation to a severe accident. These storages each have a goal and role in the advanced diagnostic procedure.

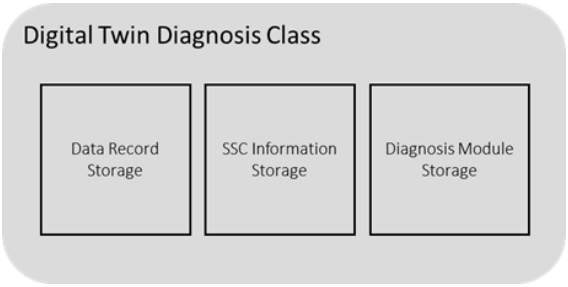


Figure 10. Digital Twin Diagnosis Class Structure

Storages in the DT-D Class supply data, information, and modules to operate knowledge-based, simulation data-driven, AI-based, and highly modularized diagnosis engine. Table 4 describes the characteristics, types, sources, and purposes of the contents in each storage. As mentioned above, DRS provides operation history, which is aggregated by measurement records, as an input dataset to AI-guided diagnosis engine to infer unobservable physical variables that directly indicate the plant damage states. In the SIS, the knowledge-based and design-based

structure, system, or components (SSC) information from the documents, such as Design Criteria Documentation or PRA Report, are kept and updated to deliver relevant information depending on the transient or accident condition. DMS secures ML models trained by the ML algorithm with simulation data to be used as an execution core in the diagnosis engine. Since AI-based models are applied in the form of modules to build a core engine, the DMS contributes to improve the modularity of the AI-guided diagnosis model.

The DT-D Class records and updates the sensory data and SSC information respectively from the actual plant in normal operation, however, in a transient or accident phase, it offers saved data, information, and developed modules to establish the diagnosis engine (see figure 11). Sensory data signifies both physical variable data and a components' operational status signal to check on pump failure, electric supply failure, and so on. SSC information can be updated in accordance with a maintenance plan or amendment of system details during operation, while diagnosis modules should be prepared before operation through a developmental workflow for the AI-guided diagnosis model.

Table 4. DT-D Class Storage Description

	Data Record Storage (DRS)	SSC Information Storage (SIS)	Diagnosis Module Storage (DMS)
Contents Characteristics	Record-based Data	Knowledge-based and Design-based Information	Machine Learning-based Model
Contents Type	Physical Variables Data	SSC Information	Computational Model
Contents Source	Installed Measuring Devices	Existing Documentations and Maintenance Plan (Design Criteria Documentation, PRA Report, and so on)	Machine Learning Training with Simulation Data
Contents Purpose	Support Input Dataset (Operation History) for Diagnosis Engine	Support related information for Diagnosis Engine	Execution Core in Diagnosis Engine

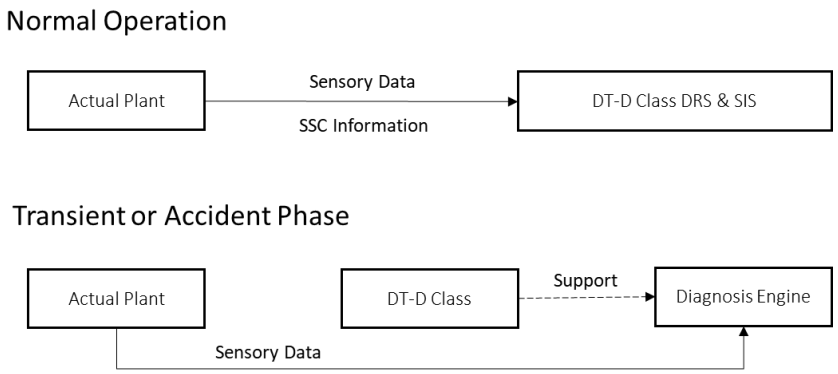


Figure 11. Role of Digital Twin Diagnosis Class

2.2.2. Diagnosis Engine

An AI-guided diagnosis engine aims to identify the plant damage states which are represented by a dominant event, SSC availability and SSF in the reactor system. In other words, the diagnosis engine includes three modules to achieve an individual objective. Therefore, the engine should be designed with a slot-modular structure, which is shown in Figure 12, on account of the unique interface that each element achieves the specific outcome from shared input datasets [23]. The sensory data and operation history are used to infer the physical damage states by using Module I; identify the event type and progression states by using Module-II; and search the useful SSC information to plant action strategy by using Module-III. Although, in order to diagnose the various types of transient or accident cases, developing modules cost a lot of time and effort, in virtue of high modularity, the diagnosis engine is flexibly transformed in accordance with conditions. Figure 13 shows the way to complete the diagnosis engine through module combinations.

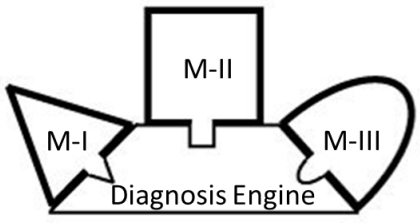


Figure 12. Diagnosis Engine Structure

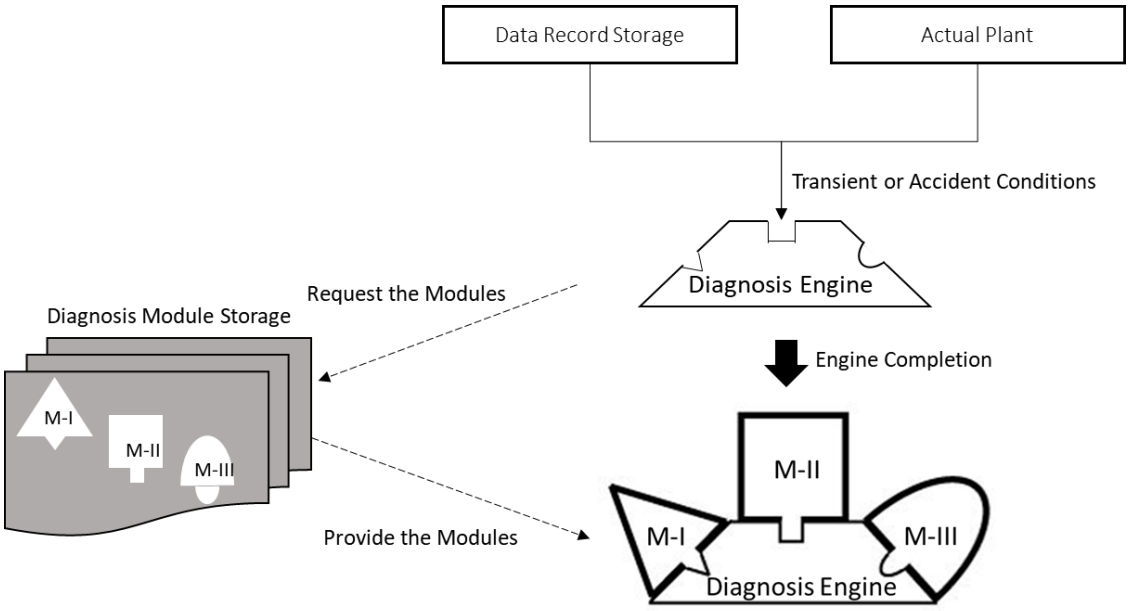


Figure 13. Diagnosis Engine Completion through Diagnosis Module Union

2.2.3. Operational Workflow

The AI-guided diagnosis model is implemented to not only infer the current plant damage states but also verify the effect of an executed action plan as a dynamic diagnosis model. As we can see in Figure 14, a general diagnosis operational workflow consists of four steps: 1) Activate the Diagnosis Model, 2) Arrange the Diagnosis Engine, 3) Operate the Diagnosis Engine, and 4) Infer the Plant Damage States. DT-D class supports the sentiment that one should arrange and operate the diagnosis engine by providing suitable modules corresponding to the transient conditions. After a specific decision making, if the operational condition is changed, the diagnosis engine is rearranged and operated because the activated diagnosis model keeps working while operating the management and control system. Then, the diagnosis engine is used to identify the next step plant damage states which are reflected by the operator's decision-making.

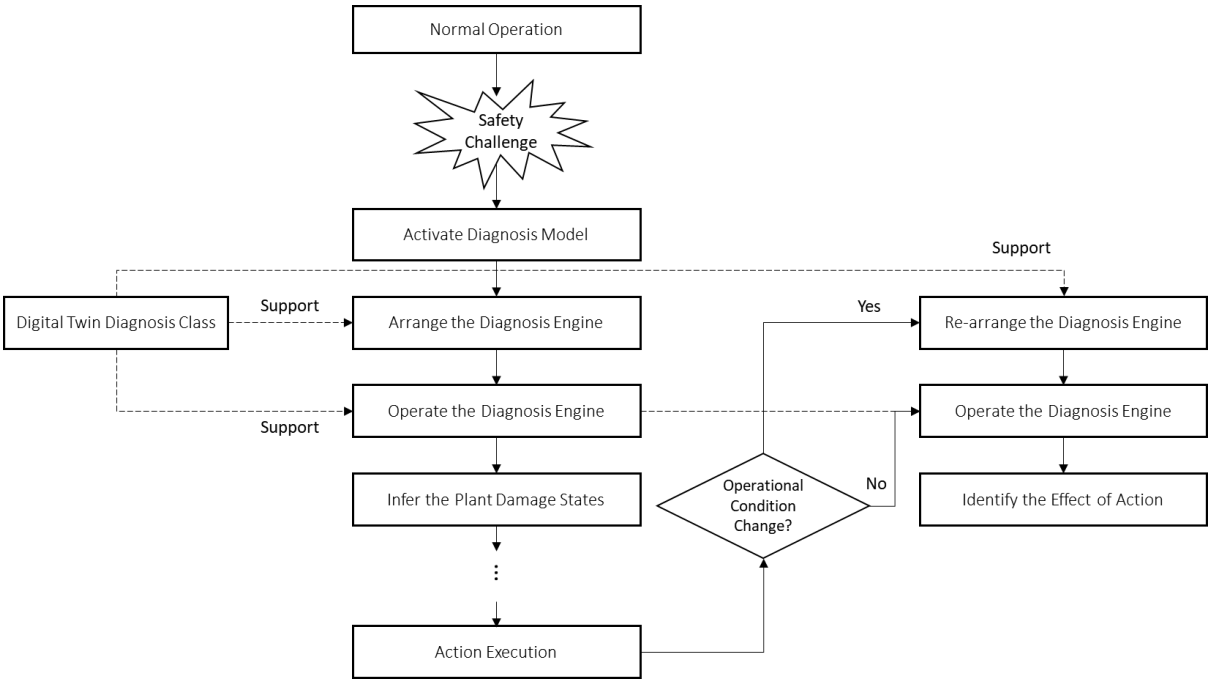


Figure 14. AI-Guided Diagnosis Workflow

Arranging the diagnosis engine determines that the modules in DT-D class would be appropriately selected by meaningful sensory data. To assemble the diagnosis engine, the diagnosis arrangement includes a filtering process that sorts out the correct signals in the incoming physical symptom measurement dataset. The workflow for extracting valuable signals among the whole signal set is based on the problem classification in table 3.

Figure 15 illustrates the steps to arrange the diagnosis engine by considering incoming signal dataset evaluation. From the actual plant, electrical sensory data travel to a diagnosis engine which is an initial version before diagnosis module combination. The sensory data from the incoming data collection is categorized into two groups, physical variable measurement signal, and component state signal, to function for different objectives in which one is selecting the SSF inference diagnosis module and the other is identifying SSC availability. The collected incoming physical variable dataset is divided into an input vector, which is composed of known and correct signal data, and an output vector, which is unknown or incorrect signal data, through two evaluation steps. The false sensory data detection model (FSDDM) aims to distinguish suspicious signals from the incoming signal set before diagnosis engine configuration. Incorrect measured values which are included in the input dataset cause the erroneous results from the diagnosis engine. Hence, the suspicious signals should be subtracted from incoming dataset before selecting

diagnosis modules to avoid error from sensor malfunction. The FSDDM, which works in the diagnosis engine arrangement process, is utilized to discern physically unreasonable signals from correct signals during the physical variables monitoring right after a safety challenge.

Figure 16 shows the information flow in these steps to describe how to extract the meaningful signals, present and correct physical symptom information, among the incoming dataset and interface with DT-D class in the operational workflow. An incoming dataset is divided into component signals and physical variable signals. The present and correct component signals are used for searching the related SSC information in the SIS. Meanwhile, by using the FSDDM, the physical variable signals are classified into missing signals, present but suspicious signals, and present and correct signals for selecting the diagnosis module in the DMS. After the classification of the physical variable dataset, the operator makes a decision on which variables would be the inputs and outputs of the AI-guided diagnosis system. The operator can choose the suspicious physical variables as the outputs if it is necessary, however, in the research, the output is assumed to the SSF. Through the processes, the appropriate diagnosis modules are assembled as core parts for the diagnosis engine in accordance with the signal evaluation result.

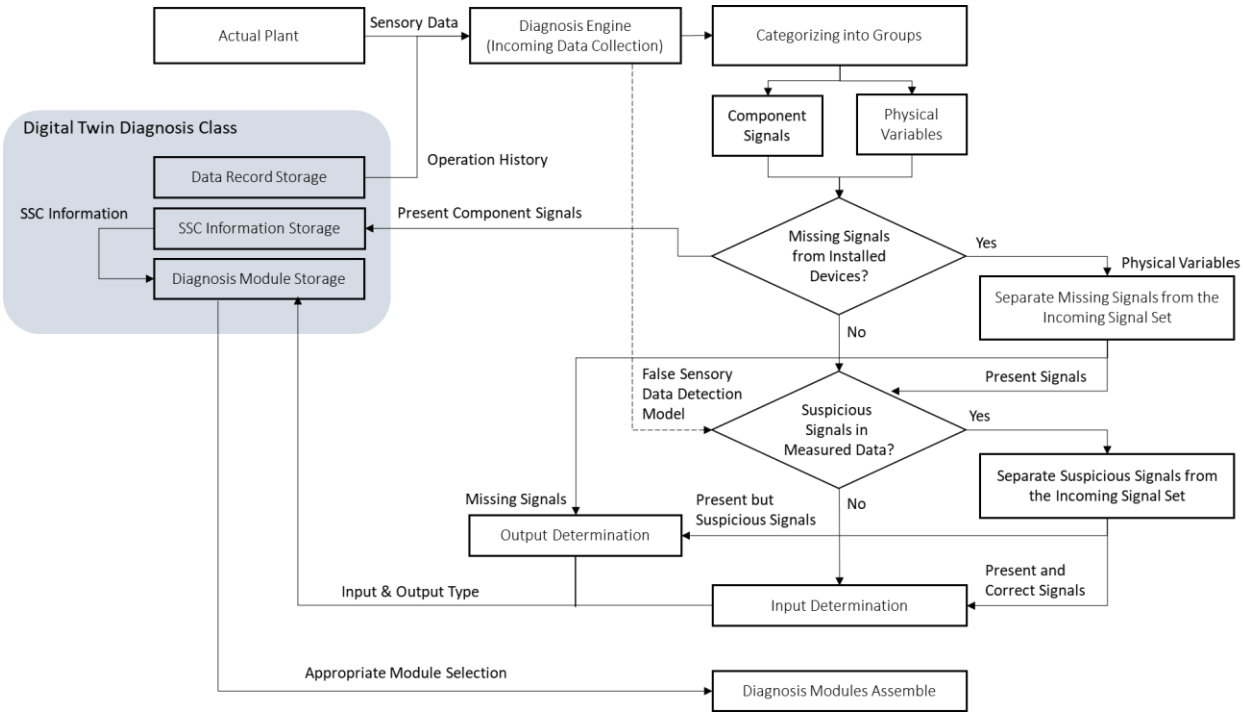


Figure 15. Diagnosis Engine Arrangement Workflow

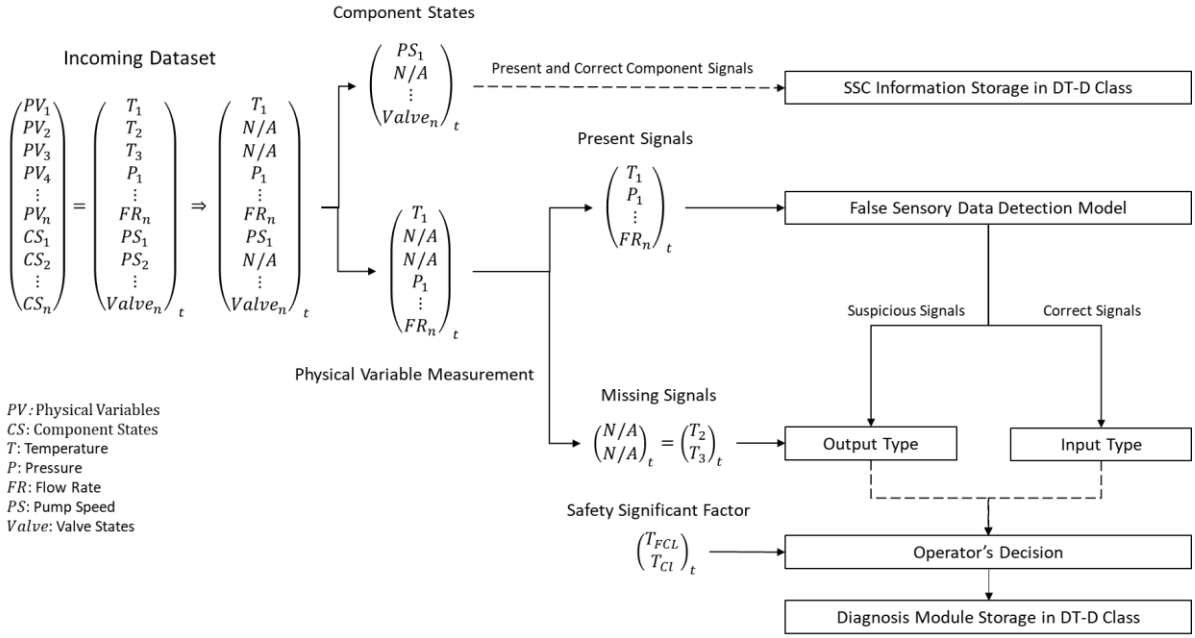


Figure 16. Information Flow to determine Inputs and Outputs for selecting Diagnosis Module

Diagnosis engine operation demonstrates the process to identify the plant damage states based on the signals and operation history to provide initial conditions of the prognosis stage. Physical damage states for the predictive model, dominant event and SSC availability for planning the action strategy are outcomes from the diagnosis engine for which diagnosis modules are assembled in the previous step. Figure 17 illustrates the AI-guided diagnosis engine operation flow that represents the interface between the actual plant and prognosis stage in the NAMAC system. The research focuses on developing the SSF inference model for the AI-guided diagnosis operation.

A detailed description of backgrounds for the NAMAC system can be found in Appendix B.

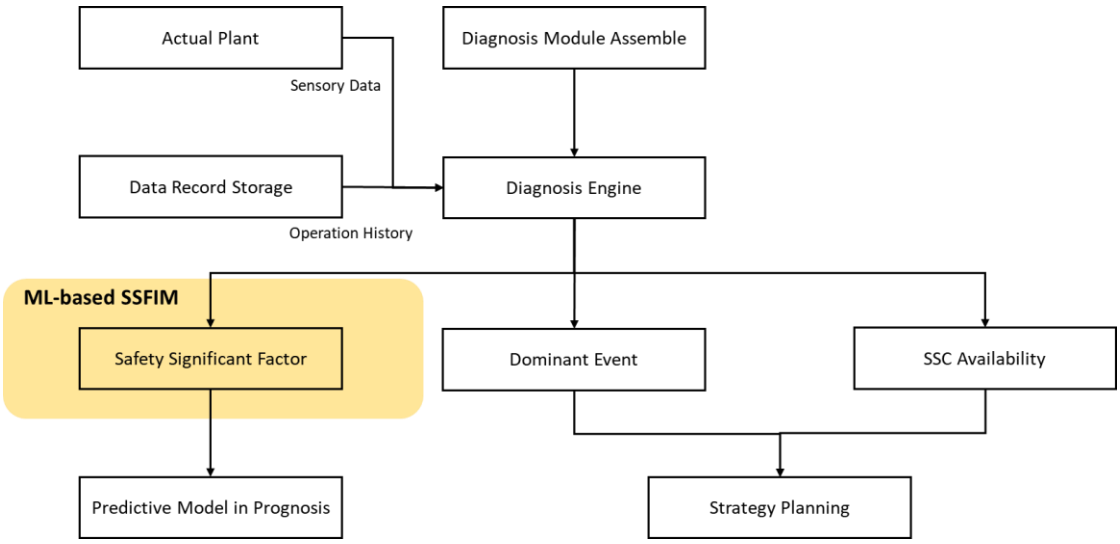


Figure 17. AI-guided Diagnosis Model Operation

2.3. Literature Review

Currently, diagnosis programs are developed by logical and existing knowledge. Accident Diagnostic, Analysis and Management (ADAM) system and Risk-informed Severe Accident Risk Diagnosis (RISARD) System have been developed for comprehensive accident analysis in nuclear power plants [24, 25]. The diagnostic module in both systems are established by a rule-based approach from the selected plant or code parameters that represent the plant symptoms. In the RISARD system, the accident scenario is analyzed based on the plant damage state sequences that are determined by occurrence frequency-based ranking. In other words, the most probable plant damage state is assumed for the next step of the scenario-based prognosis. The systems may be challenged by overlooked potentials that happen by less probable scenarios. However, the ML-based model is able to diagnose the plant states regardless of the frequency, unless less probable scenarios are omitted in the training dataset on purpose.

The ML is applied to develop advanced diagnosis tools to reduce human error. Supervised learning predicts a reasonable outcome about unknown situations by understanding the patterns from other similar situations. The ANN has been mainly utilized for “identification of the accident initiator and the plant state” because a severe accident includes rapid and complicated physical phenomena in a confined building [26]. For the training in the ML, deterministic computational simulators such as MELCOR, MAAP, or RELAP5 code produce training datasets. Symptom

Based Diagnostic System (SBDS) was developed by using the two hidden layers FNN to predict the accident features in a loss of coolant accident (LOCA) scenario [27 - 31]. Through the physical symptoms, which were simulated by RELAP5, event conditions including pipe break size, location, and emergency coolant cooling system (ECCS) states during the accident were identified. The SBDS is a symptom-based approach to achieve the event-based diagnosis. Additionally, to obtain a transients' type, severity, and location, two FNNs were combined by using the sum aggregation function. In case of the knowledge-based operator support system, Principal Component Analysis (PCA) was utilized to extract the features for the input vector in the FNN to identify the fault size and locations due to tube rupture [32]. The Dynamic Neural Network Aggregation (DNNA) model was able to evaluate the LOCA, Steam Generator Tube Rupture (SGTR), Steam Line Rupture (SLR), and Feedwater Line Break (FLB) scenarios. The RNN was utilized to classify transients for dynamic event recognition [33]. ALADDIN, which was developed by a combination of fuzzy clustering and the RNN, used a Wavelet On-Line Pre-processing (WOLP) to extract the features compactly by windowed wavelet decomposition due to long-term temporal dependencies of the time-series datasets [34]. To estimate the confidence level of the accident type, the RNN was combined with convolution neural network (CNN) in the novel fault diagnosis scheme [35]. The confidence level, which indicates the severity of the fault, was determined by the difference between outputs of the RNN and CNN.

The ML model is also applied to diagnose the accident conditions, such as a severe accident. In the smart support system, there are seven modules to diagnose DBA and a severe accident by using cascaded fuzzy neural network [36]. The smart support system estimated the LOCA break size and predicted the hydrogen concentration in the containment vessel with reactor vessel water level. The multi-layer CNN was used to identify the off-site doses within a certain distance from the plant for the evacuation plan [37]. These high techniques provide effective support to the operator for reliable diagnostic processes.

Since physical phenomena that influence the plant damage status are time-dependent, RNN is a more suitable method to use in the diagnosis process in comparison to FNN. The FNN is learning with the given data independently, which is, in other words, the sequence of data that is not considered in training, whereas RNN is updated not only by current input data but also by previous information.

In this research, the RNN is used for the ML-based SSFIM as a part of an AI-guided diagnosis model, which deals with the long-term time series data during a transient. In previous work of SBDS [30], input parameters, which are inlet temperature, outlet temperature, steam flow, etc., had a relatively uniform pattern or one-direction pattern over the entire transient period of 60 seconds. Because the assumption to diagnose the LOCA scenario was that 60 seconds are sufficient to capture the accident conditions, the FNN methodology could be applied in a short length of time. Also, to overcome the time-dependent problem, the 'aggregation sum function' considered the outputs of the FNNs of previous, current, and next time steps [31]. The methodology seems to be a valid approach during a short time window (several seconds). Even in the previous RNN model, a large amount of training datasets would be a major weakness to the developed model [32]. However, a transient case generally lasts for longer than several seconds. Also, physical symptoms in a transient would diversely change according to the scenario because the management and control system should keep operating the AI-guided diagnosis model to identify the effect of an operator's action. In virtue of the advanced modules, the SSFIM is able to handle a large amount of time-dependent dataset with a long-time window for a transient.

2.4. Summary

In this chapter, the AI-guided diagnosis is introduced in the aspect of a design and operational workflow. The AI-guided diagnosis aims to identify the dominant event, the SSC availability, and physical plant damage states during the entire plant states to provide useful information for the prognosis and strategy planning in the NAMAC system. The AI techniques are utilized to design the models and the digital twin is also used to support the diagnosis model. By using the concepts, the research work focuses on developing the ML-based model to infer the SSF in the following chapters.

CHAPTER 3. TECHNICAL COMPONENTS

In this section, several technical components are suggested with the developmental workflow to achieve the research objectives by confirming hypotheses. The technical components describe 1) what specific techniques are used based on the assumptions, 2) what challenges may be faced, and 3) how the problems can be solved to confirm the hypotheses.

3.1. Research Assumptions and Hypotheses

For the designing of the ML-based diagnosis model, assumptions provide foundations to establish the hypotheses. There is one assumption to reflect the philosophy of the AI-guided diagnosis and are three assumptions to cover the limitations in the development process.

3.1.1. Assumptions

i. Fuel centerline temperature and cladding temperature are the SSF.

The SSFIM aims to infer the unobservable variables from measurable physical variables. According to the data digitization report from Idaho National Lab (Yoon et al, 2014), tungsten/rhenium thermocouples are installed to measure FCL, coolant, and structural-material temperature including cladding temperature. In virtue of strength of the materials, the accuracy of device is within ± 1% at the anticipated FCL temperatures from 1700 ℉ ~ 2000 ℉ [33]. However, the limitation of the FCL thermocouples is 2400 ℉ (≈ 1588 K) at which the Inconel can be damaged [38]. Then, the core damage cannot be monitored. Even though the minor core damage may occur around 1250 ℃ (≈ 1523 K), the extent of the core damage state cannot be monitored because, as we can see in table 5, the core damage is observed around 1746 ℃ (≈ 2019 K) [39]. From this perspective, using the measured FCL temperature may be fair enough to identify the core condition in transient or DBA, but, in the BDBA sphere, inferred FCL temperature is necessary to identify the core damage states since the NAMAC system covers the full range of operation conditions. If the ML-based SSFIM is valid in transient cases, it would be applied to DBA or even the severe accident sphere. In BDBA or a severe accident, the FCL temperature should be inferred to identify how much the core has been damaged, which is never physically

estimated by the thermocouples. In conclusion, for the AI-guided diagnosis model, the FCL and CL temperatures are assumed to be SSF which represents the plant damage states.

Table 5. Reactor Conditions during Instantaneous Reactivity Insertion Accidents from Experimental Breeder Reactor II (EBR-II) Level 1 Probabilistic Risk Assessment (p. 313)' by D. J. Hill, 2018, Argonne National Laboratory

Inserted Reactivity ($)	Hot Pin Peak Fuel Temperature (°C)	Hot Pin Fuel Melting Time (sec)	Core Damage State
0.2	1124	-	No Core Damage
0.25	1189	-	No Core Damage
0.3	1250	38.7	Minor Core Damage
0.4	1367	10.01	Minor Core Damage
0.5	1486	3.64	Minor Core Damage
0.7	1746	0.508	Core Damage
0.85	1953	0.159	Core Damage
1.0	2167	0.108	Core Damage

ii. Deterministic code is a best-estimate code

The deterministic simulation results, which are used to train and test the ML algorithm, are assumed to be real physical phenomena in the reactor system. A large amount of data must be prepared to train and test the ML algorithm, but the real data amount is too little to train the model because of the rare occurrence of a nuclear accident. To compensate for the lack of actual data, data feed from the simulation code is the alternative. Therefore, the assumption that GOTHIC is the best-estimate code to generate the physical variable data in the reactor system eliminates the limitation of data sparsity. GOTHIC is an integrated thermal-hydraulics code for safety analysis from the Zachry Nuclear Engineering. It solves the conservation equations for mass, energy, and momentum based on the lumped parameter model to simulate various thermal-hydraulic phenomena in the reactor system. In order to support the assumption, a validation process between the computational simulation code and experiments is required. Through the benchmark study, GOTHIC code proves its capability by comparison between simulation results and experiment data of the Shutdown Heat Removal Test (SHRT) [40]. Reactor Analysis and Virtual control Environment (RAVEN) is a multi-purpose framework, which can be directly coupled with the deterministic code, to generate a large amount of data according to the head event, accident conditions, or operator's action strategy.

By solving the conservation equations for energy, mass, and momentum through the deterministic thermal hydraulic software package with the RAVEN framework, abnormal physical symptoms formulated with numerical values are used to train, validate, and test the ML-based SSFIM. The assumption plays the role of a bridge between a digital environment and the real world. According to the research assumptions above, the SSF is not measured by sensors in the core region and the numerical results from the deterministic code replace the actual physical variables in the reactor system. Therefore, in the research, the SSF is regarded as the FCL and CL temperatures from the GOTHIC simulation result.

iii. To focus on the SSF inference, other technical functions of the AI-guided diagnosis model are fulfilled.

The objective of the research is identification of the plant damage states through the SSF inference by the ML algorithm. In order to concentrate on the topic, other aims of the AI-guided diagnosis model, such as dominant event identification and SSC availability identification, are assumed to be known states. The technical components, which are also based on ML technologies, will be discussed in the further work part.

iv. Not only the incoming data but also short-term operation history in the DRS are intact and correct.

During the diagnosis system operation, the physical variable measurement signal set should continuously flow in the system at regular intervals. Specifically, not only one incoming dataset at every second but also thorough retention of short-term operation history in the DRS to convey the recorded data to the diagnosis model is required. In other words, the FSDDM does not need to be activated for the study. To satisfy the requirements, it is assumed that there is no data absence and no false data while the ML-based SSFIM is working.

3.1.2. Hypotheses

By confirmation of hypotheses, the objective of the research is achieved. There are three hypotheses to prove the capability of the ML-based SSFIM in a transient case.

i. The RNN would be an appropriate algorithm to deal with time dependent physical phenomena.

There are many neural network types in the ML algorithm: FNN, RNN, CNN, and so on. Because the RNN is known to treat the sequential data among the neural networks, adequacy of the utilization of RNN for the diagnosis would be confirmed.

ii. By using only a few measurable physical variables, SSF would be able to be inferred by the ML model.

GOTHIC code produces a lot of types of physical variable dataset. However, the SSFIM may not need too many types to figure out the FCL and CL temperature. To confirm the hypothesis, the research would try to answer the questions: how many physical variable measurements are required to infer the SSF? And what kind of physical variables are decisive inputs for the SSFIM?

iii. The well-trained model would show robust model performance even though there are some unstable measured signals in the input dataset.

The ultimate goal of the study is the development of a high-quality ML model to be well operated even in unstable states due to measurement noise in the incoming signal dataset. Therefore, the well -trained ML-based SSFIM would show acceptable model performance in testing process.

3.2. Developmental Workflow

The SSFIM aims to infer physical plant damage states from the incoming signals after a safety challenge. The ML-based diagnosis module developmental workflow consists of three parts: 1) data generation from deterministic code, 2) setting up the ML algorithm, and 3) training and testing the ML algorithm. A physical variable dataset used to train, validate, and test the developed model is produced with consideration of the reactor design, control action, and event mode through accident scenario assumptions. There are several indispensable components, including the appropriate ML algorithm type, model structure with advanced modules, and learning rate used to train the model, to compose an ML model. Finally, a well-trained diagnosis model, which satisfies the success criteria in the training and testing process, would achieve the research objectives. If the trained model performance does not satisfy the criteria, as we can see in figure 18, previous

steps in the workflow have to recur until the tested outputs converge on actual values. There is a potential problem in each step to obstruct the competent ML model development: 1) a data insufficient problem, 2) a model or algorithm problem, and 3) a training problem. These obstacles and solutions will be discussed in the next section in detail. If the model is able to infer the SSF level or missed information level in the test dataset, the completed ML-based diagnosis modules will be saved with a specific naming in the DMS of DT-D class.

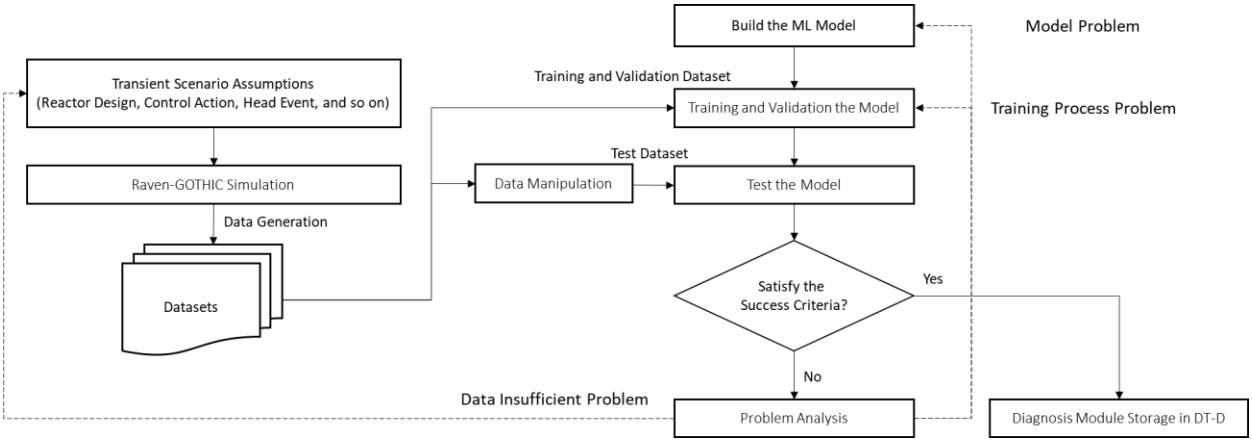


Figure 18. Safety Significant Factor Inference Model Developmental Workflow

3.3. Data Generation

As commented in the assumptions, the physical symptoms in multiple control volumes are expressed in numerical values by the thermal-hydraulic calculation in the GOTHIC. The simulation data is regarded as real data to train, validate, and test the ML model. In this section, there are several elements related to simulate the code for data generation: 1) design of the EBR-II model, 2) transient scenario with manipulated variables in the code, 3) physical symptoms from the system results, and 4) potential challenges with corresponding solutions for application to the ML-based diagnosis model development.

3.3.1. Simplified EBR-II Model

EBR-II is a pool type sodium-cooled fast reactor including primary-sodium system which surrounds the reactor in a primary tank, Intermediate Heat Exchanger (IHX), and secondary cooling system. The core heat is removed by the primary cooling system and transported to a steam

generator in the secondary cooling system through the IHX loop. The GOTHIC model uses both the sodium-side model and the water-side model to simulate the physical phenomena in the primary, intermediate, and secondary loops [41]. For the research, the EBR-II model, which was simplified by replacing the intermediate and secondary loops with boundary conditions from the full model steady state values is used [42]. Figure 19 depicts the schematic diagram of the EBR-II simplified model divided into the primary system and intermediate loop.

The simplified model contains 21 control volumes including a primary tank, primary sodium pumps and pump pipes, a high-pressure lower plenum, a low-pressure lower plenum, a reactor core, an upper plenum, an outlet pipe, and an IHX system. There are four channels in the reactor core region: active core assemblies, control rod assemblies, inner reflector assemblies, and outer blanket assemblies. Also, the IHX system comprises six control volumes: the primary shell, the secondary down-comer, the lower plenum, the upper plenum, the tube, and the outlet pipe to the secondary cooling system. In the GOTHIC, the thermal conductor model enables the code to calculate heat transfer between the surfaces of the core, liquid sodium, and other control volumes in the primary tank. Hydraulic connections between control volumes or control volume and boundary conditions are specified by 27 flow paths. Boundary conditions include IHX intermediate side temperature and flow rates and a shutdown cooling system. By using various packages, physical symptoms in each control volume are simulated in a numerical dataset.

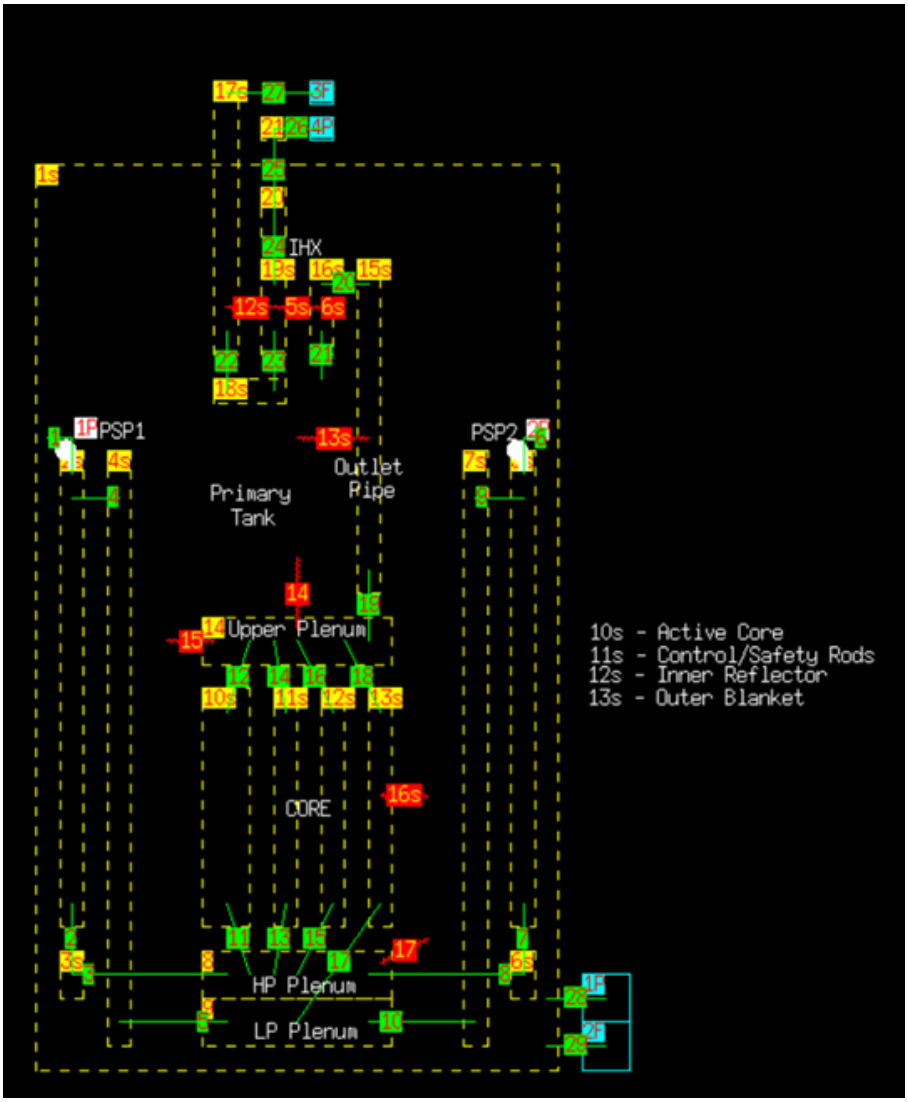


Figure 19. EBR-II Simplified Model in the GOTHIC Code

3.3.2. Transient

Transients due to two pump operation status are simulated to provide a database to confirm the hypotheses. Three types of transients are simulated to train the SSFIM and to manipulate the data to create the test dataset. Accident scenarios are as follows: 1) pump #1 has failed, but pump #2 does not have any action, 2) pump #1's speed has ramped down, and pump #2 has a set temperature point to speed up, and 3) pump #1's coast-down has occurred from 100% to 50%, and pump #2's speed has ramped down and up multiple times. The following sections go into detail about the transient cases.

- Transient Case A

LOF is one of the initiating events that may lead to BDBA, including core damage when the safety system fails. In the accident sequence analysis of the EBR-II PRA report, a transient associated with a single pump LOF owing to the electrical faults, such as clutch breaker trip, output breaker trip, and input breaker trip, is discussed. Also, mechanical faults, such as binding, seizure, and bearing failure, are described with the effect to pump speed reduction [39]. Figure 20 shows the coast-down curve by electrical faults in the Motor Generator (M/G) and pump shaft seizure which causes the most dramatic coast-down in pump speed. Depending on failure types, the pump speed curve form and dead time are varied.

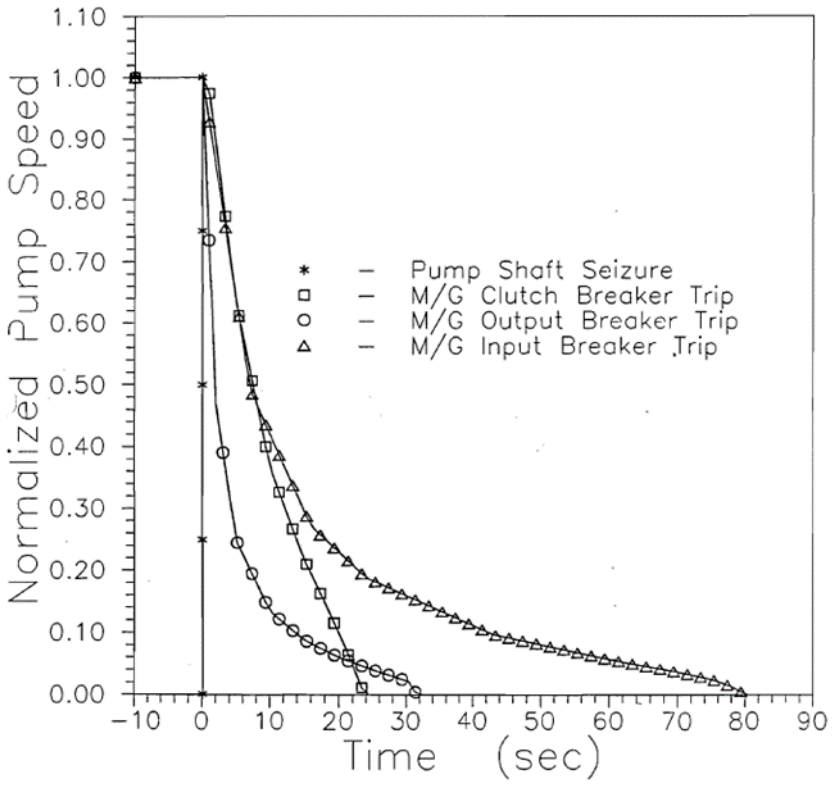


Figure 20. Pump Speed Coast-down Curve due to Pump Failure. Reprinted from 'Experimental Breeder Reactor II (EBR-II) Level 1 Probabilistic Risk Assessment (p. 142)' by D. J. Hill, 2018, Argonne National Laboratory

Transient case A is based on a single pump LOF in which pump #1's speed has ramped down, but pump #2 maintains a normal operation state. Final speed and the ramping-down period should be preselected to calculate the physical variable data in the computational code. The

ramping-down speed is linear with various slopes which are determined by coast-down time and final speed. The following equation shows the normalized pump #1 speed ($S(t)$) with time:

$$S(t) = \begin{cases} 1 - \left(\frac{1 - S_f}{t_c}\right) t & 0 \le t \le t_c \\ S_f & t_c \le t \end{cases} \quad (5)$$

S_f and t_c are the normalized final pump speed and coast-down time, respectively. In transient A, there are fifteen final pump speeds from 0% to 45%, and five coast-down times from 1 second to 65 seconds. Since the shortest time and the longest time, in which pump has been dead, are less than 1 second and about 80 seconds, respectively, in figure 21, the range of coast-down time in the GOTHIC model is assumed with consideration of the primary flow coast-down test results [39]. Figure 4 shows the pump speed coast-down curve of transient case A in GOTHIC simulation. Thus, in total, 75 episodes (15 final pump speeds × 5 coast-down time) are produced to train the ML algorithm.

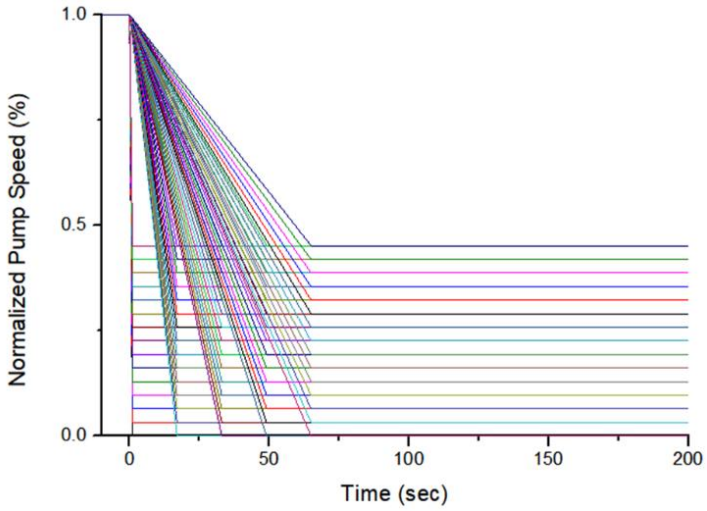


Figure 21. Pump Speed Coast-down Curve of Transient Case A in GOTHIC Code

- Transient Case B

Transient B has pump #1 coast-down, then the operator increases the pump #2 speed to maintain the core temperature. The simulation is to infer the safety significant factor level to identify the effect of the operator's action. Diagnosis is not over when the plant damage state associated with the safety challenge is identified, but it is continuously operated to examine the effect by a certain action for the system management. Pump #1's speed has been ramped down,

and the pump #2 starts to speed up when the fuel centerline (FCL) temperature reaches a set point. The linear ramping-up speeds are determined by the final speed and its starting time. In the operational workflow, it is assumed that the FCL and cladding (CL) temperatures are the SSF, but the calculated FCL temperature is used to generate the transient scenarios in the developmental workflow.

Figure 22 illustrates the pump operation status with time. The final pump #1 speed is 50%, whereas there are ten pump #2 ramping up speeds which vary from 105% to 150%. Also, seven FCL temperature set points are regularly distributed from 675 °C to 690 °C. Therefore, a total of 70 episodes' datasets (1 final pump #1 speeds × 10 pump #2 ramping up speed × 7 set points) are generated from the simulation code.

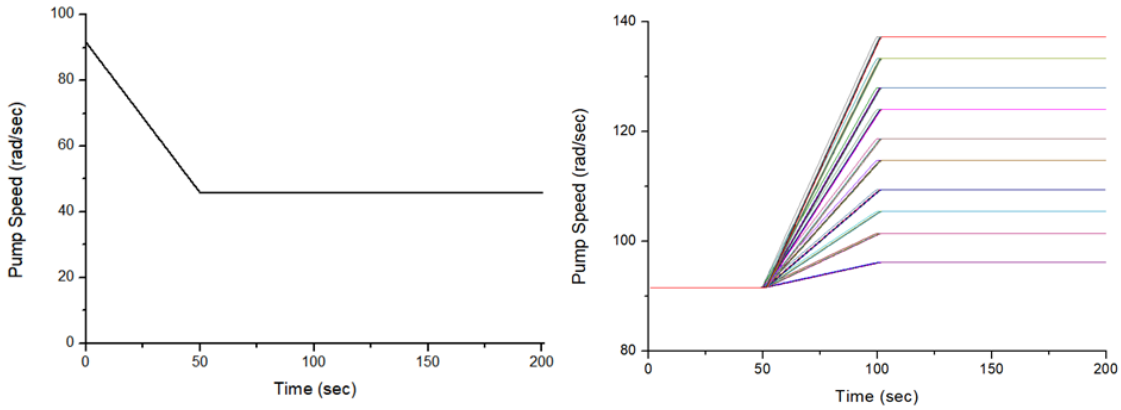


Figure 22. Pump #1 Speed (Left side) and Pump #2 Speed (Right side) in Transient Case B

- Transient Case C

Transient case C is based on an LOF with pump #2 action in which pump #1 and #2's speeds have ramped down, but the operator can control pump #2's speed. It is assumed that the control actions are carried out by the operator with his/her own judgment. The ramping-down and ramping-up speeds are linear with various slopes depending on the final speed and its reaching time. Pump #1's speed starts to ramp down from 100% to 50% speed at 20 sec during 50 sec. Figure 23 illustrates pump #1's speed over time. Meanwhile, pump #2's operation state is manipulated to simulate transient case C. Figure 24 shows the pump #2 states with 25 cases in which the speed dramatically changed from 114 sec to 115 sec (left side), and 25 cases in which the speed slowly changed from 114 sec to 199 sec (right side) after pump #2 speed's ramping-up.

Manipulated variables with their ranges are summed up in table 3-1. In total, 75 scenarios in transient A, 72 scenarios in transient B, and 50 scenarios in transient case C are simulated by the GOTHIC.

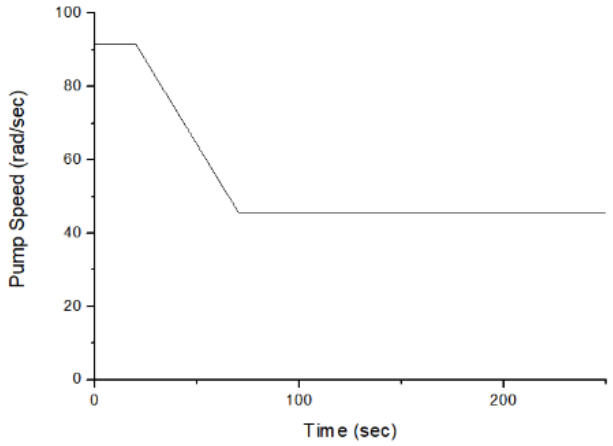


Figure 23. Pump #1 Speed in Transient C Case

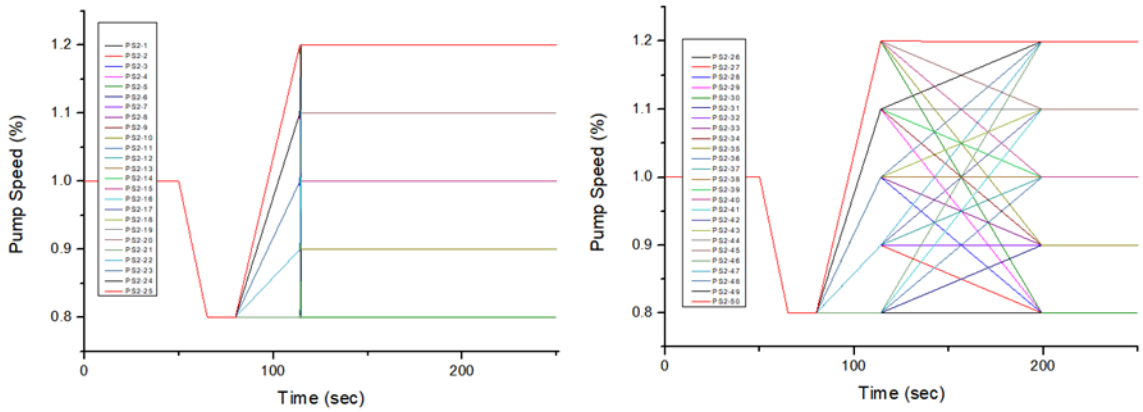


Figure 24. Pump #2 Operation Status in Transient C

To train, validate, and test the ML-based SSFIM, three cases are considered and simulated to generate the data. Transient A and transient C are the simplest and the most complicated cases among assumed conditions, respectively. The complexity of the scenario is determined by how many events are manipulated in one episode. In table 6, scenarios with manipulated variable descriptions are organized to compare features among transient cases A, B, and C.

Table 6. Description of Transient A, B, and C

Transient Case	Transient A	Transient B	Transient C
Number of Head Event	1	2	4
Scenario Description	Pump #1 Coast-down Pump #2 Constant [100%]	Pump #1 Coast-down Pump #2 Speed Ramping up	Pump #1 Coast-down Pump #2 Speed Ramping up and down
Manipulated Variables	Pump #1 Speed [0% ~ 45%]	Pump #1 Speed [50%] Pump #2 Speed [105% ~ 150%]	Pump #1 Coast-down [50%] Pump #2 Speed Ramping up and down 1st action : [50%] 2nd action: [80% ~ 120%] 3rd action: [80% ~ 120%]
Simulation Time	200 sec	200 sec	250 sec
Number of Scenarios	75	70	50

3.3.3. Physical Symptoms in the EBR-II Model

Abnormal pump operation conditions induce changes of physical phenomena deviated from the normal states in the reactor system. The physical variables are measured by installed devices in the system to send the observed information to the diagnosis system. In this section, physical symptoms are shown of transient A #53, transient B #34, and transient C #21. Figure 25 illustrates the pump operation status of both cases.

- ✓ In transient A-53, pump #1 has coast-down to 22% at 50 seconds, while pump #2 keeps the speed constant (left-side).
- ✓ In transient B-34, pump #1 has coast-down to 50% at 50 seconds, then pump #2 starts to increase the speed to 125% when the FCL temperature reaches 685 °C (middle).
- ✓ In transient C-21, pump #1 has coast-down to 50% from 20 seconds to 70 seconds, and pump #2's speed also has ramped down to 80% from 50 second to 65 seconds. So, the operator starts to ramp up the speed to 120% from 80 seconds to 114 seconds. But, after ramping-up, the pump speed decreases again to 80% until 199 seconds.

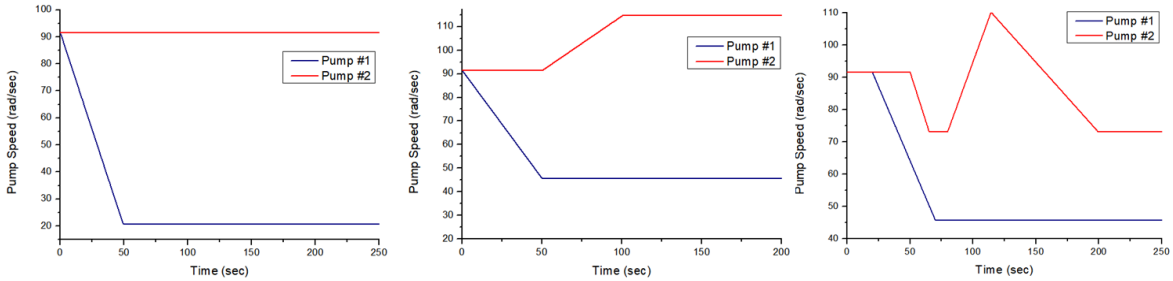


Figure 25. Pump Operation status in Transient A Scenario #53 (left), Transient B Scenario #34 (middle), and Transient C Scenario #21 (right)

Abnormal physical phenomena in transient A are illustrated in figure 26.

- ✓ The FCL and CL temperatures increase (top left).
- ✓ Liquid sodium temperatures in the upper plenum (UP) and outlet pipe rise as well as the core region temperature (top right).
- ✓ Since mass flow rate is determined by the pump speed, pump #1 has a negative mass flow rate in which a reverse flow appears due to pump #1's coast-down. However, core inlet mass flow rate into the active core channel converges to 150 kg/sec, because pump #2's mass flow increases despite the constant speed (bottom left)
- ✓ Pressure in the High Pressure Lower Plenum (HPLP) is dominantly affected by pressure pump #1's mass flow rate

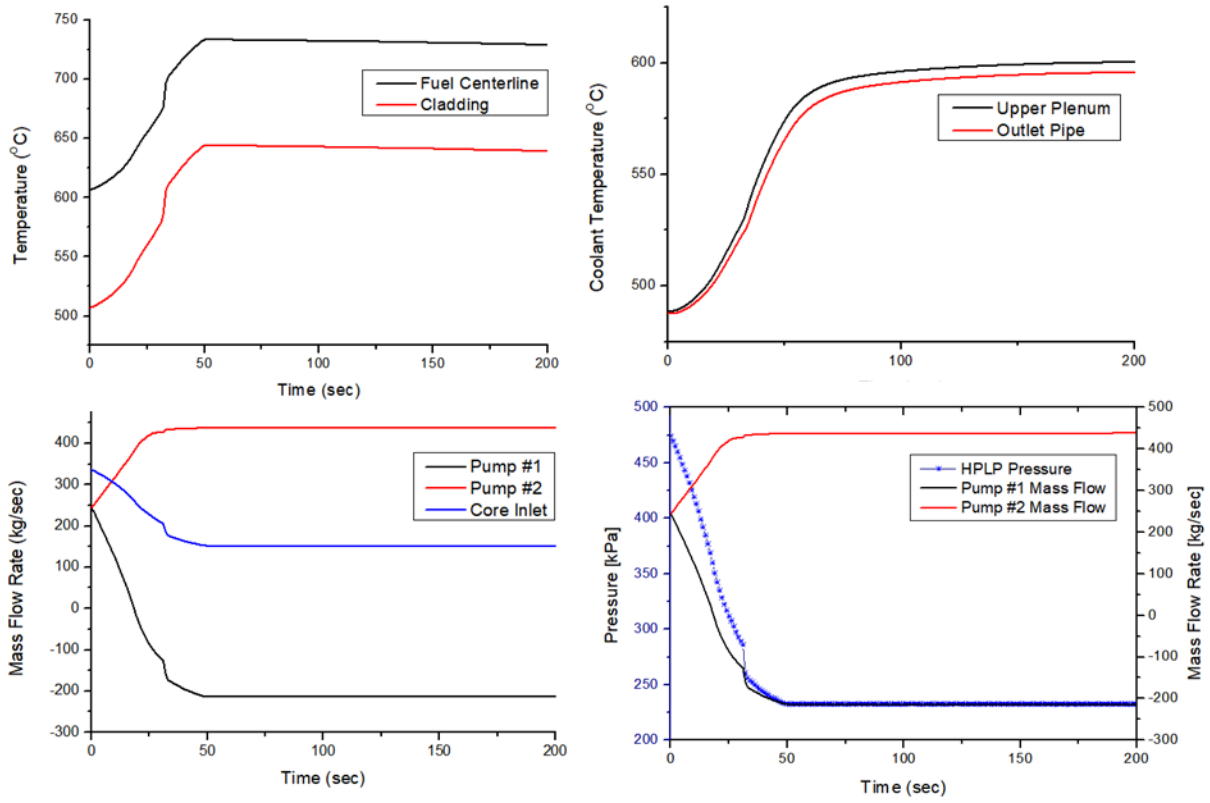


Figure 26. Physical Phenomena in the Reactor System in Transient A (Scenario #53)

Similarly, physical symptoms in transient B are illustrated in figure 27.

- ✓ Increased FCL and CL temperature decrease in virtue of pump #2's speed ramping up (top left).
- ✓ Increased coolant temperatures in the UP and outlet pipe decrease like core region temperature (top-right).
- ✓ Mass flow rates in transient B are similar to the transient A scenario, but the final core inlet mass flow is maintained at 210 kg/sec, which is a larger amount than in transient case A (bottom left).
- ✓ HPLP pressure is influenced by pump #1's mass flow rate, but it is risen slightly because of pump #2's speed ramping up (bottom right).

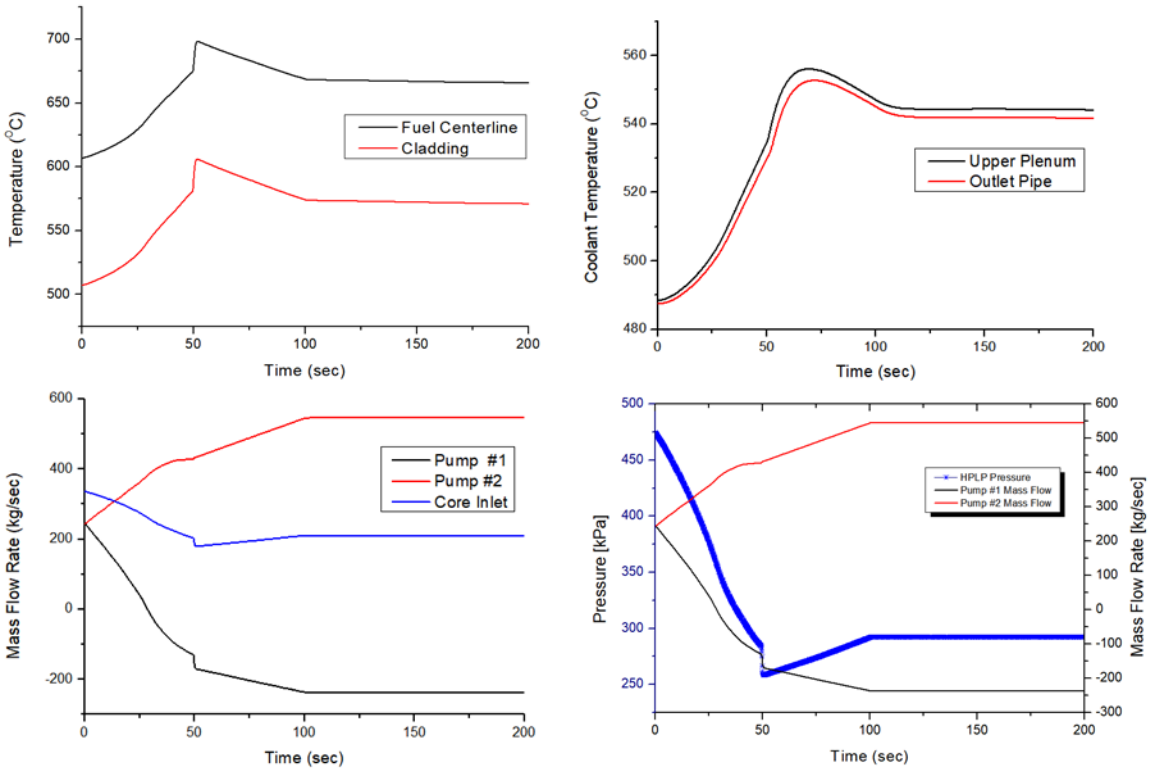


Figure 27. Physical Phenomena in the Reactor System in Transient B (Scenario #34)

Transient C, which includes multiple actions, shows multiple patterns in the physical phenomena in figure 28.

- ✓ FCL and CL temperature repeatedly increase and decrease caused by pump #1 and #2's speeds (top left).
- ✓ Liquid sodium's temperature also repeatedly increases and decreases (top right).
- ✓ There is still a reverse flow in pump #1, the amount of mass flow is changed corresponding to pump #2's operation status. The magnitude of a reverse flow gets larger when pump #2's speed ramps up, because the increased coolant is discharged into not only the active core assemblies but also the primary tank through pump #1's pipe. After all, the core inlet flow rate converges to 180 kg/sec (bottom left).
- ✓ Pump #1's mass flow rate is dominant in pressure in HPLP, as expected, but the pressure repeatedly increases and decreases owing to mass flow rate in pump #2 (bottom right).

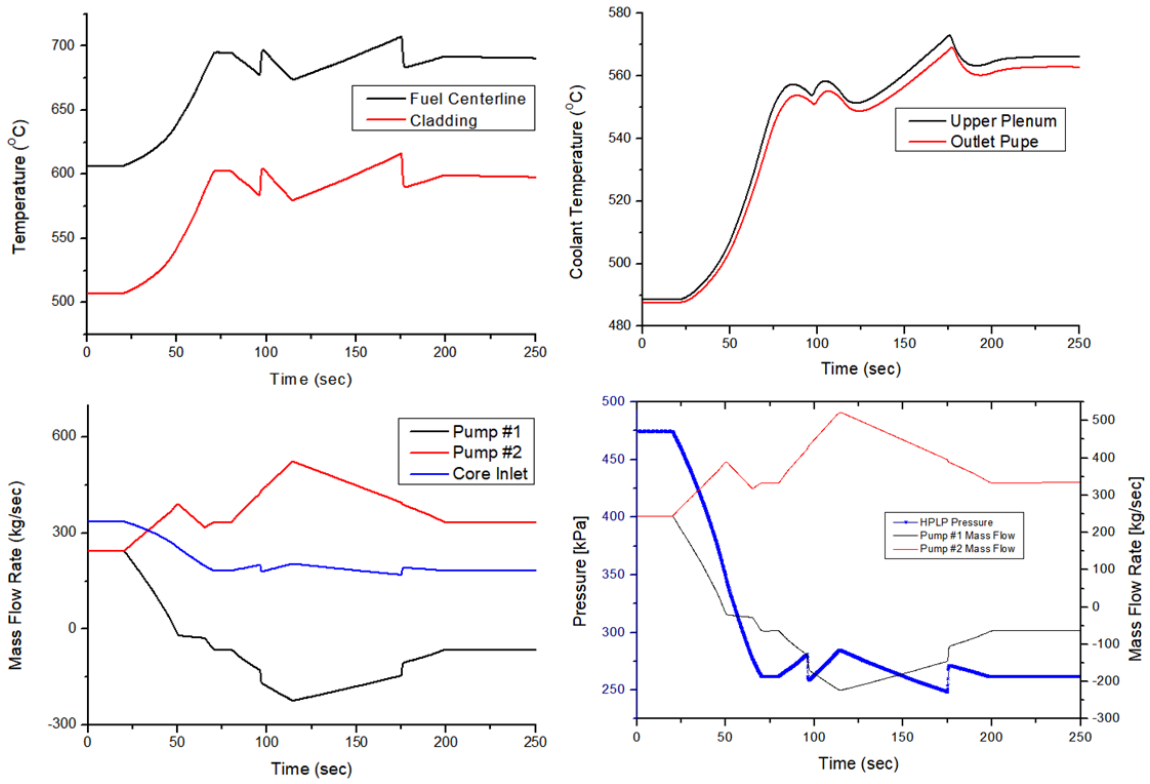


Figure 28. Physical Phenomena in the Reactor System in Transient C (Scenario #21)

Physical symptoms in the reactor system are changeable depending on the pump operation controls. The complicated physical phenomena in the reactor by multiple actions in the operation strategy cause difficulty in plant damage state identification. Each physical variable contains 15,000 data points $(= 1\left(\frac{datapoint}{sec}\right) \times 200(\frac{sec}{case}) \times 75(cases))$ in transient A, 14,000 data points $(= 1\left(\frac{datapoint}{sec}\right) \times 200(\frac{sec}{case}) \times 70(cases))$ in transient B, and 12,500 data points $(= 1\left(\frac{datapoint}{sec}\right) \times 250(\frac{sec}{case}) \times 50(cases))$ in transient C. The datasets are treated as measured values from the actual plant to train the ML algorithm.

3.3.4. Challenges

To develop the ML model, the representative issue of data related problems is data insufficiency. There are two questions to identify the issue coverage: 1) do the simulation cases cover all the feasible transient scenarios? and 2) do the assumed transient scenarios represent the AOO in EBR-II?

Sub-assumptions under the accident scope are established to answer the questions above. The accident scope is confined to the transient case to diagnose the plant damage states, since the AOO is the primary stage, which might be evolved to DBA or BDBA depending on the safety action strategy. Since the target operation status is the AOO, several specific and proper conditions are required to propose that the transient scenarios are used to represent the AOO in EBR-II. Four additional sub-assumptions are:

i. LOF represents the AOO in EBR-II,
ii. transient A covers the pump coast-down scenarios,
iii. transient B and C cover the pump coast-down with operator actions scenarios,
iv. and generated GOTHIC data is enough to cover the transient scenarios which are determined by pump operation status.

Then, sufficient physical variable datasets counted as real physical phenomena in AOO are prepared to train and test the ML-based SSFIM.

3.4. Machine Learning Model Components

In the research, the RNN is used to deal with time-dependent data to infer the sequential SSF. In this section, the RNN structure is described with challenges to develop the model.

3.4.1. RNN Structure Description

To learn the dynamic patterns from transient simulation data, the deep learning model, which is combined with advanced modules, is necessary. Figure 29 illustrates the RNN structure to train the ML model that is able to infer the SSF from measured physical variables. The following hyper parameters are used to build the network for training:

- ✓ Four-layer RNN with 30 neurons in each layer.
- ✓ Gated Recurrent Unit (GRU) is installed in each neuron to prevent a vanishing gradient issue.
- ✓ ReLU function is used as an activation function.
- ✓ Dropout method with 0.5 of dropout probability is used to prevent an over-fitting problem.
- ✓ Sequence length is determined randomly among 5, 7, and 10 sequence sizes to recognize a pattern better.

- ✓ *U, W,* and *V* are model parameters to be updated during training by backpropagation. The parameters are in the form of a matrix to match the dimensions between layers. For example, U is a $[m \times h]$ matrix, where m is an input dimension, which indicates input numbers, and h is a hidden size, which is the number of neurons in the hidden layer. W is a $[h \times h]$ matrix. And V, which is a $[h \times o]$ matrix, is called fully-connected layer parameter to match the dimensions between hidden layer and number of outputs.

A detailed description of the technical components in RNN structure can be found in Appendix A.

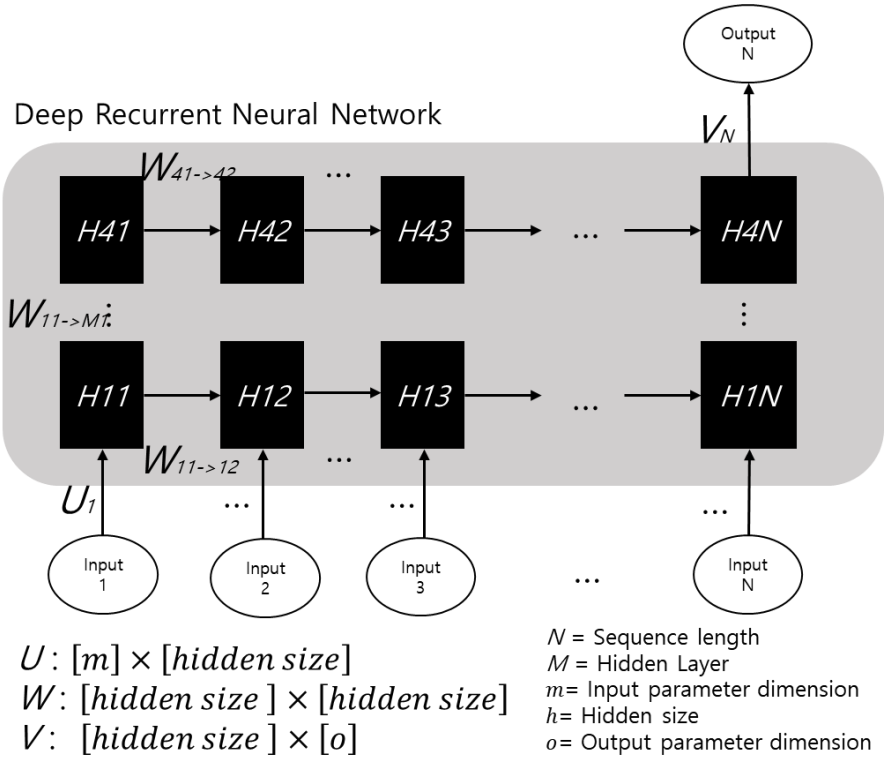


Figure 29. RNN Structure for the ML-Based Safety Significant Factor Inference Model

The Long Short-Term Memory (LSTM) and GRU modules are special units installed in the hidden layer for resolving the vanishing gradient problem, one of the major issues in a deep neural network. Since the GRU is a more simplified module than the LSTM, in virtue of fewer number of gate units, the GRU is used to build the RNN. A detailed description of the LSTM and GRU module can be found in Appendix A.

3.4.2. Challenge

The challenge in the model building process is an algorithm problem. If the algorithm or model structure is not appropriate for inferring the SSF, the algorithm or structure should be modified. The problem will be solved by hypothesis #1's confirmation.

3.5. Training and Testing Process

Through the training and testing processes, the RNN is trained and validated by using generated datasets, then the trained ML-based SSFIM is tested with a manipulated test dataset. The processes include physical variable data preparation, data preprocessing, and learning rate employment to support tuning the model parameter during training.

3.5.1. Physical Variable Data Preparation

The relationship between independent variables, also known as inputs, and dependent variables, used as outputs, is analyzed through the learning process. Therefore, in order to develop the ML model, the input and output features should be selected with discretion. A simplified EBR-II model contains 21 control volumes in which physical symptoms can be simulated. It would be ideal if all the physical phenomena can be monitored in every control volume, however, from the cost-effective perspective, installing devices all over the place is hardly practical. The main purpose of the research is SSF inference from a few of the chosen physical variables. The GOTHIC code calculates the liquid sodium temperature and pressure with time in each control volume, and the mass flow rate with time in each flow path. In the research, three physical variables are chosen as features in accordance with the following suggestions:

- Sodium coolant temperatures in the core region are not allowed to be used.

In the simplified EBR-II model, the core region consists of four assemblies' channels: an active core, a control rod, an inner reflector, and an outer blanket. The liquid sodium temperatures in the area are sensitive to change in the FCL and CL temperature since the fuel-related components are encompassed by coolant. Conversely, in the abnormal operational states or other accident conditions, the region turns into a vulnerable zone that needs to be monitored. Surely, the integrity of the devices should be maintained in DBA, but it is not guaranteed in BDBA. Therefore, even though the scope of the research is limited in the AOO, coolant temperatures in the core

region are not used to develop the ML-based SSFIM, which will be applied to other accident conditions.

- A pressure in the HP Plenum is selected.

Pressures are also able to be gauged in every control volume. Among them, pressures in the pump pipelines and LP show a wide variation compared to other control volumes because the change of pressure depends on mass flow rate. Therefore, HP plenum pressure is selected from the available data.

- Mass flow rates related to pump operation status are chosen.

Mass flow rate is a type of physical variable used to reflect the pump operation status. In the GOTHIC model, mass flow rate is calculated in the flow paths, referred to hydraulic connections between control volumes. There are three flow paths used to calculate the mass flow rates in the measurable physical dataset: 1) a path from the primary tank to pump #1's HP pipe, 2) a path from the primary tank to pump #2's HP pipe, and 3) a path from the HP plenum to the active core assemblies.

Table 7 shows chosen physical variables.

Table 7. Physical Variable Description

GOTHIC Label	Physical Variable	Location
TL1s45	Coolant Temperature	Primary Tank
TL2s2		Pump #1 Vertical HP Pipe
TL3s2		Pump #1 Horizontal HP Pipe
TL4s4		Pump #1 Vertical LP Pipe
TL5s2		Pump #2 Vertical HP Pipe
TL6s2		Pump #2 Horizontal HP Pipe
TL7s4		Pump #2 Vertical LP Pipe
TL8s1		HP Lower Plenum
TL9s1		LP Lower Plenum
TL14s1		Upper Plenum
TL15s3		Outlet Pipe
TL16s5		IHX Primary Shell
TL17s1		IHX Secondary Downcomer
TL18s2		IHX Secondary LP
TL19s5		IHX Secondary Tube
TL20s1		IHX Secondary UP
TL21s1		IHX Secondary Outlet Pipe
PR8s1	Pressure	HP Lower Plenum
FL1	Mass Flow Rate	Pump #1
FL6		Pump #2
FL11		Active Core Inlet

3.5.2. Data Processing for Training, Validation, and Test Dataset

For the ML model training and testing, data preprocessing by data rescaling, data splitting, and data manipulating is required. Data is rescaled by normalization, divided into three datasets for the cross validation by using a hold-out method, and manipulated with consideration of measurement error to test the trained model.

- Data Normalization

Simulation data generated from the GOTHIC are identical, numerical, and actual scale values. The raw data are barely suitable to train the ML algorithm, because the ranges and units

vary depending on the physical variables' types. In other words, since the incoming dataset from the reactor, such as temperatures, pressures, or mass flow rates, have different magnitude, units, and ranges, their influence on the updates in the ML algorithm would lean toward a certain physical variable. Training the neural network is a process to find the optimized model parameter that allows minimal MSE loss between inferred value and the actual value. If the input data have different scales, the information that have wide ranges in a large scale are dominantly reflected in the training process. Therefore, rescaling should be preceded before training the ML algorithm to improve the model stability.

The raw data is normalized by fixed upper and lower bound values, the empirical values from the simulation results. Generally, a min-max normalization is used as a feature scaling method for independent variables, however, the method is not useful, since the max values of SSF and physical symptoms are not known in abnormal operation status. Table 8 shows the assumed lower bounds and upper bounds of each physical variable.

- ✓ The melting point and boiling point of liquid sodium are 97.8 ℃ and 882.8 ℃, respectively. According to the transient simulation results, the lowest temperature is an initial temperature of 305 ℃ at IHX secondary down-comer, whereas, the highest temperature is 645 ℃ at the active core channel in transient A #1 scenario. The lower bound and upper bound are assumed to be 300 ℃ and 700 ℃, respectively.
- ✓ The maximum value of FCL and CL temperatures in the simulation are 743 ℃ and 654 ℃, respectively, whereas, in the research, the upper bounds of FLC and CL temperature are respectively assigned to 1000 ℃ and 700 ℃. The lower bounds of the FCL and CL temperatures are set to 600 ℃ and 500 ℃, respectively, to cover the initial values.
- ✓ There is a pressure measurement at the HPLP in the reactor. The bounds are 100 kPa and 500 kPa to cover the minimum value and maximum value from the simulation results.
- ✓ In the case of mass flow rates, reverse flow is shown at the flow path between the primary tank and pump #1 due to the coast-down. Hence, to make the normalized range [-1, 1] for mass flow rate in pump #1, upper bound and lower bound are set to 300 kg/sec and 0 kg/sec, respectively. On the contrary, since the coolant positively flows in pump #2 and the core inlet by virtue of ramping-up speed, the lower bound and upper bound are 100 kg/sec and 500 kg/sec, respectively.

Table 8. Upper Bound and Lower Bound for Data Rescaling

Physical Variables	GOTHIC Label	Lower Bound	Upper Bound
Liquid Temperature	TL1s45 ~ TL21s1	300 ℃	700 ℃
Fuel Centerline Temperature	TA21s1	600 ℃	1000 ℃
Cladding Temperature	TB21s11	500 ℃	700 ℃
Pressure	PR8s1	100 kPa	500 kPa
Flow	FL1	0 kg/sec	300 kg/sec
	FL6, FL11	100 kg/sec	500 kg/sec

- Training and validation dataset

On account of labeled datasets for training in supervised learning, the cross-validation method is utilized to conduct a performance test [43, 44]. The cross-validation is a method to solve the over-fitting problem. When the dataset is used to both train and test the model, the ML algorithm would implement the prediction by memorizing the data. Then there would be no significant difference between the testing accuracy and the training accuracy, however, the model would show poor model performance on new and unseen datasets. This is the over-fitting problem caused by re-substitution evaluation. A well-generalized model, also known as a well-trained model, reacts to new data with successful prediction by splitting the dataset into training data and test data. To develop the ML-based SSFIM, the dataset is divided into a training dataset for updating the model parameters, a validation dataset for choosing the best model in the training process, and a test dataset for evaluating the model performance. The approach prevents the over-fitting problem.

Cross-validation enables the objective function in the learning algorithm, such as the loss function, to be optimized in the training process and to be applied to evaluate the model performance in the testing process. The typical cross-validation methods are the hold-out method and the k-fold cross validation. In the hold-out method, the model is trained with a training dataset and improved by using the test dataset for selecting the best model (see figure 30). Although the test dataset does not derive the backpropagation to update the model parameter, the dataset becomes a training dataset because the test dataset is repeatedly re-used during epochs in the

training process. This is called “information leakage,” which causes the over-fitting problem. In order to not make any bias in the model performance, the test dataset must be used only one time. From this perspective, the hold-out method is appropriate for evaluating model performance rather than developing the ML model by the best model selection (see figure 31).

A major consideration is a split ratio of training dataset to test dataset. If the training set ratio is small, the model is hardly generalized because of data sensitivity, on the other hand, the over-fitting problem occurs on account of high variance in model performance when the test set ratio is small [45]. The general ratio of data is divided as 2/3 training dataset and 1/3 test dataset, though, the proportion of 70:30, 80:20, or others is determined by the user as occasion demands. The advantage of the hold-out method is a relatively fast learning speed, on the other hand, the weakness is that the data split ratio sensitively affects model performance when the amount of data is small.

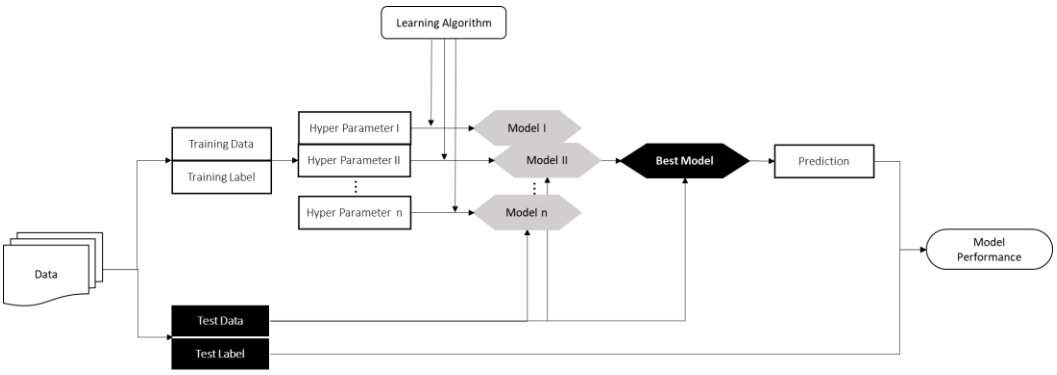


Figure 30. Hold-out Method Workflow for Performance Estimation by Model Selection

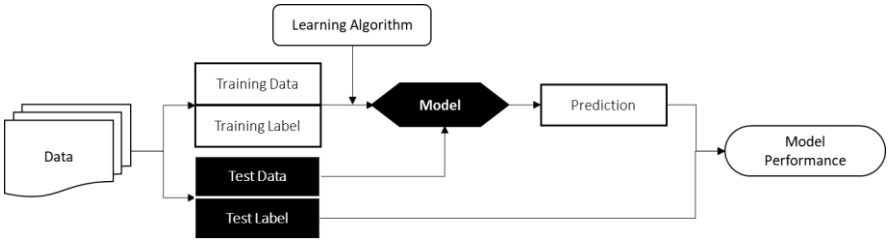


Figure 31. Hold-out Method Workflow for Performance Estimation without Model Selection

Another method is a k-fold cross validation in which the dataset is divided into *k* sub-groups to improve the accuracy under the condition of the small dataset. In other words, the

learning process by the hold-out method is repeated *k* times. Figure 32 illustrates how the method works in the training and testing processes. The following are the steps for the best ML model development by using the K-fold cross-validation method.

Step 1. A test dataset is decoupled from the whole dataset.

Step 2. The whole dataset except a test dataset is divided into *k* data folds that consist of *k-1* training data folds and one validation data fold.

Step 3. The learning algorithm by assigned hyperparameters is used to train and validate the model.

(a) *k-1* training data fold is used to train the ML algorithm.

(b) The other validation fold is used to estimate the model performance.

(c) Step (a) and (b) are repeated k times.

(d) Model performance is calculated to numerical mean value.

Step 4. Step 3 is repeated with various hyperparameter assignments.

Step 5. The best model is selected with the best model performance.

Step 6. Finally, the best model from step 5 is evaluated by the test dataset.

The method is useful in small datasets because the entire data is utilized for both training and testing. However, iterative learning brings about relatively long times needed to train the model.

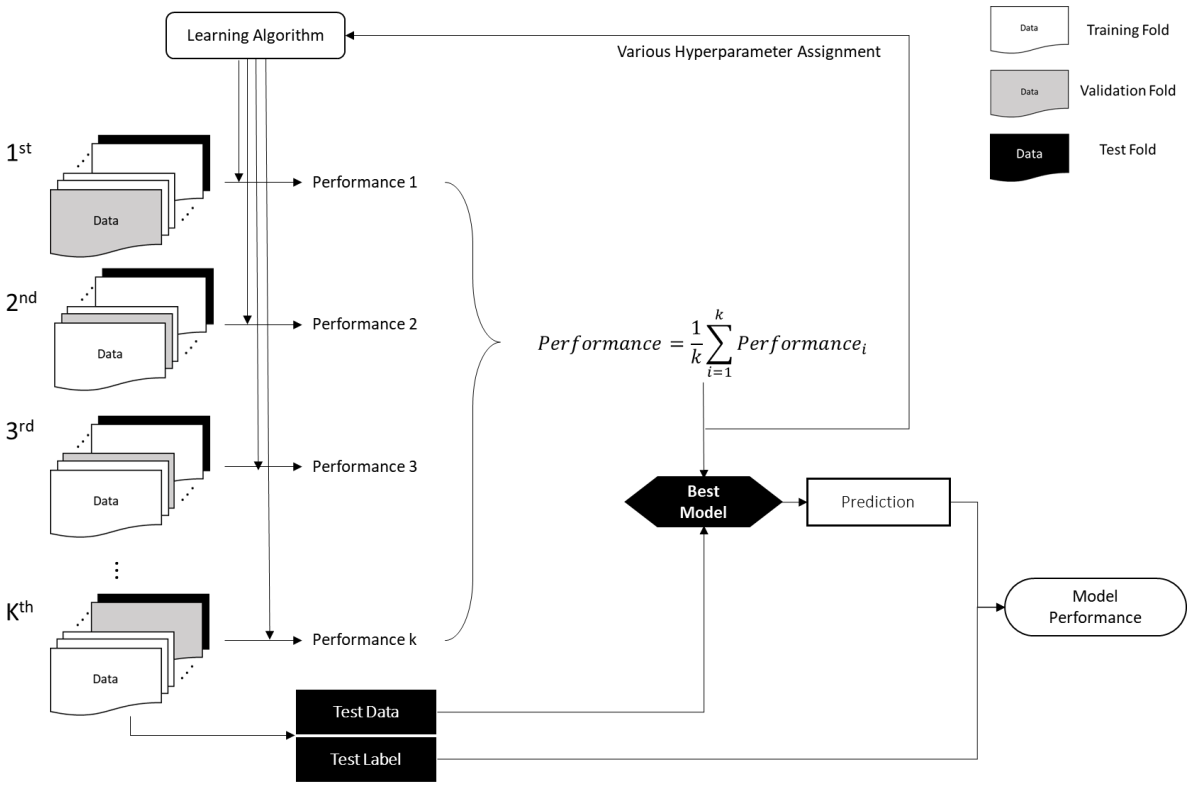


Figure 32. K-fold Cross Validation

To compensate for the shortcomings from both the hold-out method and the k-fold cross validation method, a three-way hold-out method is used to develop the ML-based SSFIM. As mentioned above, the hold-out method is a proper approach to estimate the performance rather than the best model selection. On the contrary, k-fold cross validation is ineffective from the learning time perspective despite usefulness for a small dataset. In the research, the three-way hold-out method, which compensates defects from both methods, is suggested. As we can see in figure 33, the three-way hold-out method is an improved variation on a basic hold-out method by adding the model selection process to develop the ML-based SSFIM composed of the best model. The datasets of the basic hold-out method are divided into two sections: a training set and a test set; meanwhile, the three-way hold-out method splits the dataset into three sections: a training set, a validation set, and a test set. The validation dataset is used to evaluate the models that are learned from the training set during the training process, then the test set is used to test the best model's performance. The learning process of the three-way hold-out method would be fast, because it is a kind of hold-out method. Additionally, since sub-assumption (iv) in Chapter 3.3.4. mentioned

that the amount of data is enough, k-fold cross validation does not need to be used. The application of the hold-out method to the research will be introduced in the training process section.

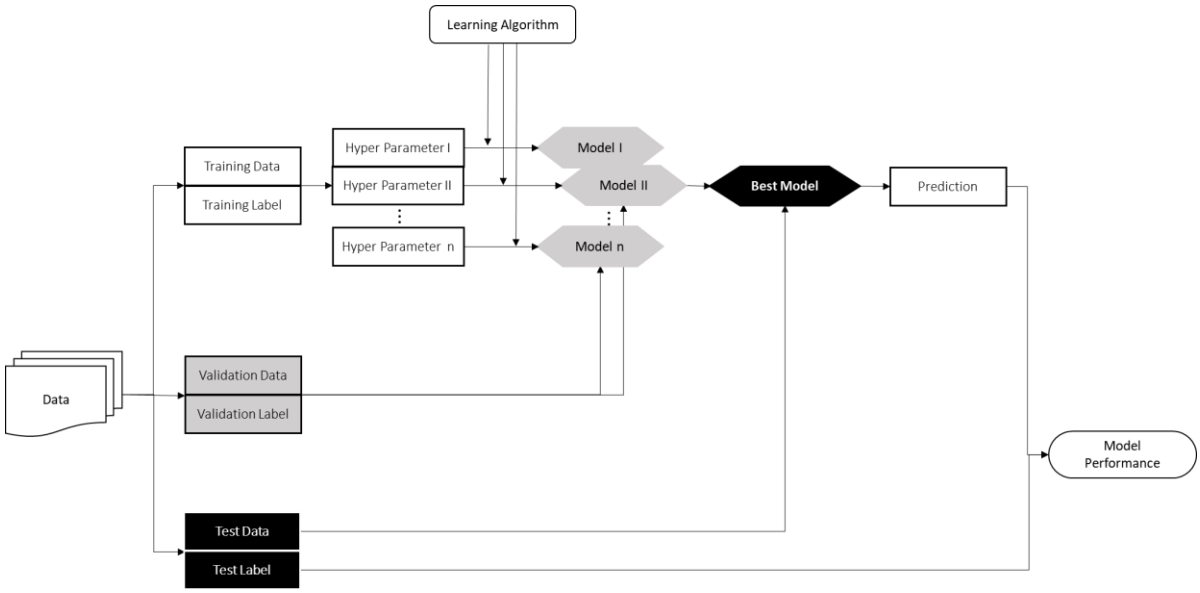


Figure 33. Three-way Hold-out Method

Tables 9, 10, and 11 describe the training, validation, and test data in transients A, B, and C, respectively. In table 12, a ratio of divided datasets is shown in which roughly 70% of the dataset is used for training, 25% is the validation dataset, and the rest is used as the test dataset.

Table 9. Training, Validation, and Test Dataset in Transient A

Number	Coast Down Time (sec)	Final Pump Speed (%)	Dataset	Number	Coast Down Time (sec)	Final Pump Speed (%)	Dataset
1	1	0	Training	39	33	25.8	Training
2	1	3.22	Training	40	33	29.03	Training
3	1	6.45	Training	41	33	32.25	Validation
4	1	9.67	Validation	42	33	35.48	Training
5	1	12.9	Training	43	33	38.71	Validation
6	1	16.13	Training	44	33	41.93	Training
7	1	19.35	Validation	45	33	45.16	Training
8	1	22.58	Test	46	49	0	Training
9	1	25.8	Training	47	49	3.22	Training
10	1	29.03	Training	48	49	6.45	Training
11	1	32.25	Validation	49	49	9.67	Validation
12	1	35.48	Training	50	49	12.9	Training
13	1	38.71	Validation	51	49	16.13	Training
14	1	41.93	Training	52	49	19.35	Validation
15	1	45.16	Training	53	49	22.58	Test
16	17	0	Training	54	49	25.8	Training
17	17	3.22	Training	55	49	29.03	Training
18	17	6.45	Training	56	49	32.25	Validation
19	17	9.67	Validation	57	49	35.48	Training
20	17	12.9	Training	58	49	38.71	Validation
21	17	16.13	Training	59	49	41.93	Training
22	17	19.35	Validation	60	49	45.16	Training
23	17	22.58	Test	61	65	0	Training
24	17	25.8	Training	62	65	3.22	Training
25	17	29.03	Training	63	65	6.45	Training
26	17	32.25	Validation	64	65	9.67	Validation
27	17	35.48	Training	65	65	12.9	Training
28	17	38.71	Validation	66	65	16.13	Training
29	17	41.93	Training	67	65	19.35	Validation
30	17	45.16	Training	68	65	22.58	Test
31	33	0	Training	69	65	25.8	Training
32	33	3.22	Training	70	65	29.03	Training
33	33	6.45	Training	71	65	32.25	Validation
34	33	9.67	Validation	72	65	35.48	Training
35	33	12.9	Training	73	65	38.71	Validation
36	33	16.13	Training	74	65	41.93	Training
37	33	19.35	Validation	75	65	45.16	Training
38	33	22.58	Test				

Table 10. Training, Validation, and Test Dataset in Transient B

Number	Set Point (°C)	Final Pump Seed (%)	Dataset	Number	Set Point (°C)	Final Pump Seed (%)	Dataset
1	672	105	Training	36	672	130	Training
2	675	105	Training	37	675	130	Training
3	678	105	Validation	38	678	130	Validation
4	681	105	Training	39	681	130	Training
5	684	105	Validation	40	684	130	Validation
6	687	105	Training	41	687	130	Training
7	690	105	Training	42	690	130	Training
8	672	110	Training	43	672	135	Training
9	675	110	Training	44	675	135	Training
10	678	110	Validation	45	678	135	Validation
11	681	110	Test	46	681	135	Training
12	684	110	Validation	47	684	135	Validation
13	687	110	Training	48	687	135	Training
14	690	110	Training	49	690	135	Training
15	672	115	Training	50	672	140	Training
16	675	115	Training	51	675	140	Training
17	678	115	Validation	52	678	140	Validation
18	681	115	Training	53	681	140	Test
19	684	115	Validation	54	684	140	Validation
20	687	115	Training	55	687	140	Training
21	690	115	Training	56	690	140	Training
22	672	120	Training	57	672	145	Training
23	675	120	Training	58	675	145	Training
24	678	120	Validation	59	678	145	Validation
25	681	120	Training	60	681	145	Training
26	684	120	Training	61	684	145	Validation
27	687	120	Training	62	687	145	Training
28	690	120	Training	63	690	145	Training
29	672	125	Training	64	672	150	Training
30	675	125	Training	65	675	150	Training
31	678	125	Validation	66	678	150	Validation
32	681	125	Test	67	681	150	Training
33	684	125	Validation	68	684	150	Validation
34	687	125	Training	69	687	150	Training
35	690	125	Training	70	690	150	Training

Table 11. Training, Validation, and Test Dataset in Transient C

2nd Action Pump Speed (%)	3rd Action Pump Speed (%)	Number	Dataset	Number	Dataset
			Reaching time at 199sec		Reaching time at 114sec
80	80	1	Training	26	Training
	90	2	Validation	27	Validation
	100	3	Training	28	Training
	110	4	Training	29	Training
	120	5	Training	30	Training
90	80	6	Training	31	Training
	90	7	Training	32	Training
	100	8	Validation	33	Validation
	110	9	Validation	34	Validation
	120	10	Training	35	Training
100	80	11	Training	36	Training
	90	12	Test	37	Test
	100	13	Training	38	Training
	110	14	Validation	39	Validation
	120	15	Training	40	Training
110	80	16	Training	41	Training
	90	17	Test	42	Test
	100	18	Training	43	Training
	110	19	Validation	44	Validation
	120	20	Training	45	Training
120	80	21	Training	46	Training
	90	22	Training	47	Training
	100	23	Validation	48	Validation
	110	24	Training	49	Training
	120	25	Training	50	Training

Table 12. Dataset Ratio in Transient Cases

	Training Dataset	Validation Dataset	Test Dataset	Total
Transient A	50 (66.67%)	20 (26.67%)	5 (6.66%)	75 (100%)
Transient B	49 (70%)	18 (25.72%)	3 (4.28%)	70 (100%)
Transient C	34 (68%)	12 (24%)	4 (8%)	50 (100%)

- Test Dataset

In the research, the testing process aims to not only identify the plant damage states by SSF inference but also the model's robustness through the model performance evaluation. To do so, the test dataset should be manipulated by detaching it from the whole dataset. As stated above in research assumption (ii) in Chapter 3.1.1., simulation data refers to measured physical variables from the installed devices during transient. Generally, there is a measurement error which consists of systematic error, also known as bias, and random error, also known as system noise, from the instrument [46]. The systematic error is caused by inadequate experiment design or a malfunctioning device, while the random error is a statistical fluctuation caused by a limitation of accuracy in device. The reason for the systematic error is findable and thus fixable, however the random error is unavoidable. Since the systematic error would be replicated unless the problem of the bias is solved, the error consistently happens in each measurement. The random error, which produces variability without any trends, does not affect the average. The more amount of repeated measurement, the less random error there is to be observed. Table 13 summarizes the characteristics of the systematic error and random error.

Table 13. Systematic Error and Random Error

Systematic Error	Bias - Caused by inadequate experiment design or malfunctioning device
	Consistent Error - Influences all the measurement in the same direction - Cannot be reduced by repeated measurements
Random Error	System Noise (Statistical Fluctuation) - Caused by limitation of accuracy in device - Has no pattern
	Unpredicted Error - Does not affect average due to the random direction - Can be reduced by repeated observations

The ML model, which is trained by the training dataset with the identical values without error from the best-estimate code, must figure out the reasonable SSF from the physical variables with the measurement error. Even though the AI-guided diagnosis model is developed by a

simulation-based data driven approach to overcome the limitations of actual experiments, the developed model should be applied to real operation status as a part of the management and control system. From this point of view, through the testing process, the robustness of the model should be proved by using data manipulation. In order to manipulate the simulation data for the test dataset, some sub-assumptions are necessary. The measurement error depends on what materials are used to manufacture the instruments, what physical variables are measured, and so on, since the noise occurs from the devices in the reactor system. The test datasets are fabricated based on the sub-assumptions:

(i) The simulation result is counted as a true value of each physical variable.

(ii) There is no systematic error from the installed device.

(iii) The measured value from the sensor accounts for the observed value which includes the systematic error and random error. The equation of observed value is

$$Observed\ Value = True\ Value + Bias + System\ Noise \quad (6)$$

where the true value is the simulation result from the GOTHIC. In the research, only the random error is considered for data manipulation to emulate the sensory signals from the reactor. Namely, there is no bias in measurement error.

(iv) The system noise is normally distributed within a certain percentage with random probability in every physical variable measurement.

In data, the noise is added on the simulation data for every second. The measured numerical values from the sensors is assumed by following equation:

$$PV_i(t) = PV_i(t) + N(0, \sigma \times PV_i(t)) \quad (7)$$

where $PV_i(t)$ is the numerical value of physical variable i at time t, and σ is a system noise percentage. The noise is a random number from normal distribution which consists of a mean value of zero and a standard deviation of error percentage.

Based on these assumptions, there are three types of test datasets: 1) a true value dataset by using simulation results only 2) true value with $\pm$1% random error in every measurement, and 3) true value with noise of ±0.375% random error in temperature, ±55 kPa pressure, and ±5% in mass flow rate. The magnitude of error in the last type of test datasets is consulted from the EBR-II Data Digitation that [33]:

- ✓ Thermocouples permit ± 0.375 % accuracy at coolant temperature in virtue of the materials.
- ✓ Fission-gas pressure transducer has a ± 55 kPa pressure accuracy over the 0 to 2.413 MPa.
- ✓ Flowmeter assures within $\pm 5\%$ accuracy of the sodium flow rate in measurement error.

Table 14 summarizes the test dataset types to identify the ML model robustness, $T(t)$ indicates a coolant temperature at time *t*, $P(t)$ accounts for the pressure at time *t*, and $FL(t)$ is the coolant mass flow rate at time *t*. Figure 34 illustrates the manipulated physical variable data based on transient B-32. As we can see below, the type 2 test dataset contains larger noise than type 3 in measuring the coolant temperature, but smaller noise in measuring the mass flow rate.

Table 14. Test Dataset Types for the Testing Process

	Random Error (σ)	Physical Variables
Type 1	N/A	Simulation Result
Type 2	$\pm 1\%$	$T_{type2}(t) = T(t) + N(0, 0.01 \times T(t))$ (8)
		$P_{type2}(t) = P(t) + N(0, 0.01 \times P(t))$ (9)
		$FL_{type2}(t) = FL(t) + N(0, 0.01 \times FL(t))$ (10)
Type 3	$\pm 0.375\%$	$T_{type3}(t) = T(t) + N(0, 0.00375 \times T(t))$ (11)
	± 55kPa	$P_{type3}(t) = P(t) + N(0, 55)$ (12)
	$\pm 5\%$	$FL_{type3}(t) = FL(t) + N(0, 0.05 \times FL(t))$ (13)

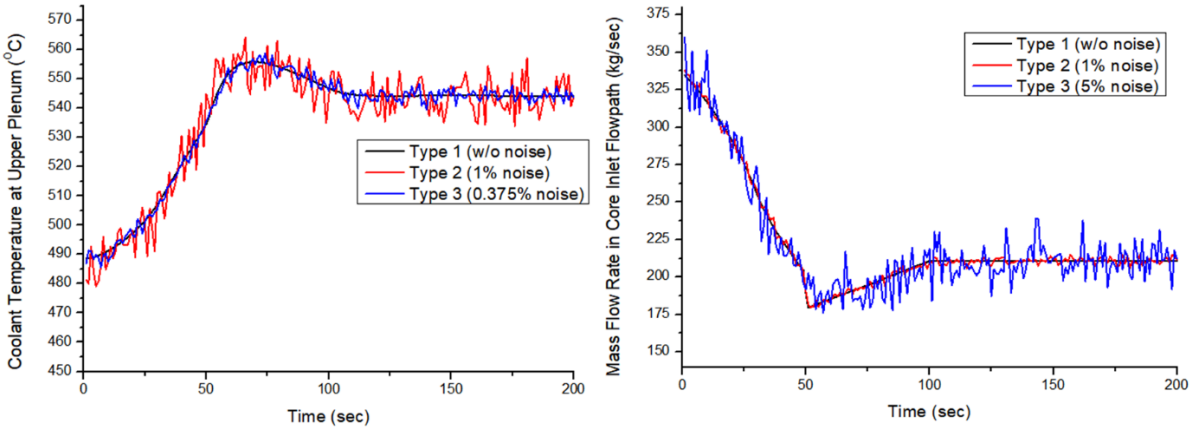


Figure 34. Upper Plenum Temperature (Left-side) and Mass Flow Rate in Core Inlet Pathway (Left-side) in Test Dataset B-32

3.5.3. Learning Rate Scheme

To train the model, one of the most influential hyper-parameters is the learning rate which affect updating the weight in the network in the training process. The weight (W) is assigned by a gradient descent algorithm given by

$$W \coloneqq W - \alpha \frac{\partial}{\partial W} J(W) \tag{14}$$

where α is a learning rate, and $J(W)$ refers to a objective function or cost function with the weight. The MSE loss function is used as the cost function in this research. As figure 35 illustrates below, a large learning rate causes the overshooting problem, in which the optimum weight is not found due to divergence, whereas updating the weight takes a very long time with a small learning rate. It should be properly adjusted to train the network. The learning rates are determined by a tuning strategy which is named the “learning rate scheme” or the “learning rate schedule”.

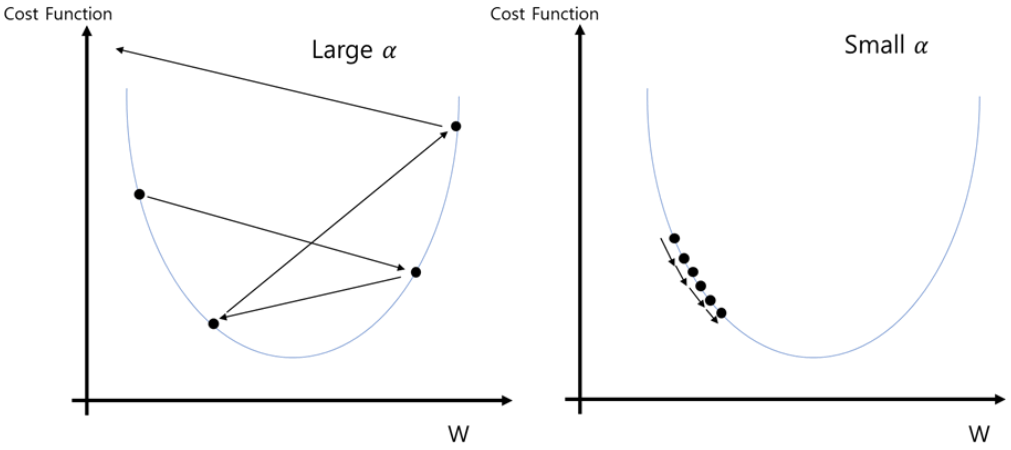


Figure 35. Learning Rate Impact in Cost Function

For the research, several learning rate schemes are introduced to train the RNN by tuning the learning rates in compliance with the user-defined rules in the schedule. Generally, in the hold-out method, the processes in which a fixed learning rate is used to train the model are repeated to find an appropriate hyper-parameter in the ML algorithm. However, the suggested learning rate schemes induce the learning rate changes within the hold-out method in the training process to curtail the learning time. For training the ML-based SSFIM, three types of learning rate schedules are utilized: 1) Cosine Annealing Warm Restart Learning Rates (CAWRLR) [47], 2) Cyclical Learning Rates (CLR) [48], and 3) Reduced Learning Rates on Plateau (RLRP). The following instruct how the learning rates are changed in each policy.

- Cosine Annealing Warm Restart Learning Rates

The CAWRLR are changed with periodical warm restart of the cosine function. The learning rate α_t at iteration t is

$$\alpha_t = \alpha_{min}^i + \frac{1}{2}\left(\alpha_{max}^i - \alpha_{min}^i\right)\left(1 + \cos\left(\frac{T_{cur}}{T_i}\pi\right)\right) \quad (15)$$

where α_{min}^i and α_{max}^i are the minimum and maximum learning rates for ranges in i^{th} iteration, respectively. T_{cur} stands for the number of epochs since the last warm restart and T_i is the number of epochs to restart. The T_i increases by a factor of T_{multi} after a restart. The learning rate schedule allows the CNN to have an acceptable model performance on a dataset of electroencephalogram (EEG) recordings and ImageNet [47]. Figure 36 illustrates the CAWRLR

activity when the $\alpha_{min}^{i} = 1 \times 10^{-9}$, $\alpha_{max}^{i} = 0.1$, and $T_{multi} = 2$ to extend the length of a cycle between warm restarts twice.

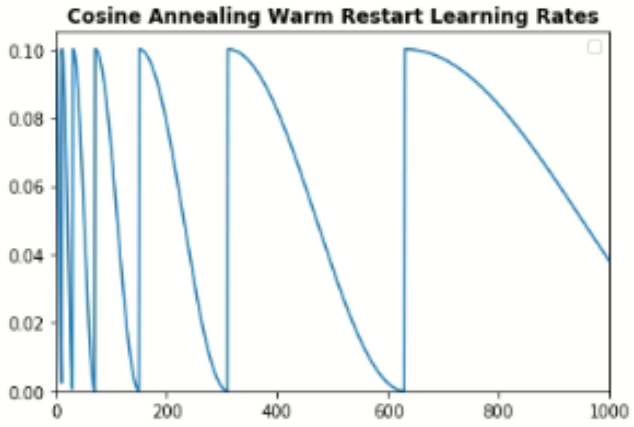


Figure 36. Cosine Annealing Warm Restart Learning Rate Scheme

- Cyclical Learning Rates

The learning rates in the CLR repeatedly increase and decrease between the upper bound and lower bound. The CLR policies depends on how the learning rates are going up and down in the ranges or how the bounds are changed at the end of each cycle. A "*triangular2"* mode, which has the repeated linear rising and falling shapes and dwindles the bounds to half after each cycle, is proposed among them. To calculate the learning rate n_t, the equation is

$$\alpha_t = \alpha_{min} + \left(\frac{1}{2}\right)^{\left\lfloor \frac{1+i}{2T} \right\rfloor)} (\alpha_{max} - \alpha_{min})(\max(0,1-k)) \tag{16}$$

$$k = 1 - \left| \frac{i}{T} - 2\left\lfloor \frac{1+i}{2T} \right\rfloor + 1 \right| \tag{17}$$

where α_{min} and α_{max} are the minimum and maximum learning rate. i and T are iteration and step sizeyes to complete half of a cycle, respectively. Also, a floor function $\lfloor x \rfloor$ is

$$\lfloor x \rfloor = \max\{n \in \mathbb{Z} : n \leq x\} \tag{18}$$

where x is a real number and $\mathbb{Z}$ is an integer. By using the CLR policy, the model accuracy is almost as high as if it were using the fixed LR policy, but it takes a smaller iteration numbers in training the CNN on CIFAR-10 and Imagenet dataset [48]. Figure 3-20 shows the CLR with $\alpha_{max} = 1.0$, $\alpha_{min} = 1 \times 10^{-9}$, and $T = 200$.

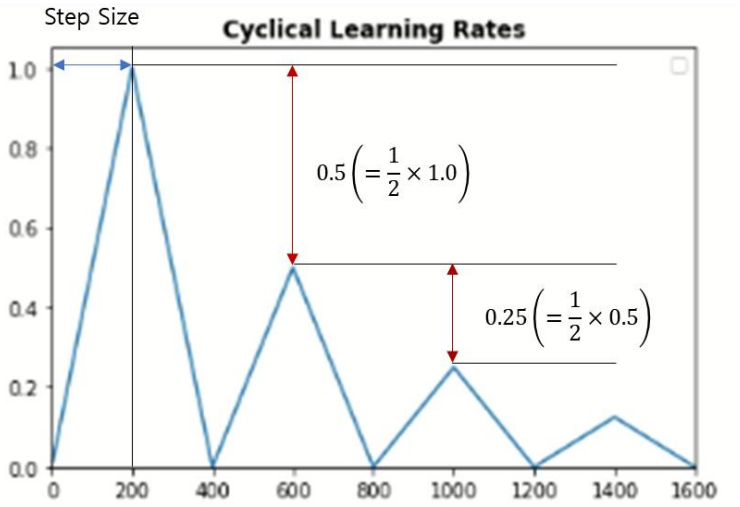


Figure 37. Cyclical Learning Rates Policy

- Reduced Learning Rates on Plateau

The RLRP reduces the learning rates in the stage when a training metric, such as the MSE loss, does not decline during a recent period. The learning rate α_t is calculated by multiplication of the previous learning rate α_{t-1} and discounting factor γ.

$$\alpha_t = \alpha_{t-1} \times \gamma \tag{19}$$

The following steps and figure 38 demonstrate the RLRP strategy in this research to train the ML-based SSFIM:

Step 1. Start the training process by setting up the initial learning rate (α_i), which is 0.01 in the research.

Step 2. Train and validate the model.

Step 3. Pay attention to validation loss.

Step 3-1. If the validation loss decreases, update the best validation loss with the new one and go to step 2 again. Or, go to step 3-2.

Step 3-2. If the validation loss does not decrease during the last k_t epochs, go to step 5. Or, go to step 3-3. The k_t is a training termination inducing number. In this research, k_t is set to 200.

Step 3-3. If the validation loss ever decreases during the last k_u epochs, go to step

4-1. Or, go to step 4-2. The k_u is a patience number. In this research, k_u is set to 25.

Step 4. Adjust the learning rate.

Step 4-1. Go to step 2 with an existing learning rate.

Step 4-2. Go to step 2 with discounted learning rate ($\alpha_{t-1} \times \gamma$). In the research, a discounting factor is 0.5.

Step 5. Terminate the training process by saving the model that has the smallest validation loss as the best model. Details are described in the training and testing workflow section.

The RLRP has a distinctive characteristic in that the learning rate has interacted with the validation loss. Two former schemes change the learning rates regardless of the evaluation result in the process and terminate the training when the prearranged epochs are completed. In other words, without checking the progress of the training process, it could be completed without reaching either the global or local minimum, because there is a chance to keep on updating the model parameters. However, in the RLRP, when the validation loss is not improved for quite a while (200 epochs are assumed in the research), the algorithm stops the epochs based on the judgment that the minimum is found in the cost function, whether it is global or not. Because the consistently diminished learning rate leads to an immense amount of training time. Since the training continues 200 epochs without any improvement in the validation loss, the learning rate becomes 1/256 of it at the end. Therefore, a predetermined number of epochs impedes the proper training process when the model reaches convergence very fast or does not find the minimum within the epochs. From this perspective, the RLRP is a beneficial method because of the interaction between the model parameters and the validation result.

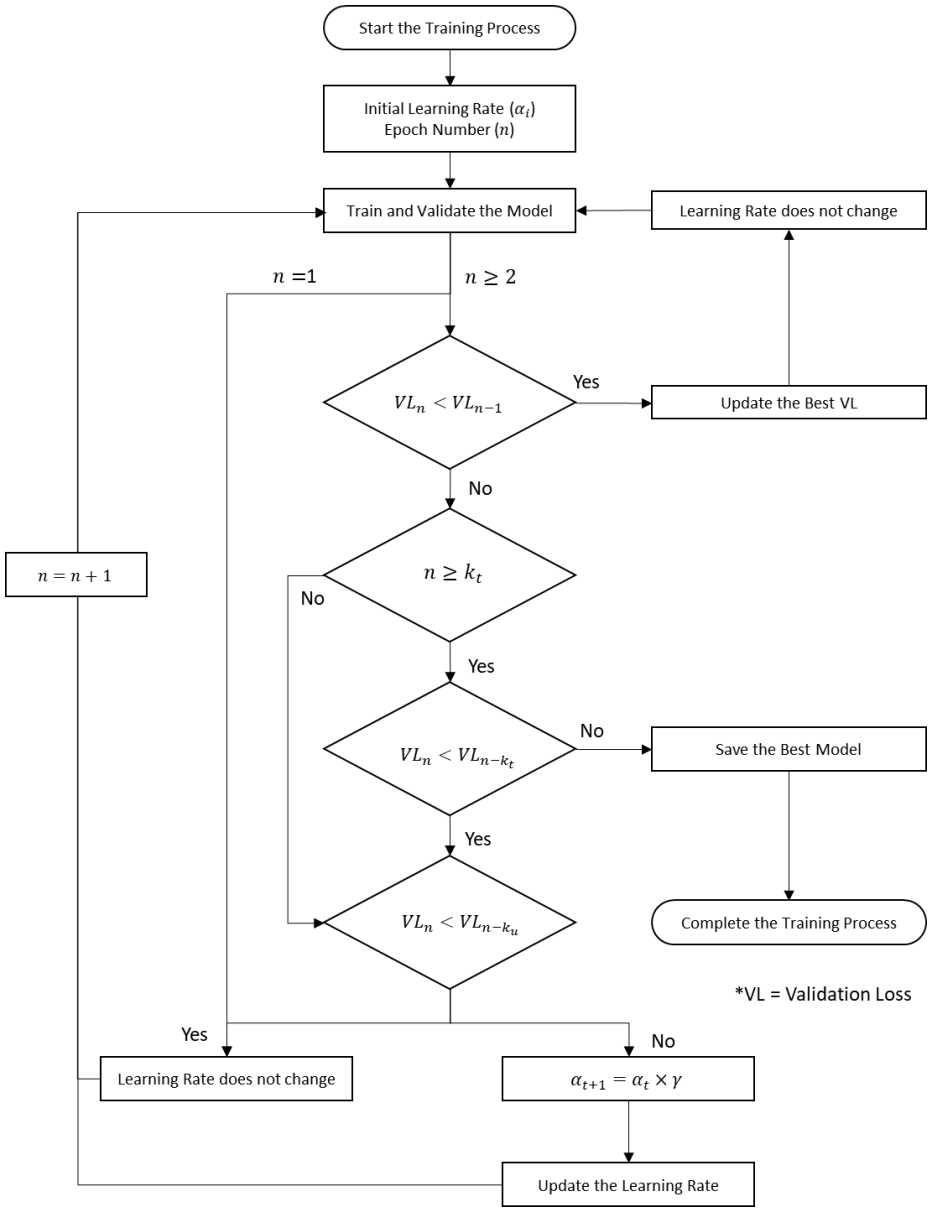


Figure 38. Reduced Learning Rates on Plateau

Through the case study, three options are used to identify which scheme is the most effective for training the RNN by comparison with each other. These learning rate schedules are widely used in the ML-field. The CAWRLR and the CLR policies are proven efficient approaches used to recognize the image data or analyze the EGG. Whereas, the RLRP theoretically facilitates the reasonable training process by reducing the learning rate from the interaction between the validation MSE loss and the rules. Based on the result, the most powerful scheme among the suggested ones is applied to subsequent case studies.

3.5.4. Training and Testing Workflow with Success Criteria

- Training and Testing Workflow

To develop the ML-based SSFIM for training and testing, an algorithm is built on the basis of the above-mentioned technical components. The training process includes not only updating the model parameters but also evaluating the model by using the validation loss. In the testing process, the final model performance is evaluated to identify the robustness to be used as a diagnosis model in the management and control system. The processes are implemented through the three-way hold-out method. Figure 39 demonstrates the steps of processes in which the training process and the testing process are left-side workflow and right-side workflow, respectively. Also, figure 40 illustrates how the three-way hold-out method is applied in the workflow.

Step 1. Learning Rate Scheme Selection

Building the algorithm is the first stage for the training process. The RNN with the three-way hold-out method is the key fixed constituent in the structure of the algorithm. The learning rates adjusted by schedule's rule are the independent hyper parameters needed to decide which scheme is efficient for training.

Step 2. Input Selection (Feature Selection)

Feature selection refers to choosing the useful input types to infer the output among the full feature set. Since the subset for the input type is arbitrarily selected, the subset should be re-selected when the model does not satisfy the success criteria in the evaluation either in training or in testing. The output type is the SSF.

Step 3. Training the Model with the Training Dataset

Step 4. Model Parameter Updates by Backpropagation

Step 5. Validating the Model with the Validation Dataset

The backpropagation for the model parameter is not implemented after a validation.

Step 6. Comparison of the Validation MSE Losses

In every epoch, the calculated validation loss is compared to the previous epoch's validation loss for singling out the best model.

Step 7. Saving the Best Model

The best model is updated by replacing the model which has the lower validation loss. After this step, training and validation are repeated until the last epoch (see figure 22).

Step 8. Repeat the Training Process until Training Termination

Step 9. Training Completion

The training process is finished when the pre-arranged epochs are completed.

Step 10. Training Process Assessment

The model performance is primarily estimated to examine whether the training process allows the model to be trained well to infer the SSF. The training success criteria, which indicates the acceptance criteria to satisfy the training process assessment, is a value of 0.09 of the best MSE validation loss [49].

Step 10-1. If the best MSE Loss is larger than 0.09, go to Step 1 or Step 2.

The mode which has the smallest validation loss in the training process indicates the most well-trained model under the given hyper parameter scheme and input and output configurations. If the model is not content with the training success criteria, there are some issues to be improved for the model in a learning rate scheme, input and output type selection, or both.

Step 10-2. If the best MSE Loss is less than 0.09, go to Step 10.

When the model passes the training evaluation, it is ready to be tested by using the manipulated dataset.

Step 11. Model Test with the Test Dataset

A Test dataset consists of three types: 1) simulation data without noise, 2) a same random noise (1%) at every physical variable, and 3) different random noises in each physical variable.

Step 12. Final Model Performance Assessment

To be used as an ML-based diagnosis model, the model should be robust despite the noises from the measurements. The testing success criteria is 1% of the Mean Absolute Percentage Error (MAPE) in actual scale [33]. The equation of the MAPE is

$$MAPE = \frac{100\%}{n}\sum_{t=1}^{n}\left|\frac{A_t - I_t}{A_t}\right| \tag{20}$$

where A_t is an actual value and I_t is an inferred value. The target variable to evaluate the model is the FCL temperature only. The inferred CL temperature would be used as a collateral information to identify the fuel cladding states.

Step 12-1. If the MAPE of the inferred FCL temperature in actual scale is larger than 1%, go to Step 2.

Since the model, which has passed the training success criteria, already uses the most useful learning scheme among the options for the case study, the learning algorithm does not need to be changed. Thus, a problem of the feature selection would be solved by re-selecting the input types.

Step 12-2. If the MAPE of inferred FCL temperature in actual scale is less than 1%, go to Step 13.

Through the testing process, the ML-based SSFIM is developed.

Step 13. Saving the Model in the Diagnosis Module Storage of Digital Twin

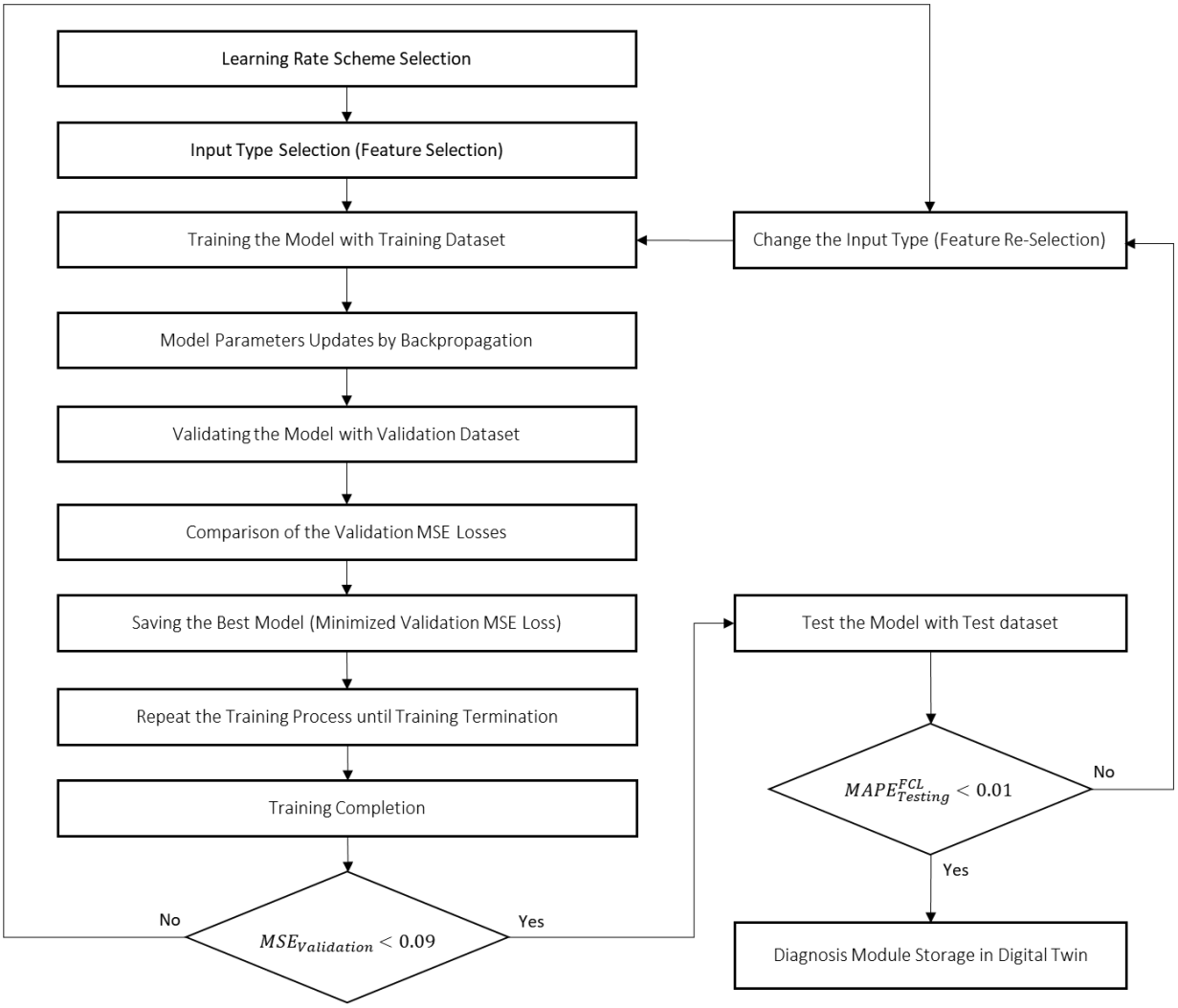


Figure 39. Machine Learning based Safety Significant Inference Model Development Workflow

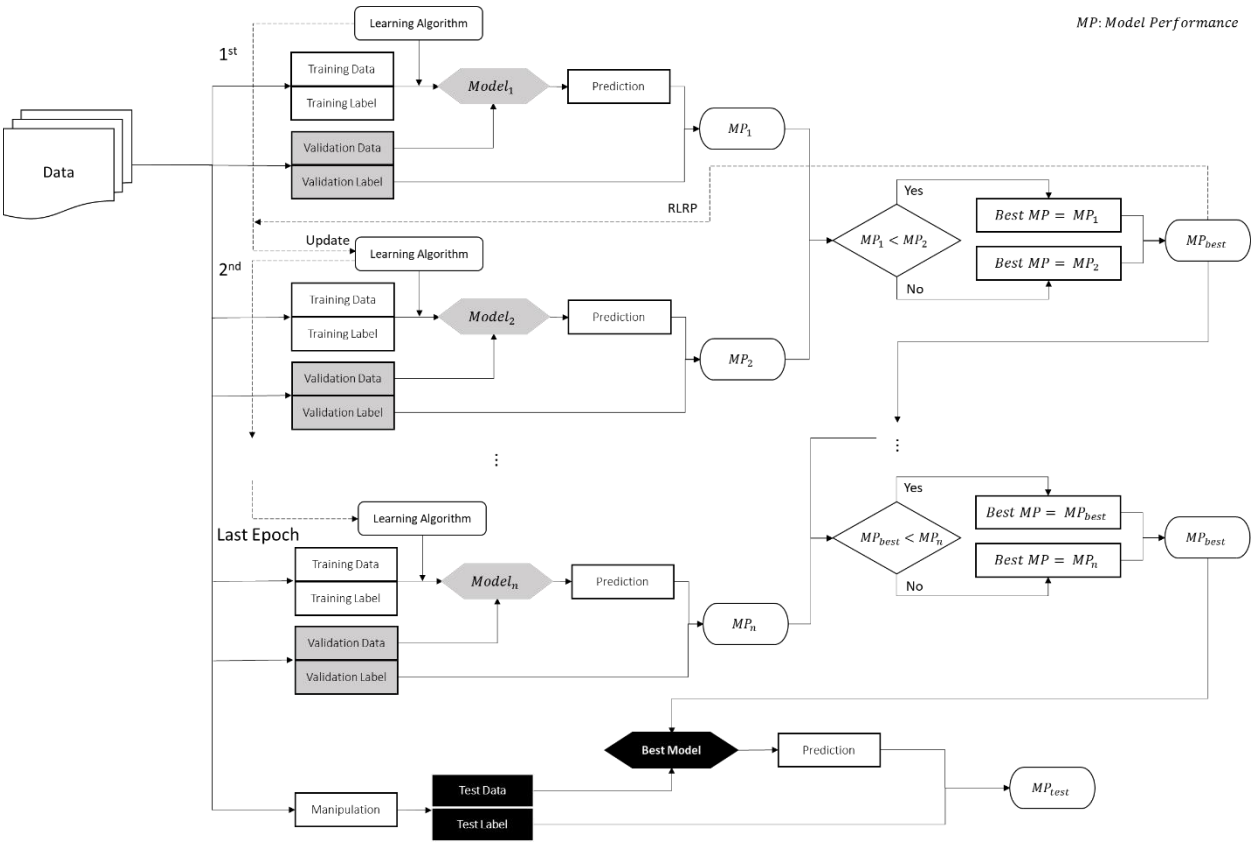


Figure 40. Three-way Hold-out Method in Training and Testing Process for Developing the ML-based SSFIM

- Success Criteria

There are two types of success criteria: training success criteria and testing success criteria. The numerical values of the criteria are different because they have distinct objectives.

✓ Success Criteria in the Training Process

Training success criteria aims to evaluate the model itself from the training. That is why the validation MSE loss, which is a normalized value, is used as a target metric. By the research of the quantitative structure-activity relationship (QSAR) model development, to be a good predictive model, the RMSE should below than 0.3 [49]. Hence, the training success criteria becomes to 0.09 of MSE value.

- ✓ Success Criteria in the Testing Process

The final evaluation aims to identify whether the developed model is applicable to infer the SSF in transient. The FCL temperature measurement is assured $\pm 1\%$ accuracy by using a tungsten/rhenium thermocouple [33]. In other words, there is an error within $\pm 1\%$ between actual temperature and measurement. Thus, if the result from the SSFIM has 1% MAPE in actual scale, the inferred value from the model is as reliable as a measured value from the thermocouple. As we assumed that the simulation data is an actual value, 1% of the simulation result can be a testing success criteria.

3.5.5. Challenges

The potential challenges of the training and testing process are divided into two parts: hyper-parameter adjustment and feature section. Both issues will be solved in the following case study.

3.6. Summary

For the development of the ML-based SSFIM, chapter 3 demonstrates necessary foundations: research assumption and hypotheses; developmental workflow for training and testing the model; and technical components, including data generation process, ML model components, training and testing process, and success criteria. Each technical component shows its challenge that would be overcome by the following case studies or excused by the research assumptions. Based on this knowledge and information, the ML-based SSFIM would be trained and tested in the following chapters.

CHAPTER 4. CASE STUDY DESCRIPTION

This case study has some distinct component requirements to accomplish a specific objective. The “No Free Lunch” theorem claims that the optimized algorithm developed to solve a certain problem is not optimized for other problems [50]. The first step for ML algorithm optimization is defining the problem with component requirements, such as interface, function, and modelling. In the research, a deep RNN is used for modelling components, while the function and interface are configured for the purposes of the objectives. The tasks in the case studies are designated to develop the robust SSFIM and draw the potential capabilities. The capabilities are identified by inferring the extended diagnosis range in the model. There are four case studies to achieve the research objectives described in the research scope chapter.

4.1. Case Study Component Requirements

- Case study 1. Building the ML Algorithm for developing the ML-based SSFIM

The case study 1 aims to build the ML algorithm for the ML-based SSFIM. It is related to the research objective that an ML-based Model is developed to infer the plant damage states. There are two sub-objectives: 1) identifying the RNN adequacy for building the ML-based SSFIM and 2) selecting the most efficient learning rate scheme to train the ML-based SSFIM. For attaining the sub-objectives, input and output are the physical variable data from the GOTHIC code and the inferred FCL and CL temperatures from the model, respectively. For sub-objective #2, three suggested learning schemes are dependent variables used to compare them each other’s results. The function of the model is that the ML-based SSFIM reasonably infers the SSF from the physical variable data. Additionally, case study 1 is a process to confirm the hypothesis (i).

- Case study 2. Inferring the SSF by using a robust model with the minimal physical variable data set

Through case study 2, the minimal physical variable set is acquired in inferring the SSF with a robust model. It is an advanced task used to materialize research objective #2 in which the input dataset is to be identified to infer the plant damage states. The function of the model is that

the ML-based SSFIM uses only a few physical variable types to infer the SSF, despite the random noise by the measurement error. Through the case study results, both hypothesis (ii) and (iii) will be confirmed. In order to prove them, input data is a set of a few physical variable data with manipulated random errors.

- Case study 3. Extending the diagnosis range by using the ML-based inference model

There are two approaches in extension of the information range: extension of diagnosis coverage and time span. Through case study 3, two advanced tasks are assigned to achieve objective #3 in which unknown information is extended from the given partial information.

The first task is that the unobserved abnormal physical phenomena are mapped by broadening the domain of the inferred variables from the core area to the entire reactor system. During a transient, even though abnormal physical phenomena must occur in every control volume, they are not observed because of cost inefficiency. However, in virtue of function of the model, the physical variables in the whole system are inferred by extending the view of information.

The second task aims to broaden the diagnosis time span by forecasting the next time step's SSF for developing the faster-than-real-time diagnosis model. Even though the diagnosis results come out from the measurement of current physical variables at the same time, the outputs are technically a past situation, because the abnormal physical symptoms are sequential phenomena during transient. Thus, predicting the next step's SSF enables the ML-based SSFIM to identify the plant damage states faster-than-real-time.

- Case Study 4. Identifying the inference ability of the ML-based inference model by extrapolating the SSF

Case study 4 aims to identify the inference ability of the ML-based model by extrapolating the SSF based on the relationship between the scenarios. In the research, inference ability is defined as being the event in which a certain model trained well on a transient can be applied to the test scenario that does not lie in the range of training scenarios. The case study evaluates the model's inference ability by testing a different type of transient data. As stated in sub-assumptions for solving the challenges of data generation, transient A, B, and C are divided by pump operation

status. Since each scenario has discrete scenarios, the data coverage or inclusion relationship between transient types would be revealed through the test results.

The case studies consist of the assignments to develop the ML-based SSFIM and show the capabilities of the model. Table 15 and 16 summarize the case study objectives and component requirements. These are organically connected to contribute for establishment of the ML-based diagnosis model by proving the inference ability of the RNN. Figure 41 illustrates the workflow of how the case studies are implemented in this section. Since each case study includes its success criteria, failure or achievement of the tasks depend on each model's performance. Even though the case study's objective is not achieved, but there are some lessons from the results for developing the ML-based model.

Table 15. Case Study Description with Component Requirements

Case Study	Component Requirements		
1	Objective	Building the ML algorithm for developing the ML-based SSFIM • Identifying the RNN adequacy for building the ML-based SSFIM • Selecting the most efficient learning rate scheme to train the ML-based SSFIM	
	Function	Inferring the SSF from the physical variables	
	Interface	Independent Variable	Learning Rate Scheme
		Dependent Variable	Validation Loss and Inferred SSF Values (MAPE)
		Input	Various Physical Variables Data (Type 1 Test Dataset)
		Output	FCL and CL Temperature
	Focuses	Learning Rate Scheme and Algorithm Adequacy	
2	Objective	Inferring the SSF by using a robust model with the minimal physical variable data set • Finding a minimal input dataset to infer the SSF • Inferring the SSF from the physical variable data including the random noises	
	Function	Inferring the SSF by using only a few measured physical variables	
	Interface	Independent Variable	Physical Variable Types and Random Error Magnitude
		Dependent Variable	Inferred SSF Values (MAPE)
		Input	Manipulated Physical Variables Data (Type 1, Type 2 and 3 Test Dataset)
		Output	FCL and CL Temperature
	Focuses	Minimal Dataset and Random Noise	

Table 16. Case Study Description with Component Requirements (Continued)

Case Study	Component Requirements				
3	Objective	Extending the diagnosis scope by using the ML-based inference model • Extending the diagnosis coverage • Extending the diagnosis time span			
	Task 1. Mapping the unobserved abnormal physical phenomena in the reactor system				
	Function	Inferring the physical variables in the whole system by extending the view of information			
	Interface	Input	Minimal Physical Variables Dataset		
		Output	Temperatures in Every Control Volume		
		Test Case	Transient B		
	Task 2. Developing the faster-than-real-time diagnosis inference model				
	Function	Predicting the next time step's SSF based on current measurement			
	Interface	Input	Minimal Physical Variables Dataset		
		Output	Next Time Step's FCL Temperature		
		Test Case	Transient C		
	Focuses	Extension of the information range, Mapping and Faster-than-real-time			
4	Objective	Identifying the Inference Ability of the ML-based inference model by extrapolating the SSF			
	Function	Revealing data coverage or inclusion relationship between scenarios			
	Interface	Input	Minimal Physical Variables Dataset		
		Output	FCL Temperature		
	Training Data	Transient A	Transient B	Transient C	
	Testing Data	Transient B and C	Transient A and C	Transient A and B	
	Focuses	Inference Ability, Extrapolation and Transient Scenarios			

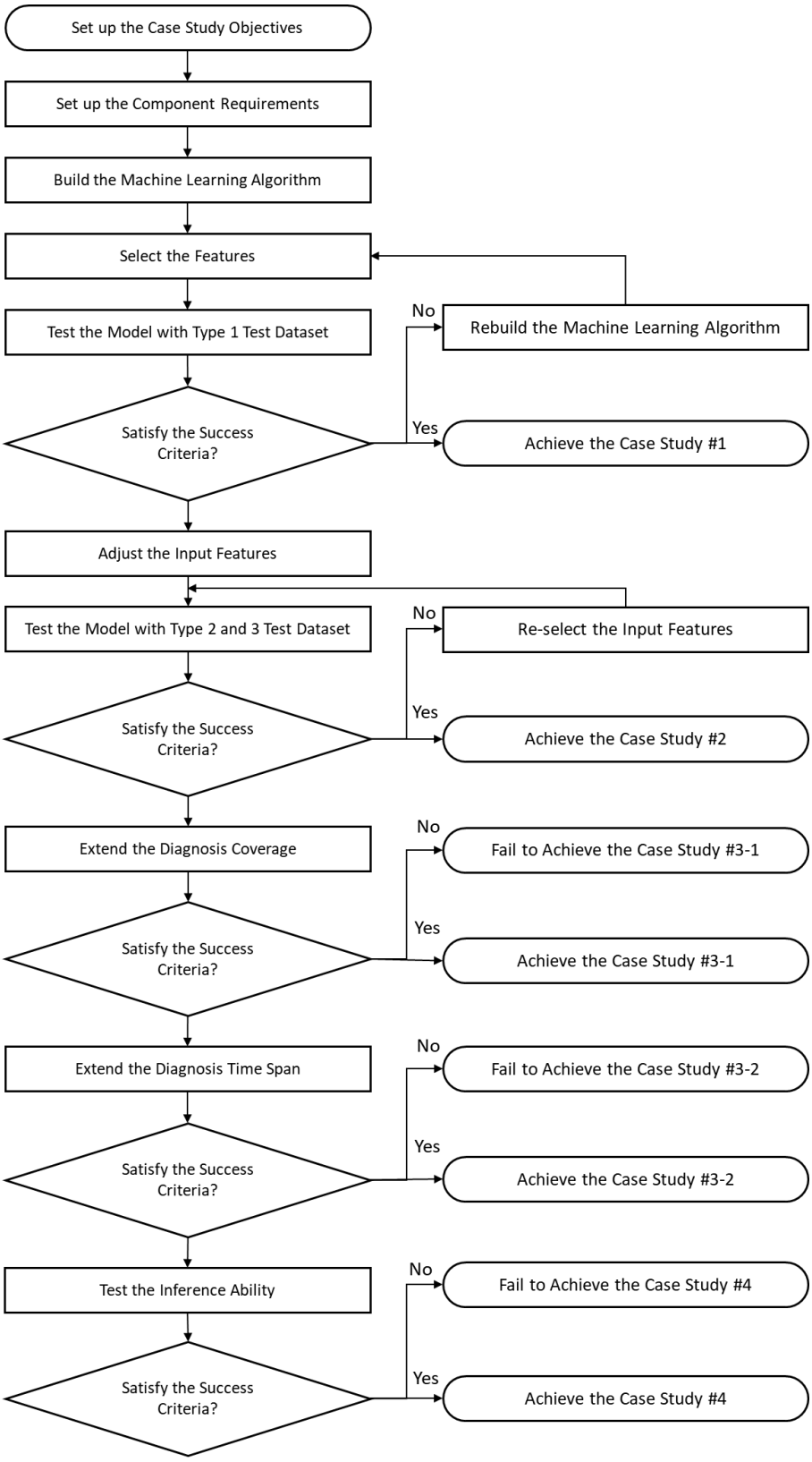


Figure 41. Case Study Workflow

4.2. Model Name

Since the SSFIMs are classified by interface requirements, the models are named by revealing the input types in order to distinguish them. The form of the name follows the following rules:

- ✓ The first letter of the model means the transient case, e.g. transient A, B or C.
- ✓ The next character indicates the input types, such as temperature, pressure, and mass flow rates.
- ✓ Output is omitted in the name because it is fixed to FCL and CL temperatures.
- ✓ For example, an "A_t914_fl16" is a model which uses temperatures in the control volume #9 and #14; and mass flow rates in the flow path #1 and #6 as inputs in transient case A.
- ✓ Additional information is stated at the end of the model name, if it is necessary,

Based on the rules, the assigned input types and target transient cases are known from the models' name. The models designate the input variables by using the control volume numbers which are shown in table 17.

Table 17. Control Volume Number Labels

Number	Control Volume	Number	Control Volume
1	Primary Tank	12	Inner Reflector
2	Pump 1 HP Vertical Pipe	13	Outer Blanket
3	Pump 1 HP Horizontal Pipe	14	Upper Plenum
4	Pump 1 LP Pipe	15	Outlet Pipe
5	Pump 2 HP Vertical Pipe	16	IHX Primary Shell
6	Pump 1 HP Horizontal Pipe	17	IHX Secondary Downcomer
7	Pump 2 LP Pipe	18	IHX Secondary Lower Plenum
8	HP Lower Plenum	19	IHX Secondary Tube
9	LP Lower Plenum	20	IHX Secondary Upper Plenum
10	Active Core	21	IHX Secondary Outlet Pipe
11	Control Rod		

HP: High Pressure
LP: Low Pressure
IHX: Intermediate Heat Exchanger

4.3. Summary

In this chapter, the case studies are described through the component requirements and a specific objective. The case studies are connected in one direction to prove its ability as the ML-based diagnosis model. The models are named to reveal the features of the model, such as input, training dataset, and so on, to be easily discerned by users. The tasks would be implemented in the following chapter.

CHAPTER 5. CASE STUDY IMPLEMENTAION

In this section, case studies are implemented to achieve the objectives and confirm the hypotheses. The ML-based model is trained in accordance with the interface requirements and tested to accomplish the functional requirements.

5.1. Building the ML Algorithm for Developing the ML-based SSFIM

5.1.1. Learning Rate Scheme Evaluation

To find a proper learning rate scheduler among three policies in the research, the validation losses are compared. There are two types of transient A models: four temperatures for the input dataset and two temperatures and two mass flow rates for the input dataset (see table 18). Six models in total are trained to compare the results by tracking the validation loss in every epoch.

Table 18. Models used for Evaluation of the Leaning Rates Scheme

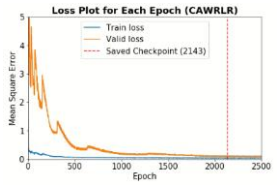
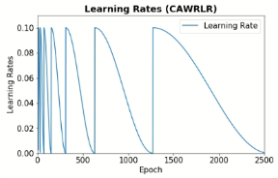
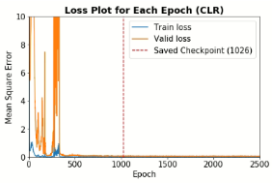
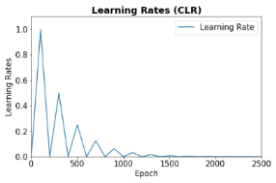
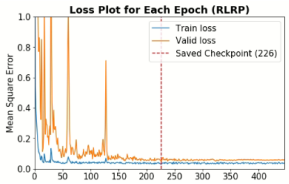
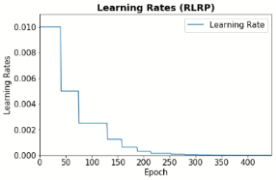
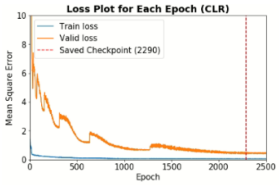
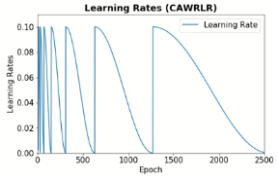
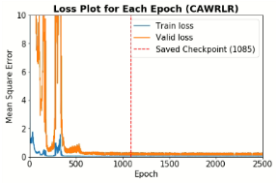
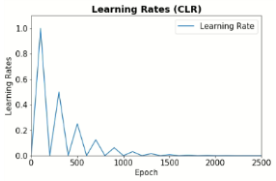
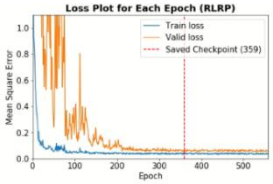
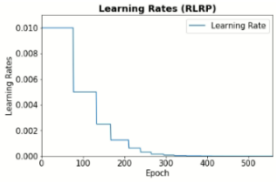
Model Name	Target Case	Input Types	Learning Rate Scheme
A_t914_fl16_CAWR	Transient A	Temperature #9 and #14 Mass Flow Rates #1 and #6	CAWRLR
A_t914_fl16_CL			CLR
A_t914_fl16_RLRP			RLRP
A_T4_CAWR		Temperature #9, #14, #16, and #21	CAWRLR
A_T4_CL			CLR
A_T4_RLRP			RLRP

As we can see in table 19, the RLRP is the most efficient scheduler for the case study. The best validation loss of the "A_t914_fl16_RLRP" is smaller than other models which use the CAWRLR and CLR. Moreover, for the model training, the CAWRLR and the CLR take 170.42 hours and 168.54 hours, respectively, whereas, when a quad-core 3.40 GHz processor is used, the training with RLRP finished in 18.82 hours. Because the training with two former schedulers would be finished at 2500th epoch, on the other hand, the RLRP terminates the training process when the validation loss does not decrease during the last 200 epochs. Owing to the rule in policy, the last learning rates of the "A_t914_fl16_RLRP" and "A_T4_RLRP" models are different from each other. Finally, in the case of the "A_t914_fl16_RLRP," the smallest best epoch among the policies allows the shortest training time among them by stopping the training at the 426th epoch.

Similarly, the “A_T4_RLRP” also converges faster compared to other schedulers. However, the last learning rate of the “A_T4_RLRP” is smaller than that of “A_ t914_fl16_RLRP,” even though the validation loss is larger. The small learning rate does not guarantee less validation loss for the better trained model.

The RLRP enables the model to train faster with smaller validation loss than other policies. Updating the model parameters to the minimized value has failed during given numbers of an epoch, since the CAWRLR and CLR, which repeat the increase and decrease of the learning rates by the rules, disturb the weight to converge (see the learning rates plot). The MSE repeats the divergence and convergence with a large amplitude compared to the RLRP (see the validation loss plot). In conclusion, the RLRP policy is used hereafter.

Table 19. Learning Rate Scheme Evaluation Results

Model Name	Train and Validation Loss Plot	Learning Rates Plot	Best Validation Loss Best Epoch/ Last Epoch Last Learning Rates Computational Time [hr]
A_t914_fl16_CAWR			0.0871 2143/2500 1.266E-04 170.42 [hr]
A_t914_fl16_CL			0.0684 1026/2500 2.051E-04 168.54 [hr]
A_t914_fl16_RLRP			0.0486 226/426 1.221E-06 18.82 [hr]
A_T4_CAWR			0.1189 2290/2500 1.266E-04 162.33 [hr]
A_T4_CL			0.401 1085/2500 2.051E-04 172.25 [hr]
A_T4_RLRP			0.0498 359/559 1.526E-07 20.18 hr]

5.1.2. RNN Adequacy Evaluation

The main objective of case study 1 is to identify the ML algorithm adequacy by evaluating the model performance. Table 20 shows the input types for developing the SSFIM. Control volumes of the coolant temperature are arbitrarily chosen but they include the inlet (LPLP) and the outlet (UP). The paths of mass flow rate consist of each pump pipe and the core inlet pipe. Also, the pressure in the HPLP is pressure input data. By combining the input data, 8 types of input datasets are used to train the RNN. A total of 24 models (8 types of inputs × 3 transient cases) are developed and tested by the type 1 test dataset of A-53, B-53, or C-37. The validation loss, RMSE, and MAPE of FCL temperature are shown in table 21 to show the training and testing results. The RMSE indicates a difference between the inferred value (I_t) and actual value (A_t) in a de-normalized scale calculated by the below equation.

$$RMSE = \sqrt{\sum_{t=1}^{n} \frac{(I_t - A_t)^2}{n}} \tag{21}$$

Table 20. Input Variables of the SSFIM for Case Study 1

Model Name (Target Case_)	Control Volume Number		
	Temperature	Pressure	Mass Flow Rates
T4	#9, #14, #16, and #21	-	-
t914_fl16	#9 and #14	-	#1 and #6
t914_pr8_fl16		#8	
T4_FL3	#9, #14, #16, and #21	-	#1, #6, and #11
T8	#1, #9, #14, #15 #16, #17, #20 and #21	-	-
T8_FL3		-	#1, #6, and #11
T8_pr8_FL3		#8	

Table 21. Training and Test Results of Case Study 1

Model Name	Input Numbers	Validation Loss	RMSE of FCL Temperature	MAPE (%) of FCL Temperature
A_T4	4	5.250E-02	1.845E+00	1.665E-01
A_t914_fl16	4	4.560E-02	1.530E+00	1.532E-01
A_t914_pr8_fl16	5	4.280E-02	1.519E+00	1.546E-01
A_T4_FL3	7	3.690E-02	1.748E+00	1.684E-01
A_T8	8	5.140E-02	1.811E+00	1.500E-01
A_T8_FL3	11	3.600E-02	1.703E+00	1.687E-01
A_T8_pr8_FL3	12	4.110E-02	1.509E+00	1.504E-01
B_T4	4	8.700E-03	8.942E-01	1.091E-01
B_t914_fl16	4	9.600E-03	8.384E-01	9.856E-02
B_t914_pr8_fl16	5	7.100E-03	7.607E-01	9.201E-02
B_T4_FL3	7	7.500E-03	7.599E-01	8.878E-02
B_T8	8	7.800E-03	1.035E+00	1.224E-01
B_T8_FL3	11	6.900E-03	6.866E-01	7.204E-02
B_T8_pr8_FL3	12	8.200E-03	8.119E-01	1.035E-01
C_T4	4	1.350E-02	3.312E+00	3.667E-01
C_t914_fl16	4	1.200E-02	2.674E+00	2.825E-01
C_t914_pr8_fl16	5	1.050E-02	1.868E+00	2.319E-01
C_T4_FL3	7	8.400E-03	1.938E+00	2.306E-01
C_T8	8	1.390E-02	3.018E+00	3.475E-01
C_T8_FL3	11	1.060E-02	2.085E+00	2.491E-01
C_T8_pr8_FL3	12	1.040E-02	1.757E+00	1.998E-01

As we can see in table 21 and figure 42, the FCL temperature inferred by the ML-based SSFIM is very close to actual FCL temperature from the simulation. There are several noticeable points in testing results:

1) The validation losses of every trained model are less than 0.09 and the MAPEs of the FCL temperature are within 1%. As we can see in table 1, the “B_T8_FL3” has 6.900E-03 of the validation loss and 7.204E-02 % of the MAPE in the FCL temperature when the trained model is tested on the type 1 test dataset. In other words, the RNN output is nearly identical

to the GOTHIC result when the random noise is not considered in the physical variable measurements.

2) The validation loss of the transient A model is larger than the transient B and transient C models. This means the model, which consists of multi head events in the transient scenario, is trained better than the model with a single head event. However, more variables in a scenario do not necessarily lead the ML model to learn better (see figure 43). The transient B model tends to have a smaller validation loss than others.
3) The RMSE and MAPE of the transient C model is larger than the other models (see figure 44). There are several chances to have differences between the actual value and the inferred value because the trend of the FCL temperature changes multiple times due to scenario.
4) The number of input data is irrelevant to the accuracy of the model. The model performance of the model depends on what type of input data is used rather than how many inputs are required. Further studies are properly executed in the following section.

In virtue of the findings, the following case study is motivated to investigate which physical variable is critical to infer the SSF and how many input types are required to get reasonable level of the SSF. In order to conduct various tasks, the RNN is utilized as a core engine in the ML model.

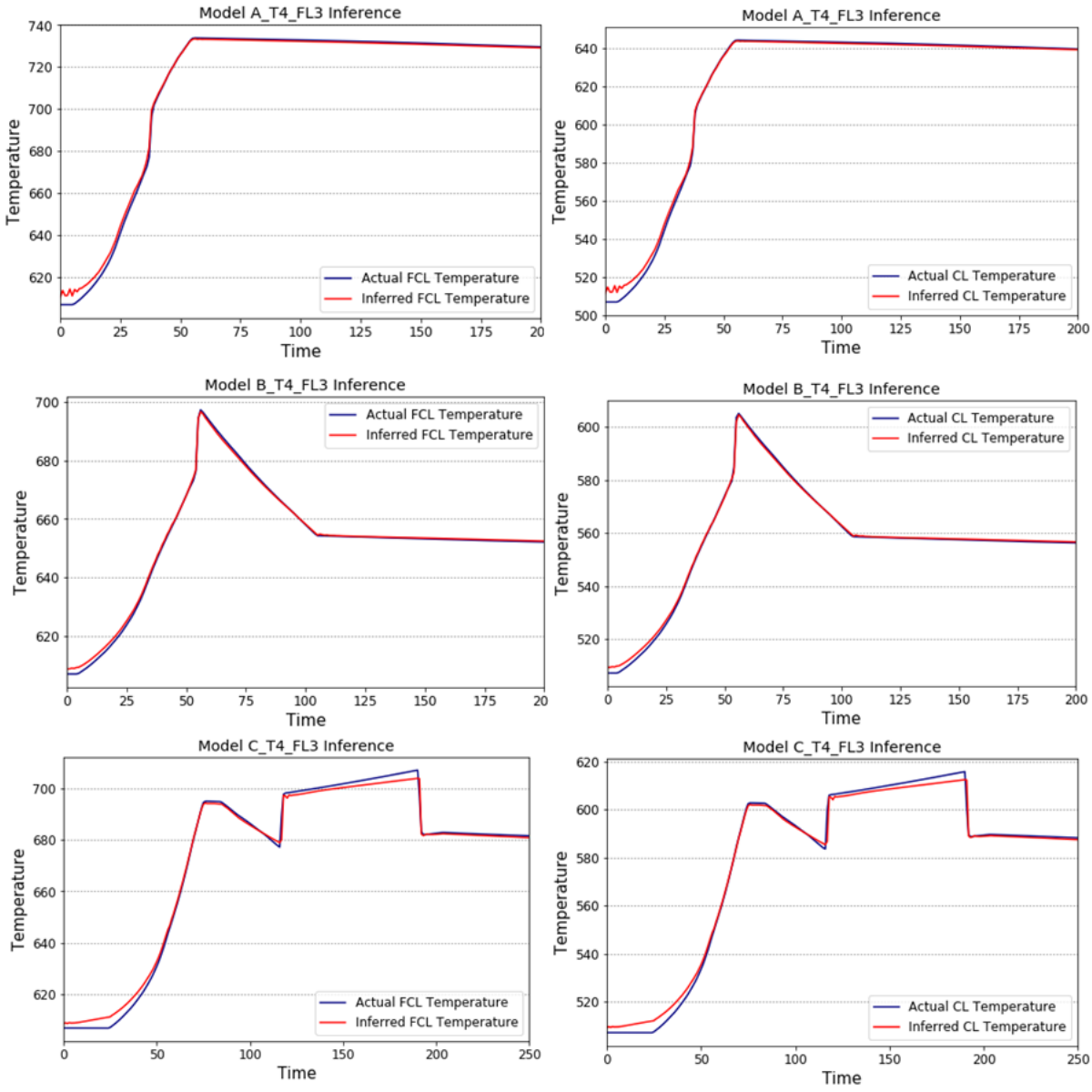


Figure 42. SSF Inference Test Results by using the T4_FL3 Model

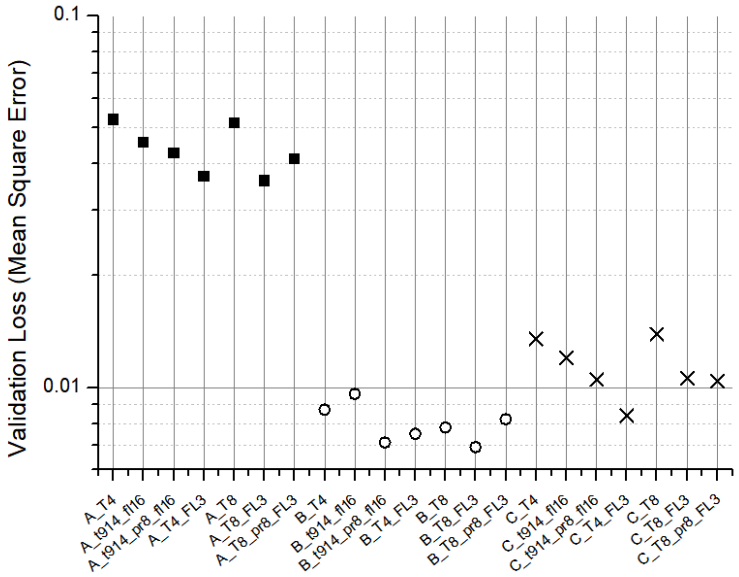


Figure 43. Validation Loss of the Trained Models

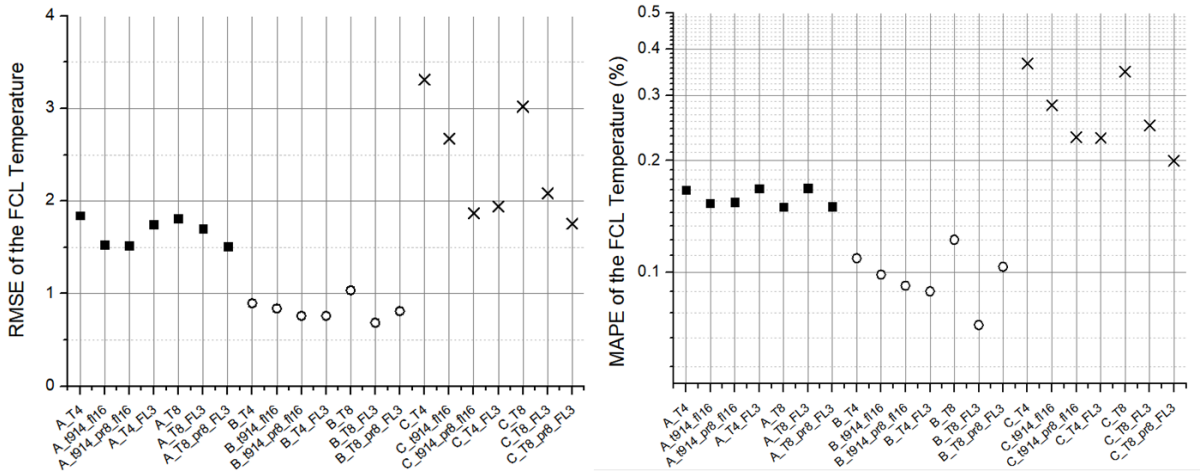


Figure 44. RMSE and MAPE of the FCL Temperature on Type 1 Test Dataset

5.1.3. Summary

The RNN with the RLRP policy is an appropriate algorithm to deal with the physical phenomena data to infer the SSF. By identifying the adequacy of the RNN through the case study results, the hypothesis #1 is confirmed and the potential challenge of the ML algorithm is solved.

5.2. Inferring the SSF by using only a few Measured Physical Variables

According to the combination of the input dataset, a total of 34 types of the models are prepared to implement the case study 2. Table 22 demonstrates that the models are named to reveal the input data by using the numbers of the control volumes. There are 8 temperatures, one pressure, and three mass flow rates to make a diverse assortment including single-type input set, for instance, temperature only model. In order to identify which variables are influential or useless in the dataset by testing, various ML-based models are trained. The A-53, B-53, and C-37 are used to show the results in the section. The section consists of type 1 test results to identify the relationship between input variables and SSF, type 2 results to sort out a robust model, and type 3 results to find a robust model with a minimal dataset.

Table 22. Inputs Description for the SSFIM

Model Name	Input Numbers	Control Volume Numbers		
		Temperature	Pressure	Mass Flow Rates
t9	1	#9	-	-
t14		#14	-	-
fl1		-	-	#1
fl6		-	-	#6
fl11		-	-	#11
pr8		-	#8	-
t914	2	#9 and #14	-	-
t9_fl1		#9	-	#1
t9_fl6			-	#6
t9_fl11			-	#11
t9_pr8		-	#8	-
t14_fl1		#14	-	#1
t14_fl6			-	#6
t14_fl11			-	#11
pr8_fl1		-	#8	#1
fl16		-	-	#1 and #6
t91416	3	#9, #14, and #16	-	-
t914_fl1		#9 and #14	-	#1
t914_fl6			-	#6
t914_fl11			-	#11
t914_pr8			#8	-
t9_fl16		#9	-	#1 and #6
t9_pr8_fl1			#8	#1
t14_fl16		#14	-	#1 and #6
pr8_fl16		-	#8	
fl1611		-	-	#1, #6, and #11
T4	4	#9, #14, #16, and #21	-	-
t914_fl16		#9 and #14	-	#1 and #6
t914_pr8_fl16	5		#8	
T4_FL3	7	#9, #14, #16, and #21	-	#1, #6, and #11
T8	8	#1, #9, #14, #15 #16, #17, #20 and #21	-	-
T4_pr8_FL3		#9, #14, #16, and #21	#8	#1, #6, and #11
T8_FL3	11	#1, #9, #14, #15 #16, #17, #20 and #21	-	
T8_pr8_FL3	12		#8	

5.2.1. Identifying the Relationship between Inputs and Outputs of the SSFIM

Figure 45 shows the type 1 testing results of MAPE. There are several findings that

1) The number of inputs does not affect the model performance. More input numbers do not guarantee a smaller error but does not increase the redundancy to infer the SSF. By using only one sensory data except "t9," the FCL temperature is successfully inferred in all transient cases (see figure 46) when there is no random noise from the measurement.
2) Temperature at LPLP is an inappropriate variable to be used as a single-input-model.
3) In conclusion, the main concern to compose the model is not the number of inputs but the variable type. The "t9" model does not qualify to be an ML-based SSFIM.

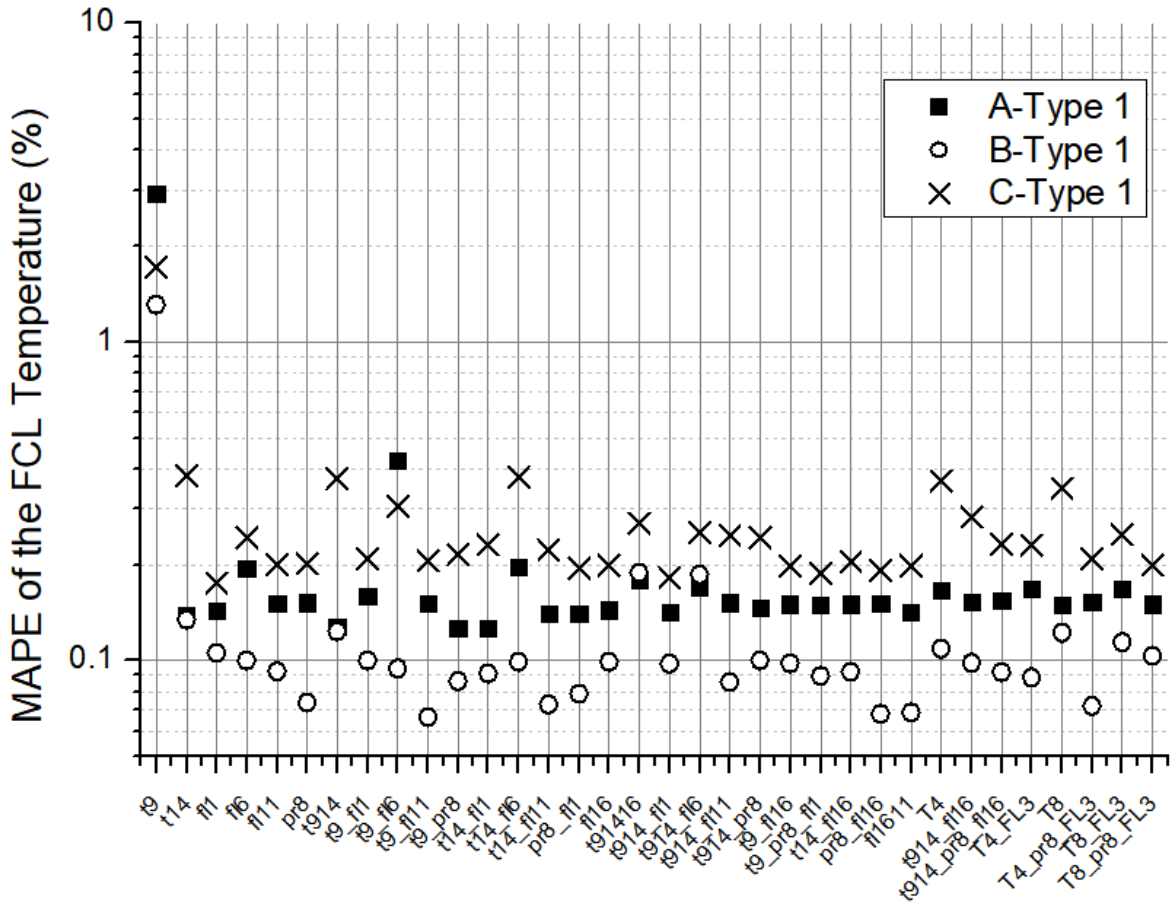


Figure 45. Type 1 Test Results of the ML-based SSFIM

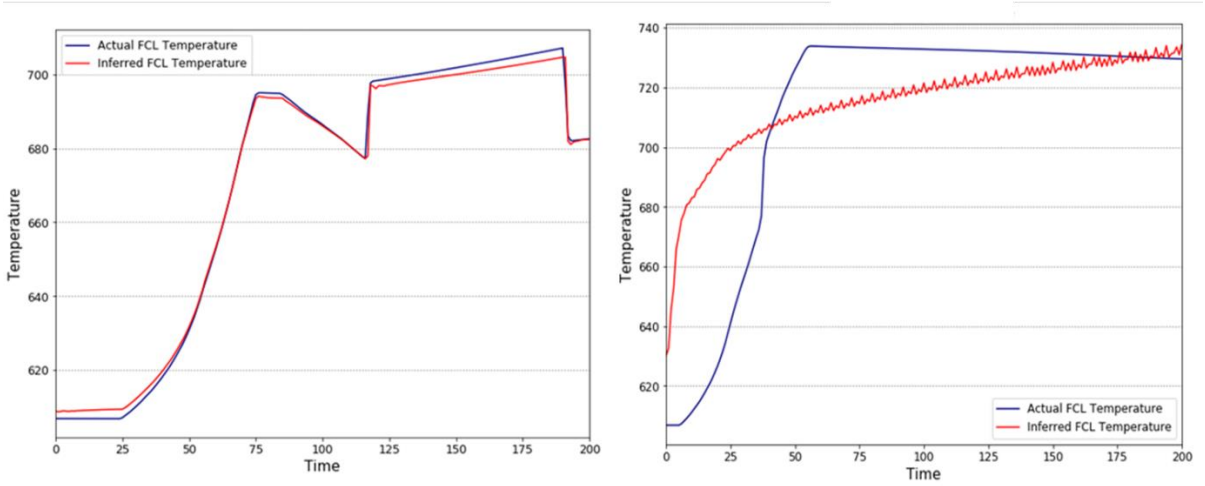


Figure 46. SSF Inference Results of "C_fl11" Model (left) and "A_t9" Model (Right)

5.2.2. Sorting out the Robust Model

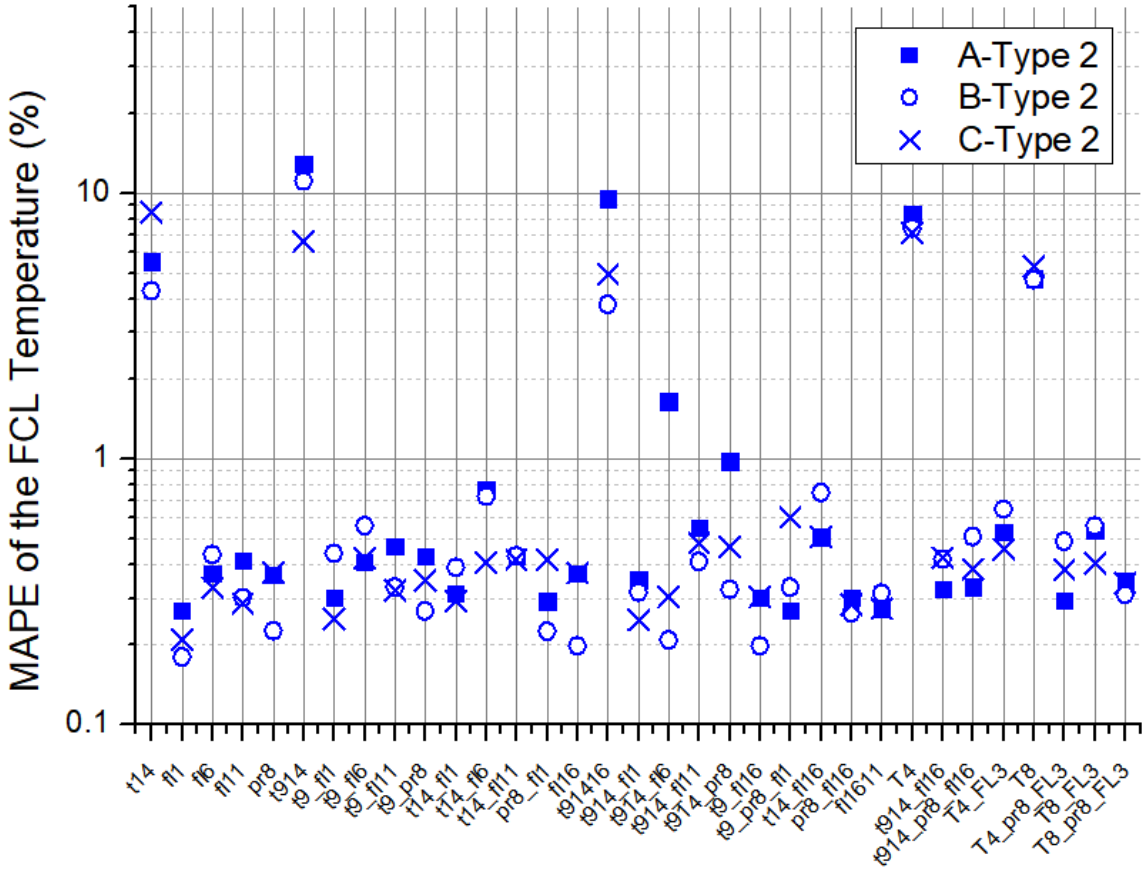


Figure 47. Type 2 Test Results of the ML-based SSFIM

Figure 47 shows the MAPE of the test results of the type 2 dataset. Findings from the results are as follows:

1) Temperature is sensitive to random noise. The temperature only models show a wide fluctuation in the inferred SSF (see figure 48).
2) On the contrary, the mass flow rate is an undisturbed physical variable to the measurement error. The mass flow rate-only models satisfy the success criteria. Moreover, even though many temperatures are included in the input dataset with the mass flow rates, the MAPE of the inferred FCL is within 1% (see figure 5-8). From this perspective, the mass flow rates are dominant and powerful variables to infer the SSF.
3) In the case of the "t914_fl6" model for transient A, the model performance is slightly over the success criteria. That is because the transient A scenario contains the stablized-state for a long period after ramping down.
4) In conclusion, the mass flow rate must be involved in the input dataset due to robustness to random noise.

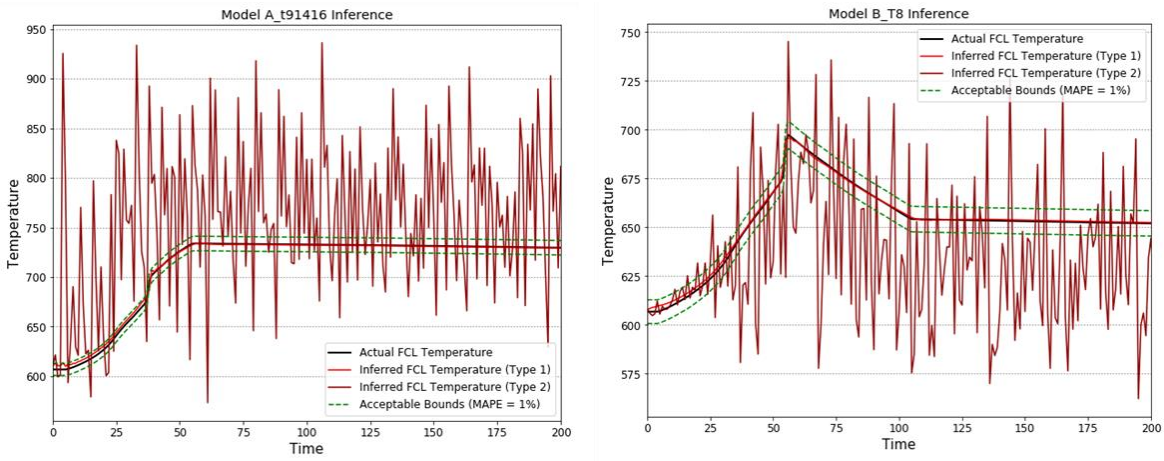


Figure 48. Inferred SSF by "A_t91416" Model (Lest) and "B_T8" Model (Right)

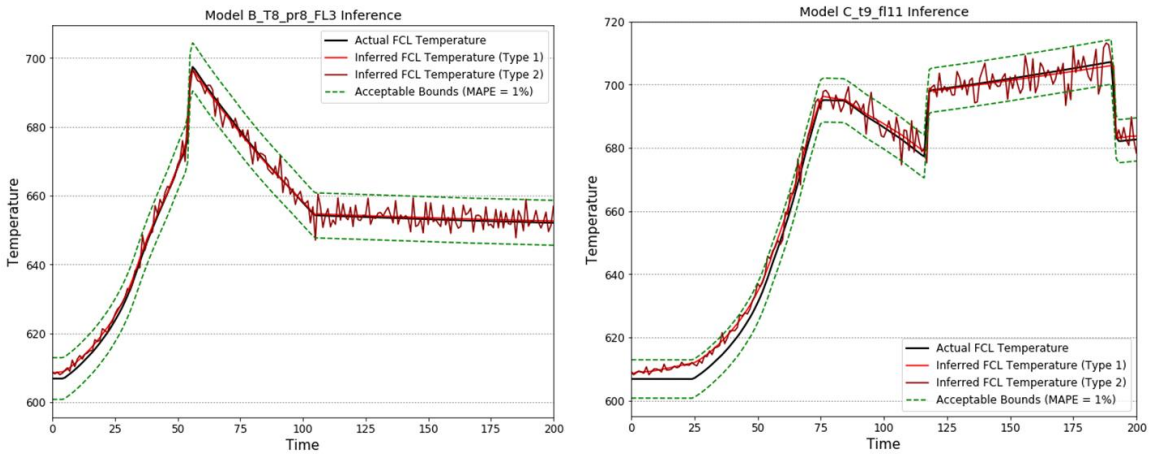


Figure 49. Inferred SSF by "B_T8_pr8_FL3" Model (Lest) and "C_t9_fl11" Model (Right)

5.2.3. Finding a Robust Model with the Minimal Dataset

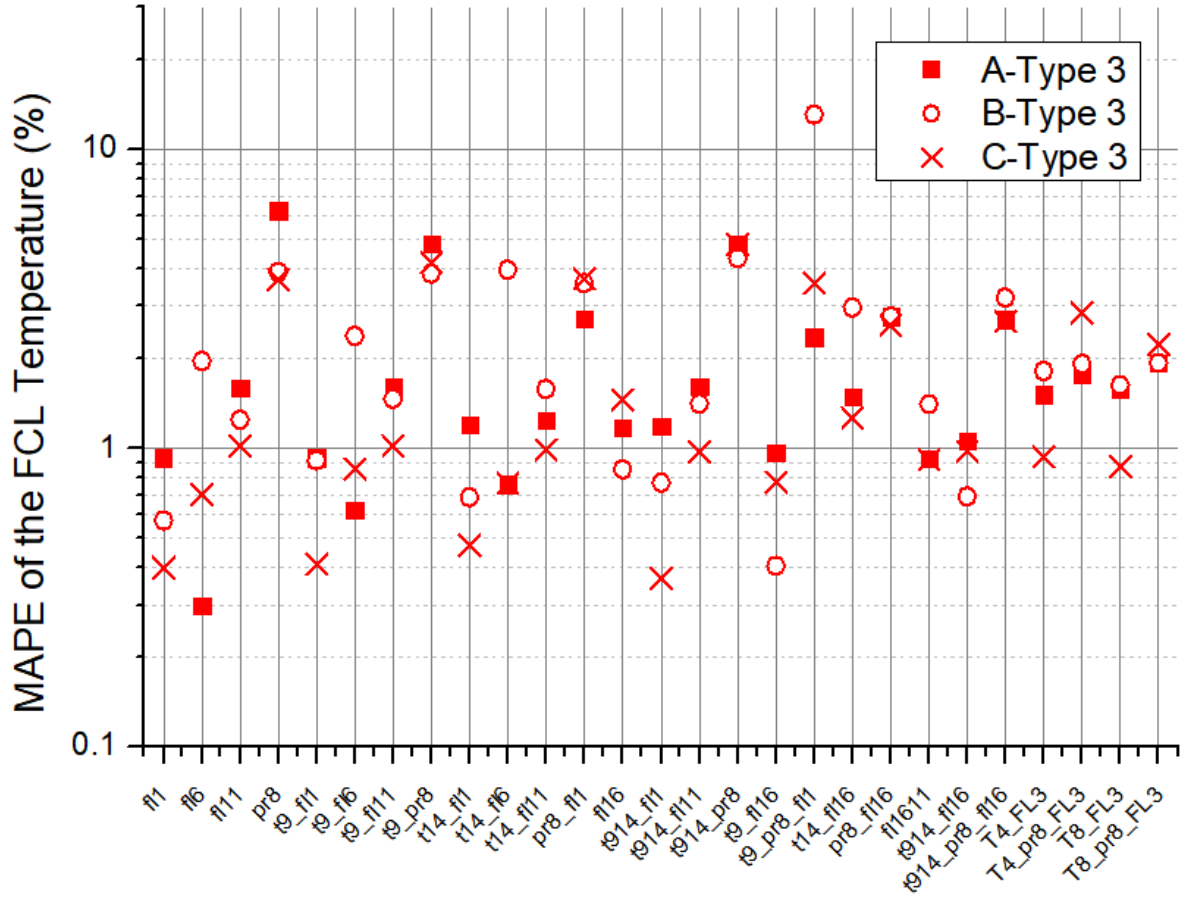


Figure 50. Type 3 Test Results of the ML-based SSFIM

Figure 50 shows the MAPE of the test results of the type 3 dataset. Findings from the results are as follows:

1) Since the type 3's random noise in the mass flow rates is bigger than the type 2's, most of the mass flow rate-only models result in the unacceptable MAPE level. However, the "fl1" model shows good performance. That means, the mass flow rates in the pump #1 path are the most robust variable among the suggested physical variables.
2) Even though the "fl1" is the crucial variable, the MAPE of the models tend to be larger than 1% when other variables are combined with it. However, adding only "t9" does not impede the accuracy because the random noise of the temperature in type 3 is less than that in type 2.
3) The pressure is treated as a useless variable because of the large random noise.

Table 23. MAPE of the FCL and CL Temperatures in Each Models

MAPE (%)	Type 1		Type 2		Type 3	
	FCL	CL	FCL	CL	FCL	CL
A_fl1	1.437E-01	1.834E-01	2.705E-01	3.402E-01	9.281E-01	1.149E+00
A_t9_fl1	1.593E-01	1.985E-01	3.003E-01	3.748E-01	9.277E-01	1.150E+00
A_t9_fl16	1.503E-01	1.932E-01	3.022E-01	3.823E-01	9.711E-01	1.204E+00
B_fl1	1.057E-01	1.341E-01	1.799E-01	2.273E-01	5.744E-01	7.210E-01
B_t9_fl1	1.002E-01	1.242E-01	4.428E-01	5.621E-01	9.119E-01	1.149E+00
B_t9_fl16	9.827E-02	1.236E-01	1.980E-01	2.545E-01	4.043E-01	5.153E-01
C_fl1	1.752E-01	2.275E-01	2.103E-01	2.728E-01	4.001E-01	5.085E-01
C_t9_fl1	2.085E-01	2.665E-01	2.502E-01	3.163E-01	4.104E-01	5.145E-01
C_t9_fl16	1.985E-01	2.534E-01	3.026E-01	3.845E-01	7.712E-01	9.758E-01

Table 24. Average MAPE of the "fl1" Model, "t9_fl1" Model, and "t9_fl16" Model

Average MAPE (%)	"fl1" Model	"t9_fl1" Model	"t9_fl16" Model
Type 1 FCL	1.415E-01	1.560E-01	1.490E-01
Type 1 CL	1.817E-01	1.964E-01	1.900E-01
Type 2 FCL	2.202E-01	3.311E-01	2.676E-01
Type 2 CL	2.801E-01	4.177E-01	3.404E-01
Type 3 FCL	6.342E-01	7.500E-01	7.155E-01
Type 3 CL	7.929E-01	9.378E-01	8.984E-01

4) Through the three steps, input datasets, which are able to robustly infer the FCL temperature within 1% accuracy, are "fl1," "t9_fl1," and "t9_fl16." The MAPE of the three models are distributed as 0.1~0.2% in type 1, 0.2~0.4% in type 2, and 0.4~0.9% in type 3

(see table 23). These values indicate that the accuracy of the inferred temperature from the SSFIM is higher than the accuracy of the measured temperature from the thermocouple.

5) The MAPE of the inferred CL temperature are distributed as 0.1~0.3% in type 1, 0.2~0.6% in type 2, and 0.5~1.2% in type 3. These are larger than the MAPE of the FCL temperature, because the CL temperature is lower than FCL temperature.
6) Based on the averages of the transients A, B, and C, the MAPE of the "fl1" model is smallest and the MAPE of the "t9_fl1" model is the largest among the three models (see table 524).
7) The MAPE of the transient A is largest among the transient scenarios since the stabilized-state with fluctuation due to noises is longer than the others (see figure 51 and figure 52).
8) In conclusion, the minimal dataset used to infer the SSF is a [mass flow rate in the pump #1 path]. In other words, if we correctly measure the mass flow rate at pump #1's flow path, it is enough to infer the FCL temperature and CL temperature. However, for further case study, not only the "fl1" model but the "t9_fl16" model is used to implement the advanced tasks.

Detailed inferred results including RMSE and MAPE of each model are shown in Appendix C.

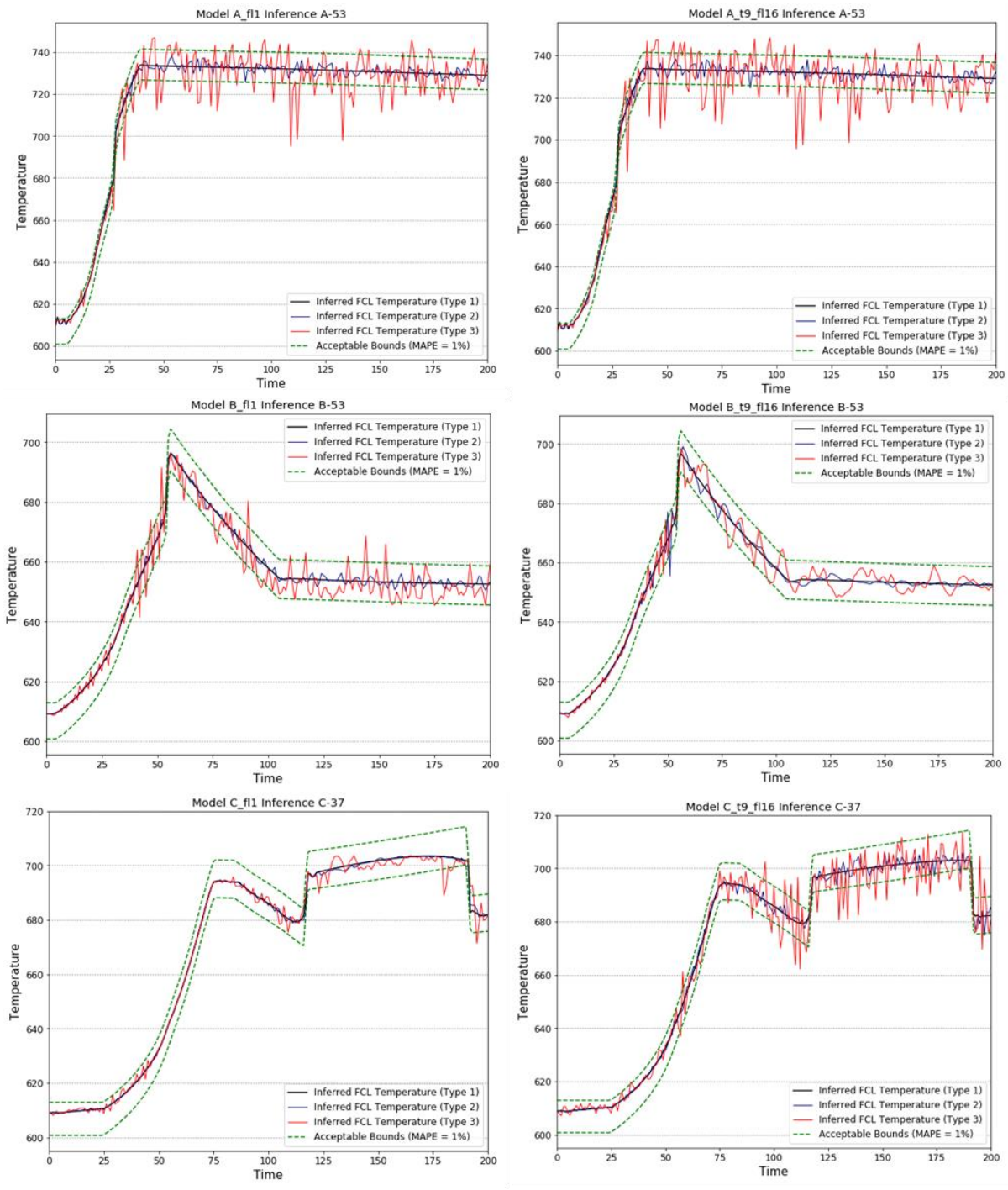


Figure 51. Inference Results by "fl1" model and "t9_fl16" model

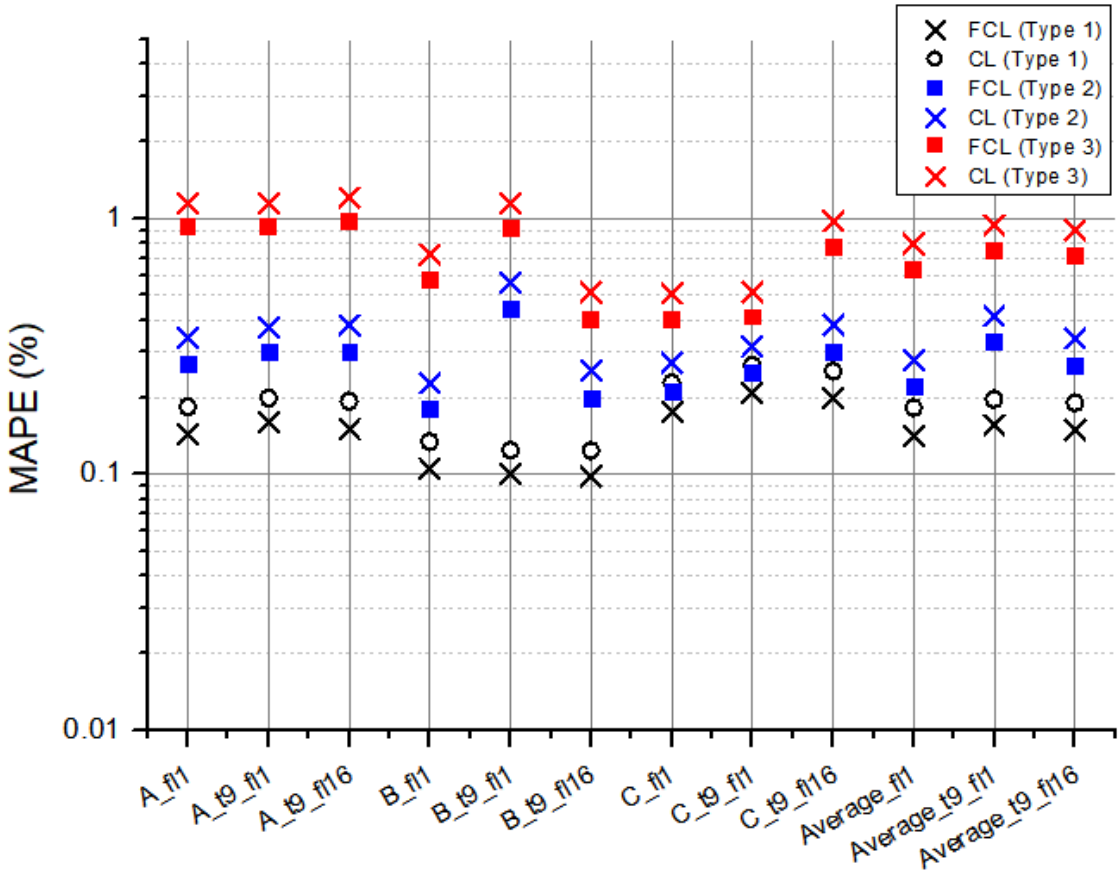


Figure 52. MAPE Distribution for Determining the Minimal Input Dataset

5.2.4. Summary

Through case study 2, the robust model, which consists of the minimal dataset, is developed. The evaluated ML-based model performance is as good as using the SSFIM instead of the thermocouples. What type of physical variable is used as an input dataset for the SSFIM matters when designing the model rather than how many input variables are required to reasonably infer the SSF. The mass flow rate in pump #1 is both the minimal dataset and the most important variable for the robust SSFIM. Since the temperature is very sensitive to random noise from the measurement, two or more temperatures become a hindrance factor. Also, the pressure is not a necessary variable. For case studies 3, the "fl1" and "t9_fl16" models, which are the best two models, are used to achieve the objectives by inferring the extended information.

5.3. Extending the Diagnosis Range by using the ML-based Inference Model

5.3.1. Mapping the unobserved abnormal physical phenomena in the reactor

As mentioned above, there are two approaches in extension of the diagnosis range: extension of diagnosis coverage and time span. In order to infer SSF, which is represented by the FCL temperature and CL temperature, two outputs are inferred from the RNN. By expanding the dimension of the output of the ML algorithm, the deduced information is not limited to the SSF. Case study 3 is implemented to map the inside of the reactor system from only a few physical variable measurements.

Based on the minimal datasets, all the temperatures in the EBR-II primary system are disclosed by the inventive ML-based inference model. Table 25 demonstrates the labels in the GOTHIC code and corresponded 29 cells in which the coolant temperatures can be measured. Since the liquid sodium temperatures are targets to be inferred, the success criteria should be changed. According to the EBR data digitalization documentation, as we assumed above, the thermocouples for the coolant temperature guarantee ±0.375% accuracy. Hence, the acceptance criterion of the model is the MAPE of the 0.375% because the reasonable ML-based inference model should be as good as the thermocouples.

Table 25. The Labels in the GOTHIC Code and Corresponded Cells in Control Volumes

Label	Control Volume	Label	Control Volume
TL1s45	Primary Tank	TL12s1	Inner Reflector Inlet
TL2s2	Pump #1 Vertical HP Pipe	TL12s4	Inner Reflector Middle
TL3s2	Pump #1 Horizontal HP Pipe	TL12s6	Inner Reflector Outlet
TL4s4	Pump #1 Vertical LP Pipe	TL13s1	Outer Blanket Inlet
TL5s2	Pump #2 Vertical HP Pipe	TL13s4	Outer Blanket Middle
TL6s2	Pump #2 Horizontal HP Pipe	TL13s8	Outer Blanket Outlet
TL7s4	Pump #2 Vertical LP Pipe	TL14s1	Upper Plenum
TL8s1	HP Lower Plenum	TL15s3	Outlet Pipe
TL9s1	LP Lower Plenum	TL16s5	IHX Primary Shell
TL10s1	Active Core Inlet	TL17s1	IHX Secondary Downcomer
TL10s6	Active Core Middle	TL18s2	IHX Secondary LP
TL10s12	Active Core outlet	TL19s5	IHX Secondary Tube
TL11s1	Control Rod Inlet	TL20s1	IHX Secondary UP
TL11s5	Control Rod Middle	TL21s1	IHX Secondary Outlet Pipe
TL11s9	Control Rod Outlet		

Figure 53 and figure 54 show the model performances of the "B_fl1" and "B_t9_fl16" model. There are several findings that are as follows:

1) In the case of the "B_fl1_mapping" model, both test results of type1 and type 2 satisfy the success criteria. However, in type 3 testing, the inferred temperatures are not acceptable at every control volume except TL17 and TL18 (see table 26).
2) The MAPE tends to be smaller at the control volumes in which the coolant does not flow into the core channels yet, such as the primary tank, pump pipes, LP, and core inlet region when compared to control volumes near the outlet, because the variation range of the outlet temperature is much higher than the inlet temperature.
3) The coolant temperatures at IHX Secondary Downcomer and LP have the smallest MAPE because the difference between the maximum and minimum is 0.025℃ at most.
4) The comparison between the results of the "B_fl1_mapping" and "B_t9_fl16_mapping" models is illustrated in figure 14. The "B_t9_fl16_mapping" model's performance is acceptable only in the type 1 test dataset. Also, even though the average of MAPE in type 3 is smaller than "B_fl1_mapping" model's, the value is still above the success criteria. From this point of view, the "B_fl1_mapping" model is better than "B_t9_fl16_mapping" model (see figure 54).

In conclusion, the function of the ML-based inference model has been successfully extended for mapping the temperature distribution in the reactor system when the random noise from the measurement is 1%. Since the MAPE is affected by the range of the variation along the transient scenario, the temperatures nearby outlet regions show a lower accuracy than temperatures at inlet region control volumes. However, if the mass flow rate at pump #1 is able to be correctly measured, all the temperatures can be approximately estimated without any observations.

Table 26. Average MAPE of the "B_fl1_mapping" Model and "B_t9_fl1_mapping" Model

Average MAPE (%) / Model Name	Type 1 Test Dataset	Type 2 Test Dataset	Type 3 Test Dataset
B_fl1_mapping	1.294E-01	1.516E-01	5.213E+00
B_t9_fl1_mapping	1.213E-01	4.045E-01	8.053E-01

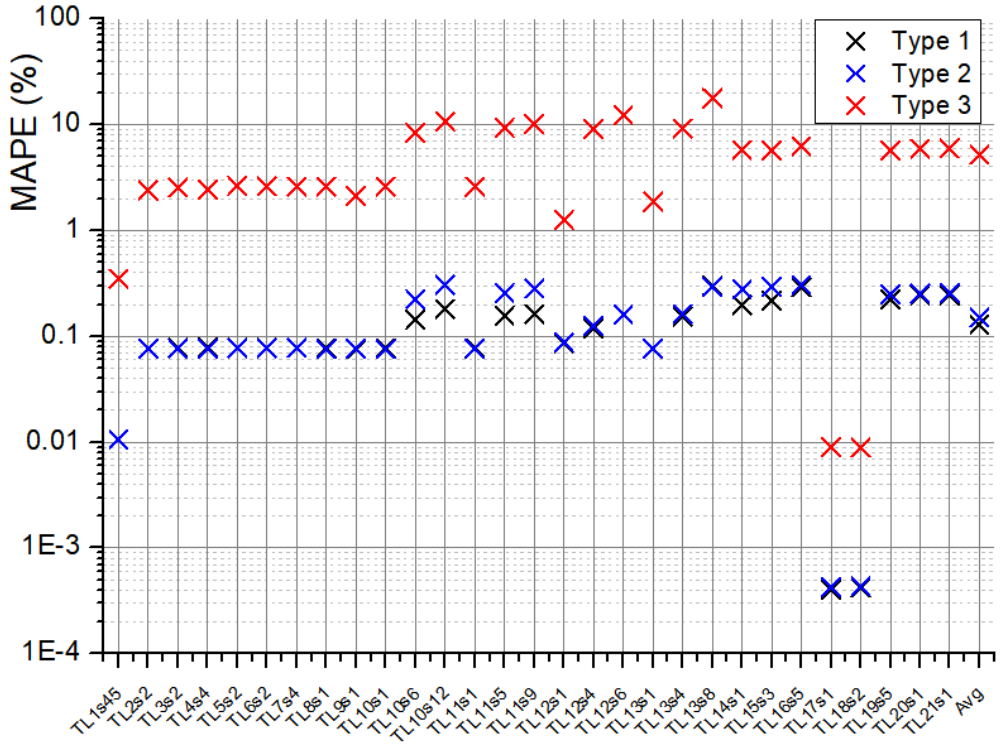


Figure 53. Mapping Results by "B_fl1_mapping" Model

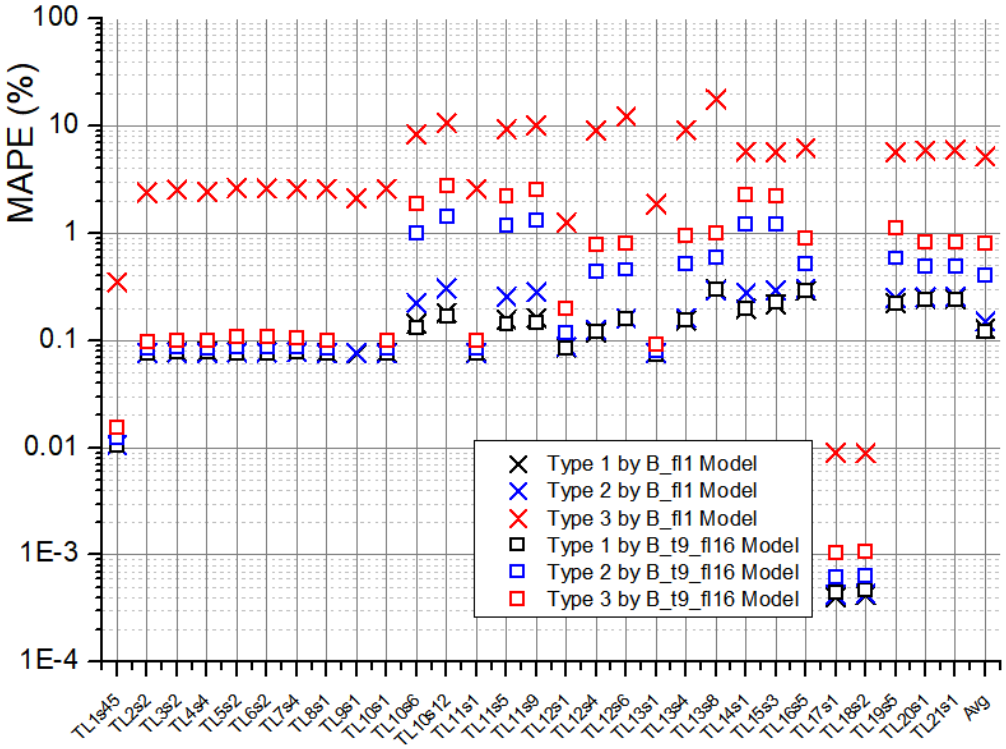


Figure 54. Comparison between "B_fl1_mapping" Model and "B_t9_fl16_mapping" Model

5.3.2. Developing the Faster-than-real-time Inference Model

The ML-based inference model is able to function the faster-than-real-time simulator by dealing with the sequential data. Since the abnormal physical symptoms are time-dependent phenomena during transient, the inferred SSF values based on the current measurement would represent the past plant damage states. From this perspective, the diagnosis sphere covers the short-term forecast. Thereby, the expression of 'prediction' is more proper rather than 'inference' in the task. In the case study, the one-second-later SSF is predicted from the current measurement and operation history because each time step is one second. The figure 55 illustrates the modified RNN structure to develop the faster-than-real-time model.

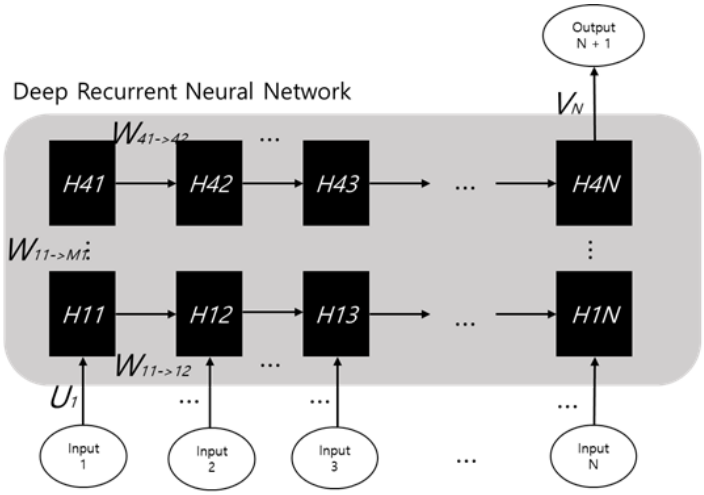


Figure 55. The RNN Structure for the Faster-than-real-time Model

Table 27 and figure 56 show the MAPE of the real-time model and faster-than-real-time (ftr) model in transient C. There are several findings that are as follows:

1) A faster-than-real-time model is able to predict the next step's FCL temperature by using the current measured data. The MAPE of the predicted SSF is less than 1% for every type of test dataset (see table 27 and figure 57).
2) In the case of the "fl1" model, the MAPE of the faster-than-real-time model is larger than the MAPE of the real-time model. However, the faster-than-real-time model's performance is better than the real-time model in the "t9_fl16" model case, because these are totally different and independent models (see figure 56).

Table 27. MAPE of the FCL Temperature from the Real-time Model and Faster-than-real-time Model

MAPE (%) / Model	Type 1	Type 2	Type 3
C_fl1	1.752E-01	2.103E-01	4.001E-01
C_fl1_ftr	1.694E-01	2.517E-01	5.510E-01
C_t9_fl16	1.992E-01	3.024E-01	7.698E-01
C_t9_fl16_ftr	1.932E-01	2.522E-01	5.716E-01

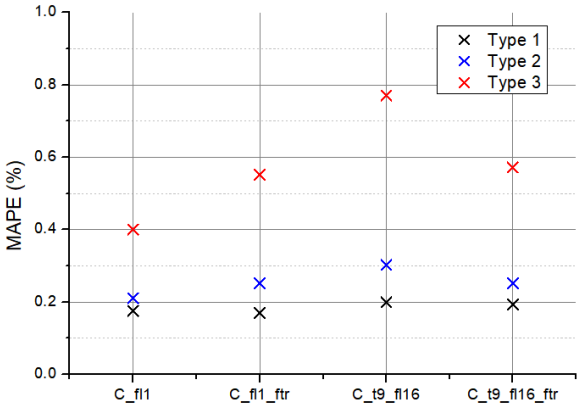


Figure 56. MAPE Comparison between Real-time Model and Faster-than-real-time Model

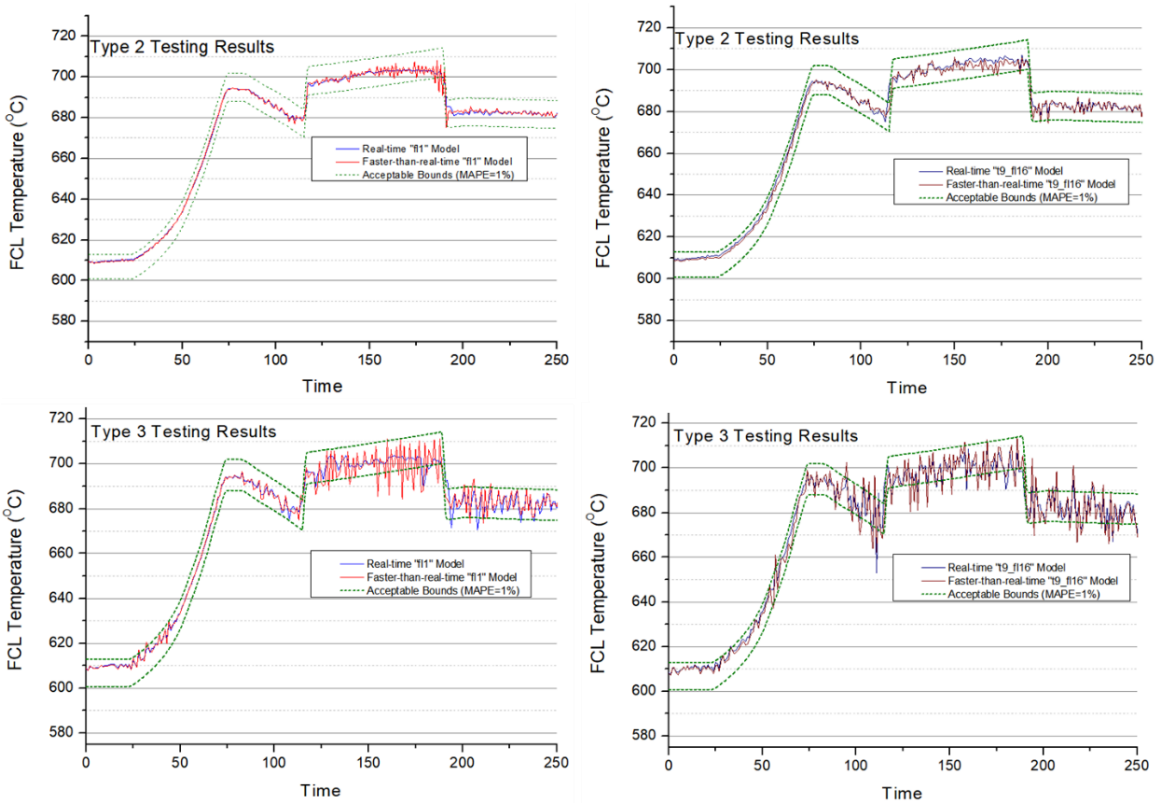


Figure 57. Type 3 SSF Inference Results by "fl1" Model (Left) and "t9_fl16" Model (Right)

5.3.3. Summary

The ML-based inference model is able to extend the diagnosis range in both coverage and time span. Firstly, by using a mass flow rate data, all the temperatures in the reactor system are inferred within 0.375% when the random noise of variable is less than 1%. The MAPEs in the outlet control volumes tend to be higher than others because the magnitude of the error mainly depends on the variance in the scenario. Therefore, for improving the accuracy, splitting into two groups, inlet temperatures and outlet temperatures, is suggested in the normalization process. Secondly, the faster-than-real-time inference model is developed to predict the next time step's FCL temperature. The diagnosis sphere can be extended to short-term prognosis to deal with the sequential issue in virtue of the ML-based inference model's capability. In conclusion, the extendibility of the SSFIM based on the RNN is realized in several ways.

5.4. Identifying the Inference Model Ability by Extrapolating the SSF

In the research, the inference ability test aims to figure out whether or not a trained model is able to be applied to out of range scenarios with acceptable results. For the first test, the test dataset's head events are the same as the training dataset's, but it passes the bounds of the range. Meanwhile, for the second test, the test dataset and training datasets do not share the head events in scenario. For instance, it evaluates whether the model developed for transient A can reasonably infer the SSF on transient B or C. Through the case study, the inference ability of the ML-based model would be identified.

5.4.1. Extrapolating the SSF in an Extended Similar Transient

The extrapolation test is implemented to identify whether the ML-based model is able to infer the SSF when pump #2's operation state is beyond its training range or not. The independent variable for the training model is a range of ramping speed, then the distance of extrapolation would be determined. Table 28 describes the scenarios of training datasets for the model and test datasets. For instance, from the distance perspective, the range of training datasets for the "B_fl1_ext_105_130" model is relatively near to the test dataset B-53 compared to that for the "B_fl1_ext_105_120" and the "B_fl1_ext_105" models. Whereas, the distance from the trained data of "B_fl1_ext_150" model to the transient B-11 is the largest among the test sets. By using

these sets, the inference ability would be estimated when pump #2's ramping speed does not lie within a range of the training dataset.

Table 28. Scenario Description of Trained Model and Test Dataset

Model Name	Trained Scenario Description		
	Pump #1	Pump #2	Set Point (°C)
B_fl1_ext_105	Coast Down to 50%	Ramping up [105%]	[672 ~ 690]
B_fl1_ext_105_120	Coast Down to 50%	Ramping up [105% ~ 120%]	[672 ~ 690]
B_fl1_ext_105_135	Coast Down to 50%	Ramping up [105% ~ 130%]	[672 ~ 690]
B_fl1_ext_120_150	Coast Down to 50%	Ramping up [120% ~ 150%]	[672 ~ 690]
B_fl1_ext_135_150	Coast Down to 50%	Ramping up [135% ~ 150%]	[672 ~ 690]
B_fl1_ext_150	Coast Down to 50%	Ramping up [150%]	[672 ~ 690]
Test Dataset	Tested Scenario Description		
	Pump #1	Pump #2	Set Point (°C)
B – 11 (Type 1)	Coast Down to 50%	Ramping up to 110%	681
B – 53 (Type 1)	Coast Down to 50%	Ramping up to 140%	681

Table 29 and 30 show the extrapolation results. There are several findings that are as follows:

1) The SSF is successfully inferred when the distance between the training dataset and test dataset is close. For example, even though the model is trained to know that the maximum capacity of the pump speed is 135%, it can correctly infer the FCL temperature with 0.21% accuracy when the pump is operated at 140%, over the known maximum capacity. However, the further the test dataset is from the training dataset, the worse its accuracy.
2) When the distance between pump speeds is 20%~25%, the inferred SSF is acceptable to be used. Thus, 20%~25% would be treated as a safe range to extrapolate the SSF for the transient B case.
3) However, the model performances become poor above 35% distance. As we can see from the result of B-11 in table 30, the inferred SSF by "B_fl1_ext_150" is much lower than the acceptable level. In other words, the model overestimates the effect of 110% of the pump speed. Similarly, in the test result of B-53, the "B_fl1_110" model under-estimate the impact of 140% of ramped-up speed. Therefore, the range of head events in scenario

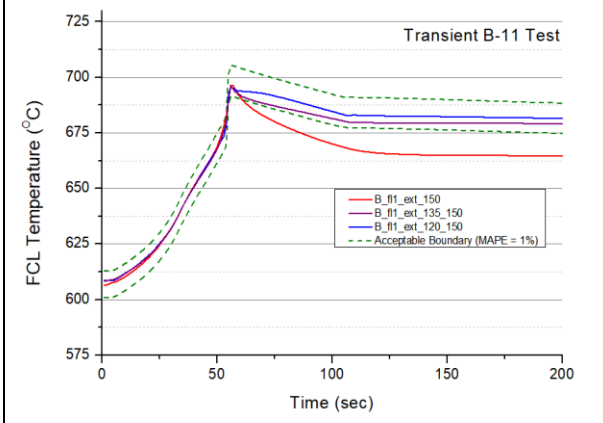
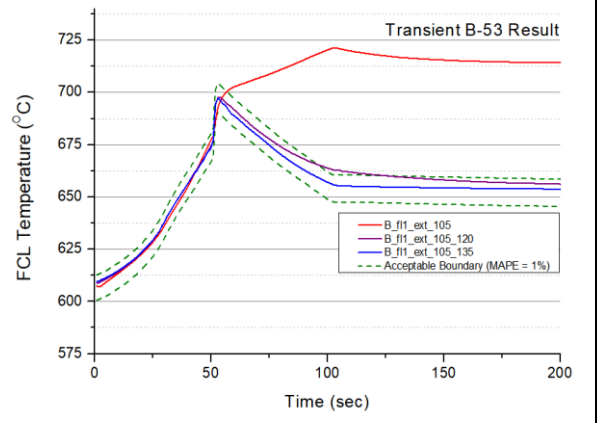
matters for training the ML-based model. The ML-based model infers the SSF based on what they have learned.

Two types of extrapolation tests have been implemented in the case study: acceptable-field extrapolation and unacceptable-field extrapolation. In the research, an acceptable-field is defined as a zone in which the extrapolation result is acceptable, likewise, an unacceptable-field indicates a zone where the SSF cannot be reasonably extrapolated by the model. The ML-based SSFIM proves its inference ability for an acceptable-field in which the distance between training and test datasets is less than 25% in the case study. But, in the case of an unacceptable-field, the model has failed to infer the FCL temperature due to underestimation or overestimation of the impact of the pump operation. The range of head event matters for training the ML-based model.

Table 29. MAPE in Extrapolation Test

MAPE	B-11 (110%)	B-53 (140%)
B_fl1_ext_105	-	6.013e+00
B_fl1_ext_105_120	-	5.838e-01
B_fl1_ext_105_135	-	2.090e-01
B_fl1_ext_120_150	1.501e-01	-
B_fl1_ext_135_150	4.805e-01	-
B_fl1_ext_150	1.719e+00	-

Table 30. SSF Inference Results in Extrapolation Tests

Transient B -11 Test	Transient B -53 Test
	

5.4.2. Extrapolating the SSF with Different Scenarios

The extrapolation test is implemented in another field: different-field. Table 31 demonstrates a trained model and test dataset. From the head events in scenario perspective, for example, since test data A-53 is a totally different scenario compared to the training data of transient B or C, the different-field is named for the extrapolation test. As mentioned above, the SSFIM has failed to achieve the satisfactory level in the unacceptable-field. Thus, the hypothesis that the model performances would be poor may be obviously improved, but there are some learning points from the results for developing the ML-based model.

Table 31. Descriptions and Relationship of the Trained Model and Test Dataset

Model Name	Trained Scenario Description	
	Pump #1	Pump #2
A_fl1	Coast Down [0% ~ 45%]	-
B_fl1	Coast Down [50%]	Pump #2 Speed [105% ~ 150%]
C_fl1	Coast Down [50%]	Pump #2 Ramping up and down 1st action: [50%] 2nd action: [80% ~ 120%] 3rd action: [80% ~ 120%]

Test Dataset	Number of Variables	Pump Status Description	
		Pump #1	Pump #2
Transient A - 53	1	Coast Down to 22.58%	Constant (100%)
Transient B - 53	2	Coast Down to 50%	Ramping up (140%)
Transient C - 37	4	Coast Down to 50%	Ramping down (50%) → Ramping up (100%) → Ramping down (90%)

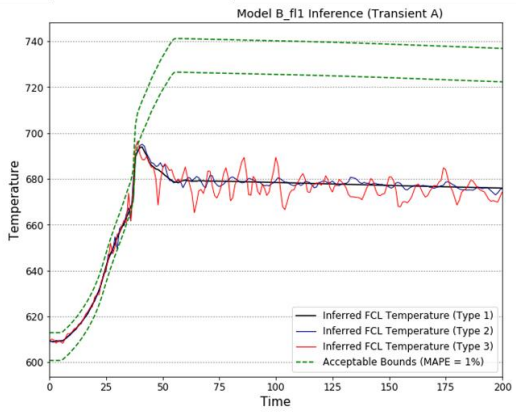
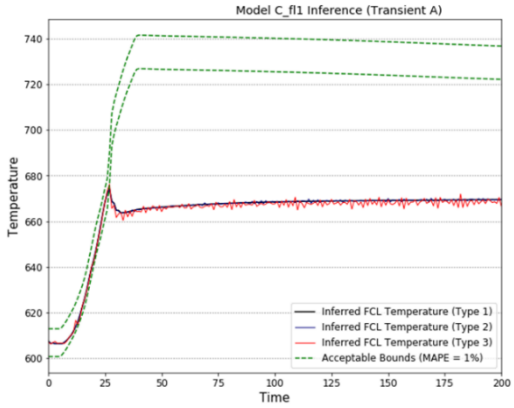
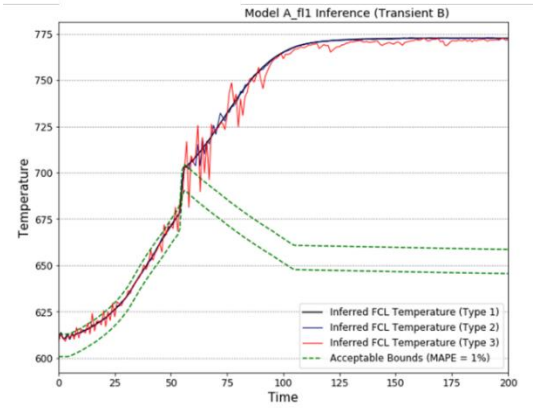
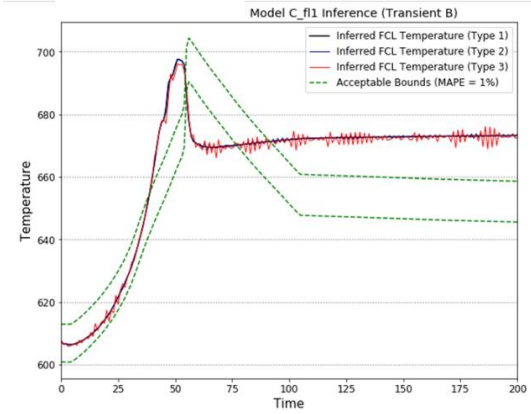
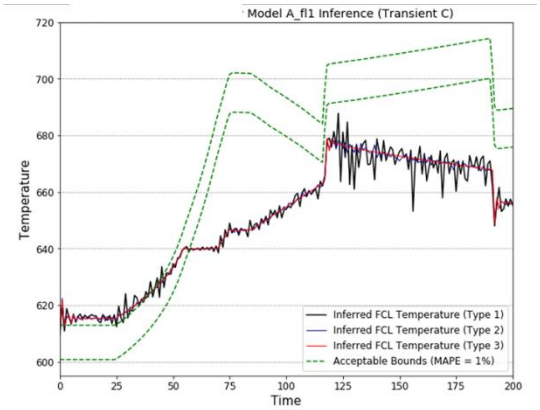
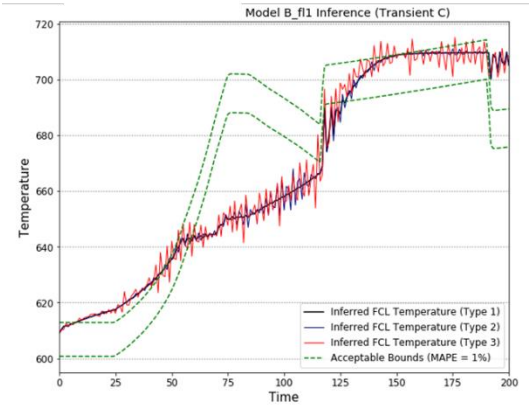
Table 32 shows the inference ability test results. There are several findings that are as follows:

1) A trained ML-based SSFIM fails in an inference ability test. A trained model on a certain type of transient case is not able to be applied to other types of the test datasets.
2) Models B and C have not learned the transient when pump #2's speed has not changed. Hence, the inferred FCL temperatures from the models show incorrect curves.
3) When the B-53 is tested by model A, the inferred FCL does not decrease because pump #2's action is not considered in transient A. Whereas, model C analogously infers the FCL

temperature pattern, but the value in stabilized-states is overestimated, because 140% of pump #2's speed are uncharted operation status for model B.

4) The FCL temperature in the C-37, the most complex transient with four head events, is poorly inferred by models A and B. Since both models have not learned about the ramping down of pump #2, dramatic peak at 70 seconds cannot be recognized. The mass flow rates in transients B and C are close to each other in converged value, hence, the inferred FCL temperature is not preposterously far from the actual temperature. On the contrary, because the mass flow rates in transient A are much lower than the others, the inferred FCL temperature is underestimated.

Table 32. Inference ability Test Results

Test Dataset	Transient A -53
B_fl1 Model	C_fl1 Model
	
Test Dataset	Transient B - 53
A_fl1 Model	C_fl1 Model
	
Test Dataset	Transient C - 37
A_fl1 Model	B_fl1 Model
	

A lesson from the different-field extrapolation test is that a trained model from a certain transient scenario is not applicable to other types of the transient. It is important that the head events should be deliberated to cover the scenarios for the training data generation. Also, in the operational workflow, the outlines of the event should be identified to use the ML-based SSFIM. For instance, in the LOF due to pump coast-down, to use the transient "A" model, pump #1's operation state is not normal and pump #2 is in normal condition. The "A" model should be changed to the "B" model when an operator makes a decision to ramp up pump #2. On the other hand, if pump #2 has failed after pump #1's coast down, the "C" model should be selected to infer SSF. From this perspective, the AI-guided diagnosis model should include the function of the head event identification to provide prior information for the ML-based SSFIM.

5.4.3. Summary

Through the case studies, the limitations of the ML-based SSFIM are shown and corresponding solutions are suggested. The study aims to identify the inference ability because the range of operating conditions is not perfectly specified. The ML-based model is not an almighty technique to identify the plant damage states when either the scenario in an encountered transient is different from the training dataset's or its range is far beyond the range of the training dataset's. Hence, in order to develop the powerful ML-based SSFIM, it is recommended that various types of transient scenarios, which are derived by a wide range of actions or multiple issue spaces, should be prepared to train the model. Additionally, the importance of the head event identification function in the AI-guided diagnosis model is enhanced.

5.5. Summary

Through the 4 case studies, the ML-based SSFIM is developed and utilized to identify the plant damage states during a transient. Each case study is connected to develop competent SSFIM and study the model's extendibility by establishing its objective to achieve the function of a diagnosis model.

For building the ML-based model, the RNN with learning rate scheduler of RLRP is a proper algorithm and technical component used to infer the FCL and CL temperatures. Despite the fact that some random noises are presented in the physical variables, the ML-based SSFIM infers the SSF within 1% of the MAPE by using only a mass flow rate at pump #1. The case study result

implies that temperature is a sensitive variable when there is noise in the measured data, whereas, the mass flow rate is a powerful variable against the noises used to infer the SSF. For the successful ML-based SSFIM performance, selecting the powerful features is a consequential issue.

By using the adapted ML-based SSFIM, the range of the inferred information is extended to broaden the view of the diagnosis. Firstly, the ML-based inference model allows the temperature distribution in the reactor system to be mapped from sensory data of a mass flow rate. Secondly, the diagnosis realm is extended to cover the short-term forecasting in the aspect of the faster-than-real-time diagnosis by predicting the one second later SSF. The capabilities of the ML algorithm would enable the SSFIM to be utilized for further studies in the diagnosis research field.

For the development of the ML-based SSFIM, it is suggested that a variety of transient or accident scenarios are used to train the model. Since the ML is a useful predictive modeling method, unless it meets entirely new cases, from the “No Free Lunch" perspective, what is important for the model performance is how many scenarios and its range are covered in the training dataset. Also, the function of head event identification is necessary in the AI-guided diagnosis model. Thus, the problems should be identified in the data generation process throughout a wide range of the transient scenario.

CHAPTER 6. CASE STUDY ANALYSIS

In this section, based on the case study results, the features of the ML-based inference model are analyzed. It includes a comparison between knowledge-based simulation and data-driven model performance; discussion of the inference ability and limitation of the ML-based model; and expected potentials to provide a meaningful basis for advanced tasks for not only the AI-guided diagnosis model but also the management and control system.

6.1. Calculation and Inference

6.1.1. Calculation by using Knowledge-based Approach Model

The SSF data is produced by running GOTHIC code. The conservation equations for mass, energy, and momentum are solved by GOTHIC_S (solver) program to calculate the physical variables in accordance with the operating conditions [51]. The SSF, which is an existent, but not easily measurable physical variable, is obtained by knowledge-based simulation code. In this study, the deterministic code is assumed as the best-estimate code, so the numerical results from the simulation are equivalent to the actual values. Since the SSF level, such as FCL temperature or CL temperature, is regarded as a solid thermal conductor in the GOTHIC code, it is determined based on thermal hydraulics by solving conservation equations for mass conservation, energy conservation, and momentum conservation [52]. Following equations and assumptions are extracts from the GOTHIC technical manual (version 8.2).

Energy conservation for solid thermal conductor considers thermal conductivity and boundary source from the wall. The energy equation for the solid conductor is given by

$$\int_{V_{cn}} \rho_{cn} c_{p,cn} \frac{\partial T_{cn}}{\partial t} dV_{cn} = \int_{A_i} D_{cn}^{e} \vec{\nabla} T_{cn} \cdot \vec{n} dA + \int_{A_b} s_{cn}^{e} dA \quad (22)$$

where the subscript *cn* means a specific conductor, ρ is the density, c_p is the specific heat, T is a local temperature of the conductor, V_{cn} is the volume of the conductor, D_{cn}^{e} is the diffusion coefficient (conductivity), s_{cn}^{e} is the energy source per unit wall area, A_i and A_b are the surface area of V_{cn} internal to the conductor and the external bounding surface area in which the conductor meets with the fluid phases, respectively. The energy balance is reduced for flat plate geometries with two-dimensional condition:

$$\rho_{cn} c_{cn} \frac{\partial T_{cn}}{\partial t} = \frac{\partial}{\partial x}\left(k_{cn} \frac{\partial T_{cn}}{\partial t}\right) + \frac{\partial}{\partial z}\left(k_{cn} \frac{\partial T_{cn}}{\partial z}\right) + Q'''_{cn} \quad (23)$$

where the c_{cn} and k_{cn} are specific heat and the conductivity, respectively, which depend on space and the local temperature (T_{cn}). Q'''_{cn} is the local internal heating rate. The energy source from the wall to the fluid, which is one of the boundary conditions, includes convection and radiation heat transfer, and condensation. The total heat transfer from the wall to the fluid (Q_w) is

$$Q_w = Q_{cond} + Q_{conv_i} + Q_{conv_l} + Q_{rad} \quad (24)$$

where Q_{cond} is the latent heat released by condensation at the conductor surface. Q_{conv_i} and Q_{conv_l} are the convective heat flux between the saturated film and superheated vapor and between the liquid and the conductor surface, respectively. Q_{rad} is the radiant heat flux. The total heat transfer is used for one of the boundary conditions of energy balance. In the GOTHIC EBR-II model in which coolant is liquid state sodium, the liquid phase fluid side convective heat rate ($Q''^{n+1}_{conv_l}$) is given by

$$Q''^{n+1}_{conv_l} \cong H_{conv_i}(T_w^{n+1} - T_l^n) + \frac{dQ''_{conv_i}}{dT_w}\left(\frac{dT_w}{dT_l} - 1\right)(T_l^{n+1} - T_l^n) \quad (25)$$

where H_{conv_i} is the liquid convective heat transfer coefficient and the superscript *n* and *n+1* indicates the time levels.

As the equations show above, to calculate the FCL and CL temperature, the liquid state fluid temperature with time is a significant term. In the same manner, the mass flow rate would also become another considerable factor to affect the conductor temperature. The mass flow rate is given by

$$F_l = A u_l \rho_l \alpha_l \quad (26)$$

where u_l and A refer to liquid velocity and area, respectively. α_l is donor cell phase fraction from the donor-acceptor scheme which deals with fluid volume [52]. In the case of an EBR-II reactor, α_l = 1, because the reactor type is a pool-type reactor in which many reactor core-related systems are submerged under the sodium coolant.

The mass flow rate, which would be measured at pump flow path or core inlet, affect the Reynolds number, the ratio of inertial to viscous forces in a fluid. The Reynolds number is given by

$$Re = \frac{D_h \rho_l |u_l|}{\mu_l} \quad (27)$$

where D_h and μ_l refer to hydraulic diameter and dynamic viscosity, respectively. The Nusselt number, which indicates the ratio of convective to conductive heat transfer at the surface, is a function of the Reynolds number and the Prandtl number. The Nusselt number is

$$Nu = f(Re, Pr) = \frac{hL}{k} \tag{28}$$

where L and k are the characteristic length and the thermal conductivity, respectively. In the equation, h is the convective heat transfer coefficient, which is also marked as H_{conv_i} in liquid phase.

By using the knowledge of the finite volume numerical method with the conservation equations, the SSF is calculated.

6.1.2. Inference by using Data Driven Approach Model

The ML model with DNN, which is combined by a number of logistic regressions, learns the patterns to identify the relationship between inputs and outputs by updating the weights. The RNN treats the time-dependent data problem in which outputs at a specific time step are predicted from corresponding sequential input data. Then, based on the case study results in Chapter 5.2., we have the question of why a specific variable would be a dominant feature in inferencing the output more than other variables. There are two hypotheses to answering this question based on two different approaches: 1) there is a particular correlation in trends between the SSF and the mass flow rate or 2) a specific variable has a distinct feature compared to other physical variables as an input for the RNN.

- Physical Variable Based Analysis by using the PCC

A main topic of the first hypothesis is the correlation between inputs and outputs. As we can see the results in Chapter 5.2., the mass flow rate in pump #1 is the most robust variable and the essential input element needed to build the robust model even though the random noise is 5%. In other words, the mass flow rate might have a particular correlation with the SSF compared to other physical variables. Thus, analyzing the relationship between input variables and output variables would confirm the hypothesis.

A Pearson Correlation Coefficient (PCC) indicates the degree of the linear relationship between two variables with a number between -1 and 1. Since it does not signify causation between two variables, a cause and effect conclusion cannot be drawn from the PCC analysis.

Mathematically, the PCC is the covariance of the two variables divided by the multiplication of overall standard deviations. The PCC (r_{xy}) is given by

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}} \quad (29)$$

where x_i and y_i are the individual sample variables' points indexed with i. n is sample size. $\bar{x}$ and $\bar{y}$ represent sample mean of variable x and y, respectively. On the basis of the absolute value of r_{xy}, the strength of relationship is verbally described according to the Evans guidance (Evans, 1996) in Table 33 [53]. Higher PCC refers to stronger relationship. Similarly, the closer a value is to a negative one (-1), the stronger the two variables are negatively correlated.

Table 33. Relationship Strength Interpretation of PCC Absolute Values

Coefficient	Verbal Description
0 – 0.19	Very Weak
0.2 – 0.39	Weak
0.4 – 0.59	Moderate
0.6 – 0.79	Strong
0.8 – 1	Very Strong

As mentioned, above, the simulation results from the GOTHIC are assumed to be the actual physical phenomena in the reactor. Since the physical phenomena due to transient occur in a confined system, the trends of the physical variables and of the SSF should be correlated with each other. The result of the PCC analysis in transient C is shown in figure 58. The findings from the result are as follows:

1) The trends between the FCL and the CL are almost the same because these are the SSF.
2) The PCC between connected variables, such as coolant temperature at UP and outlet pipe or HPLP and inlet mass flow rate, are one or close to one.
3) The SSF is highly correlated with the coolant temperature at UP and outlet temperature; coolant temperature at IHX secondary system; and mass flow rate at pump #2's path in a positive way. Whereas, it is negatively correlated with the coolant temperature at IHX

primary system and IHX secondary downcomer; pressure at HPLP; and mass flow rates in pump #1's path and an inlet path.

4) In conclusion, the "fl1" does not have a distinctive correlation with the SSF.

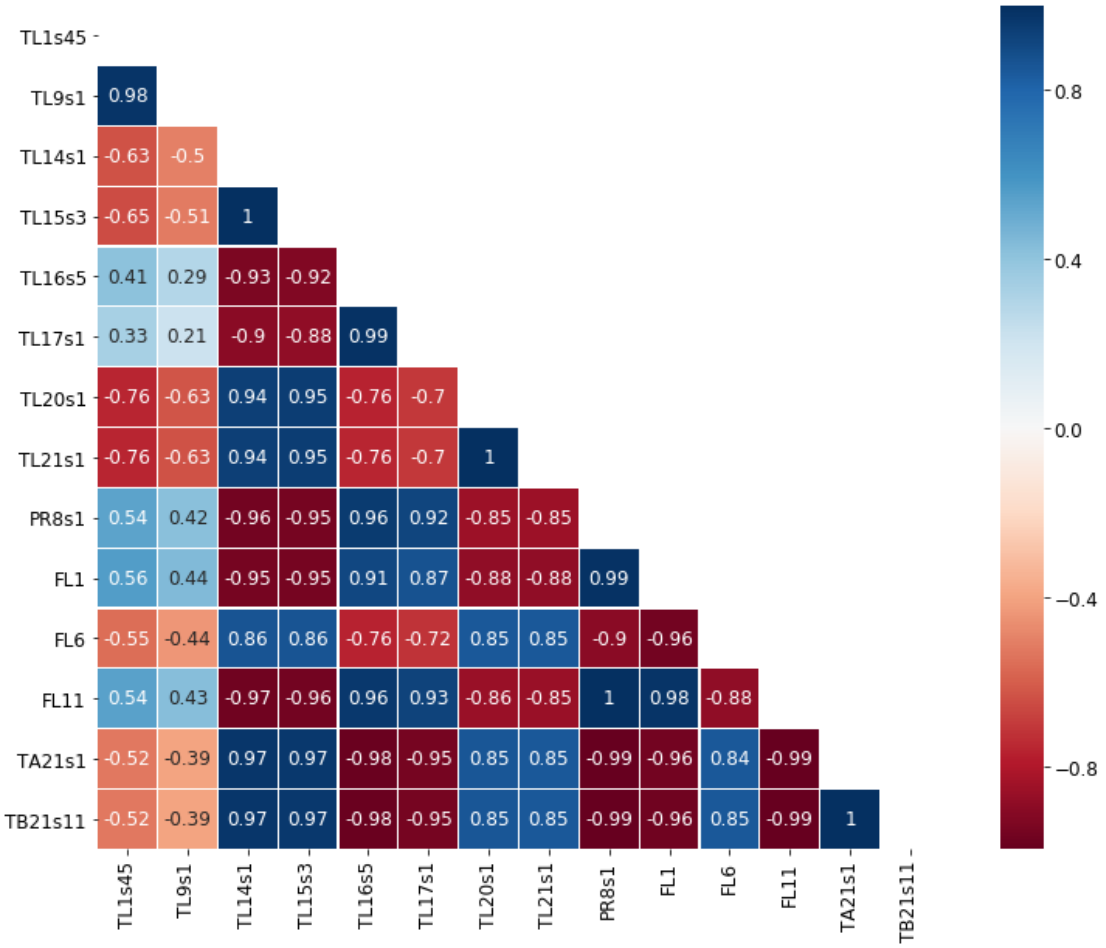


Figure 58. Average PCCs in Transient C

- Input Data Based Analysis by identifying Change Range in Normalized Scale

The second hypothesis is that the "fl1" has a distinct feature not as a physical variable but as an input variable for the RNN. Since the normalized value is used to train the network, the magnitude of the patterns is determined according to the physical variable types. Table 34 demonstrates the length of the change range in a normalized scale for each transient.

Table 34. Average Lengths of Change Range in Normalized Scale

Physical Variables	Label	Control Volume	Transient A	Transient B	Transient C
Coolant Temperature	TL1s45	Primary Tank	1.620E-03	1.333E-03	1.800E-03
	TL9s1	LPLP	7.700E-03	6.633E-03	8.775E-03
	TL14s1	UP	2.798E-01	1.696E-01	1.984E-01
	TL15s3	Outlet Pipe	2.708E-01	1.636E-01	1.921E-01
	TL16s5	IHX Primary Shell	1.218E-01	7.390E-02	7.638E-02
	TL17s1	IHX Sec Downcomer	1.000E-04	1.000E-04	1.000E-04
	TL20s1	IHX Sec UP	7.020E-02	5.393E-02	6.200E-02
	TL21s1	IHX Sec Outlet Pipe	7.018E-02	5.393E-02	6.200E-02
Pressure	PR8s1	HPLP	6.056E-01	5.399E-01	5.532E-01
Mass Flow Rate	FL1	Pump 1	1.528E+00	1.605E+00	1.069E+00
	FL6	Pump 2	4.856E-01	7.557E-01	5.289E-01
	FL11	Inlet	4.601E-01	3.915E-01	4.038E-01

The findings from the result are as follows:

1) The change range lengths of the normalized coolant temperature are much smaller than other variables. Because there is a large gap between LP temperature and UP temperature or outlet temperature, the distance between the lower bound and upper bound is larger than other variables. The “t9” model shows poor model performance because its changes in transient are too small to be learned.
2) Due to the small lengths of range, the temperature variables are too sensitive to random noises to infer the SSF. That is why temperature-only models have failed to infer the SSF in type 2 test.
3) Otherwise, the change range lengths of the normalized mass flow rate and pressure are larger than the coolant temperatures’ ones. Therefore, the models which contain the variables are able to satisfy the success criteria in type 2 test. However, the random noise in the type 3 test dataset is too large for them, except the “fl1.”
4) In particular, the range of the mass flow rate at pump #1, which is the most robust variable, is widest among the suggested variables because pump #1’s coast-down induces the reverse flow. In virtue of that, the normalized range of the “fl1” is set to [-1, 1], and the average length of the normalized range is above 1. The variable becomes the most powerful and robust input data against the random noise.

5) In conclusion, the length of change range in a normalized scale is the dominant factor used to train the ML-based SSFIM.

The hypothesis established based on input data analysis has been confirmed. The RNN algorithm recognizes the trends of the input variables rather than the time-dependent physical variables. However, the dynamic patterns of the input variables signify the physical variables generated by the thermal-hydraulic code. The following section discusses the GOTHIC code and the ML-based SSFIM.

6.1.3. Comparison between GOTHIC and ML-based SSFIM

Both GOTHIC and ML-based SSFIM are deterministic codes used to compute the specific physical variables, although the approaches are different (see figure 59). The GOTHIC calculates the physical variable by a knowledge-based approach, meanwhile, the other one infers the output data from the well-trained ML algorithm built by a data-driven approach. In order to calculate the SSF, thermal conductors in the GOTHIC model of the EBR-II reactor, there are a number of essential components: extensive knowledge and experimentation, initial conditions, boundary conditions, and a variety of related information, such as tabular functions, physical parameters, system geometry, and so on. In contrast, a sufficient amount of reliable data and an appropriate algorithm with some advanced modules are required to design the ML-based model. Owing to the prerequisites in each model, the upgrade processes are different. For the version upgrade of the simulation code, additional thermal-hydraulic features would be modeled based on theories [51], but, in the case of the ML model, additional training on newly secured datasets would be executed to update the model. In the aspect of the development and upgrade process, building the ML-based model may be simpler than designing the thermal-hydraulic simulation code.

GOTHIC and ML-based SSFIM have developed to achieve each objective. The GOTHIC code is applicable to design the reactor system or analyze the safety and operating system [51]. Whereas, the ML-based SSFIM is designed to diagnose the plant damage state by identifying the core's physical states. The ultimate goal of a model determines the approach, development method, and features. Although the SSFIM is able to rapidly predict the unknown information in real-time or faster-than-real-time, it is not possible to analyze the risk because it cannot investigate the relationship between inputs and outputs. Thus, two deterministic codes should be used in a

complementary way rather than in a competitive way. The research suggests for the complementary way that the best-estimate code provides enough and reliable training data to train the ML algorithm which will be used to diagnose the reactor. From this perspective, data generation to support the ML model becomes one of the major purposes of the GOTHIC. Table 59 summarizes the comparison between two models in the aspect of features.

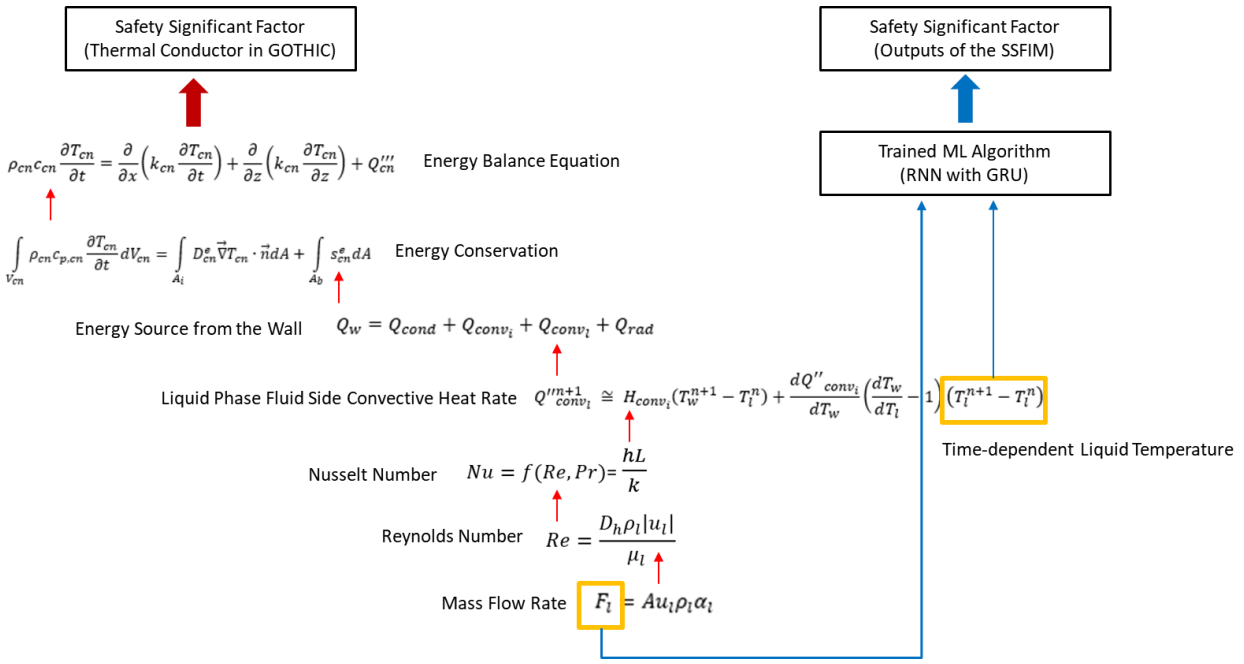


Figure 59. Safety Significant Factor Calculation by using the Measurable Physical Variables

Table 35. Characteristics of the GOTHIC and ML-based SSFIM

	GOTHIC	ML-based SSFIM
Approach	• Deterministic Approach • Knowledge-based Approach	• Deterministic Approach • Data-driven Approach
Prerequisite	• Extensive Knowledge and Experimentation • Initial Conditions • Boundary Conditions • A variety of related information	• Reliable Data • A large amount of Training Data • Appropriate algorithm with advanced modules
Upgrade	• Upgraded by updated thermal-hydraulic models based on new experience or knowledge	• Upgraded by training the new data
Purpose	• Design and licensing • Safety and operating analysis • Data generation	• Real time (or faster-than-real-time) diagnosis

6.2. Inference Ability and Limitations

Through case study 4, not only the inference ability but also the limitations of the ML-based SSFIM are estimated. From the test results, the acceptable-field and unacceptable-field are determined for extrapolating the SSF (see table 36). The inference test aims to identify whether the SSFIM is applicable to unknown scenarios which are caused by being out of the range of head event or not. In the extended similar scenarios, the SSF is acceptably inferred in an acceptable-field where it includes a ramping up speed of up to 25% in transient B. However, in the far field from the trained ranges, the SSF inference is failed by overestimation or underestimation because the model has not learned such a level of impact by pump #2's operation. For the same reason, the different-field extrapolation result obviously reveals the limitation of the ML-based model. Table 37 describes the test cases of transients A, B, and C. It seems that there are some common events among the transients. However, the range of the final velocity in transient A is from 0% to 45%, and pump #2's ramping up speed in transient B varies from 105% to 150%. To be brief, the head event is overlapped, but the details in the common event are barely overlapped. For example, the only overlapped head event between transient B and C-37 is pump #1's final speed. The other three head events are not shared between them. Finally, the result shows that a trained model from a certain scenario is not useful for other types of the scenarios.

Table 36. Acceptable-field and Unacceptable-field Determination by Inference Ability Test

Model Name	Training Data Pump #2 Ramping up Speed	Test Data Pump #2 Final Speed	Inference Ability
B_fl1_ext_105	[105%]	140%	Unacceptable
B_fl1_ext_105_120	[105% ~ 120%]	140%	Acceptable
B_fl1_ext_105_135	[105% ~ 130%]	140%	Acceptable
B_fl1_ext_120_150	[120% ~ 150%]	110%	Acceptable
B_fl1_ext_135_150	[135% ~ 150%]	110%	Acceptable
B_fl1_ext_150	[150%]	110%	Unacceptable

Table 37. Example of Different-field based on Inclusion Relationship between Transients

	Number of Head Event	Description	
		Pump #1	Pump #2
Transient A	1	Coast Down [0% ~ 45%]	-
Transient B	2	Coast Down [50%]	Pump #2 Speed [105% ~ 150%]
Transient C	4	Coast Down [50%]	Pump #2 Ramping up and down 1st action: [50%] 2nd action: [80% ~ 120%] 3rd action: [80% ~ 120%]

Different-field	Transient A and B	Transients A and C	Transients B and C
Common Events	Pump #1 Coast Down	Pump #1 Coast Down	Pump #1 Coast Down Ramping up
Details in Common	-	-	Pump #1 Final Speed

Both the range and various types of head events are considerable factors to generate the training data. There are two options to solve the different-field model issue for SSF inference. The first one is developing an integrated model by training all the data in one model. The other one is that the head event, such as component failure or success, is identified before selecting the diagnosis module. In the AI-guided diagnosis system information flow, the component state signals would be extracted to identify the SSC availability. Then, it can also be used to identify the head event to convey the information to the operator for deciding which SSFIM would be selected to properly diagnose the plant damage state. Figure 60 illustrates suggested information flow to select the diagnosis module.

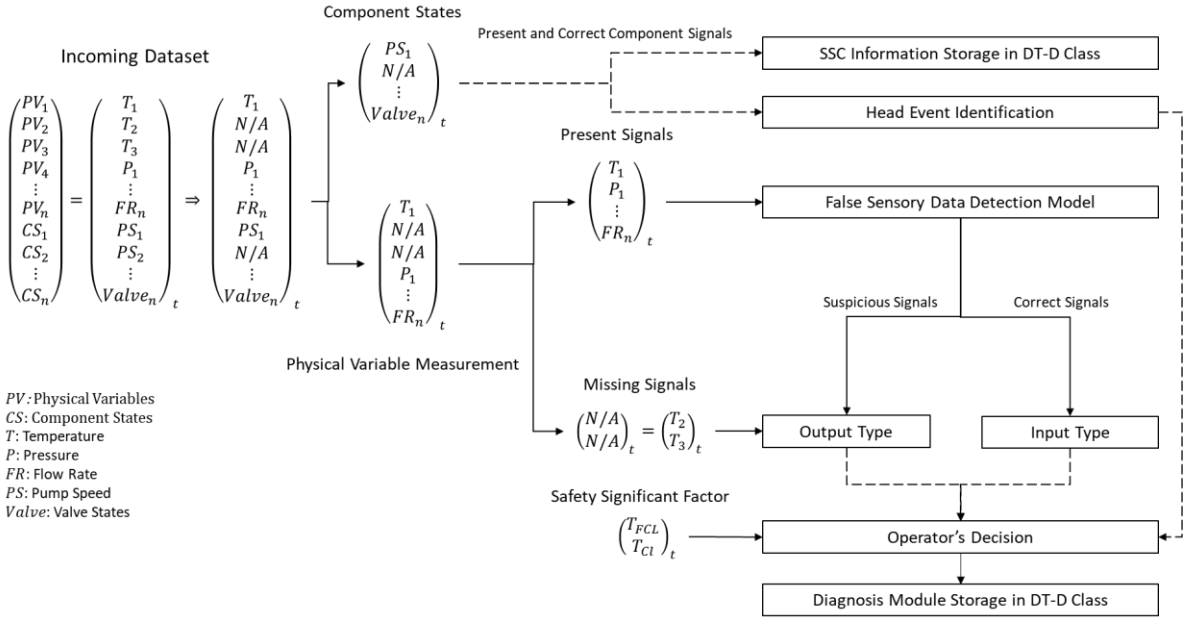


Figure 60. Suggested Information Flow to Select the Diagnosis Module from the DMS in DT-D Class

6.3. RNN Capabilities for Advanced Tasks

6.3.1. Potential Usage for Dominant Event Identification

Through case study 3, the ML-based SSFIM's capability is proved by extending the inferred information range in aspect of both diagnosis coverage and time span. The ML-based inference model may support the development of an AI-guided management and control system or another ML-based diagnosis model by providing the necessary virtual environment or information.

The ML-based inference model offers the mapping data to identify the abnormal physical phenomena in the reactor system. Previously, CNN application to diagnose the abnormal status in the LWR system has been studied. The CNN classifies the abnormal conditions by the leakage of coolant and charging line according to the two-dimensional characteristic map [54]. In the study, 6,720 analog data are used for mapping the data. However, practically, the sensors cannot be installed all over the place in the system to take a snapshot of the 6,720 physical variable data. Therefore, the ML-based inference model may be utilized to provide the inferred physical variables to classify the event type depending on the abnormal physical status. The "B_fl1" model infers temperatures in 29 control volumes' cells. Some of them are illustrated in figure 61. The

ML-based model is expected to provide a two-dimensional physical variable map to identify the dominant event that causes the abnormal status (see figure 62). With observations of only a part of the information, the unobserved physical phenomena are mapped to analyze the transient or accident progression.

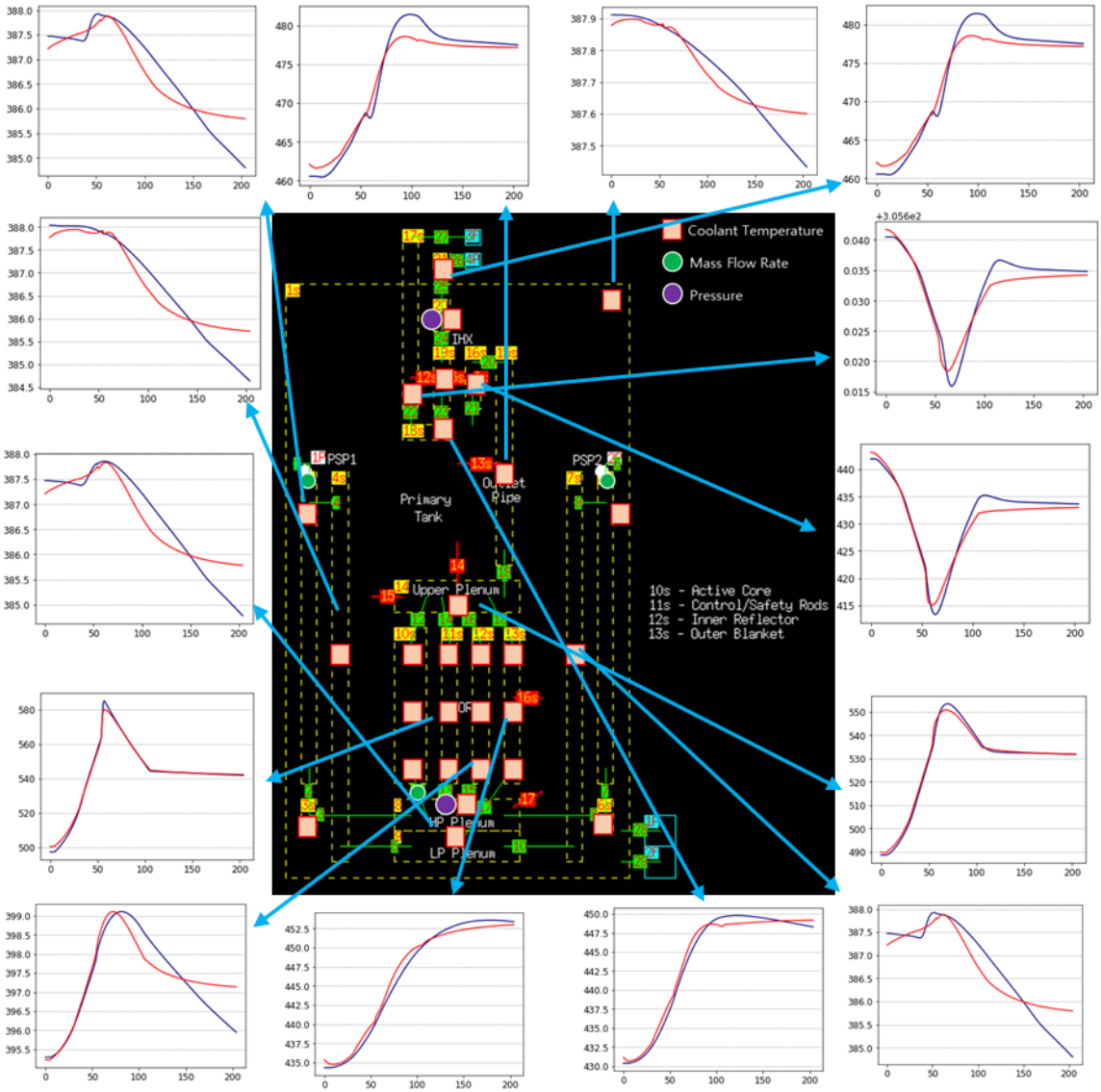


Figure 61. Temperature Distribution in the Reactor System

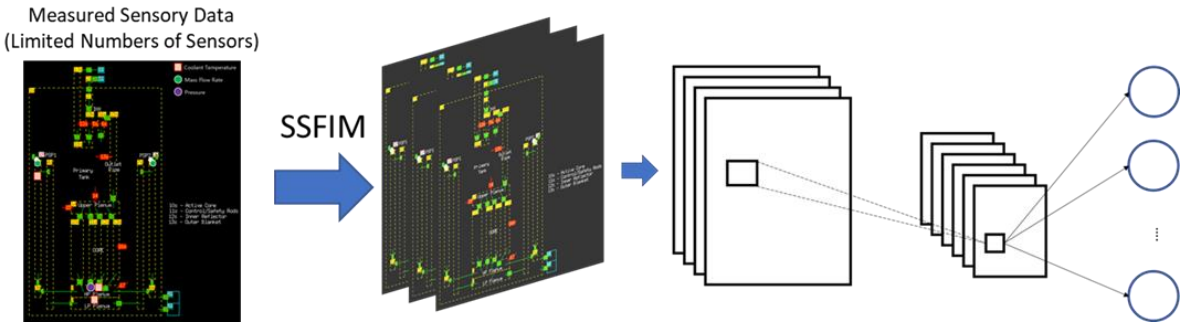


Figure 62. Suggestion for using the SSFIM to Provide Information to CNN

6.3.2. Potential Usage for Prognosis

The RNN is widely used to study the prognosis based on temporal patterns and information. One of the most popular fields to use the RNN is forecasting the stock price movement. By connecting two LSTM models, one for a neural language model and the other for stock prediction, an automated training system is proposed. The system analyzes the information from news articles about a company and predicts the probability of a direction of the stock price changes [55]. Mohan et al. use the RNN architecture to train the models on a dataset that includes closing stock prices and news articles of the Standard and Poor's (S&P) 500 companies. Based on the inputs of the model, which consist of prices, text polarity, and textual information, the next day's stock prices are predicted [57]. The case study 3-2 extends the diagnosis sphere by proving the short-term (one second later) forecasting ability of the ML-based SSFIM. Based on the approach, even the farther into the future could be predicted from the operation history and current measurement by using another type of the RNN architecture. Figure 63 illustrates the many-to-one architecture used as a basis of the ML-based SSFIM and many-to-many architecture, which has a potential to be used as a prognosis tool. In order to design the prognosis model based on the RNN, there are two major considerations:

- ✓ How the model reflects the operator's action that are planned in the input dataset
- ✓ How the model contains the future's physical state variables in the input dataset

Currently, in the NAMAC project, the prognosis model based on dynamic recurrent network with LSTM is being developed to predict the reactor states.

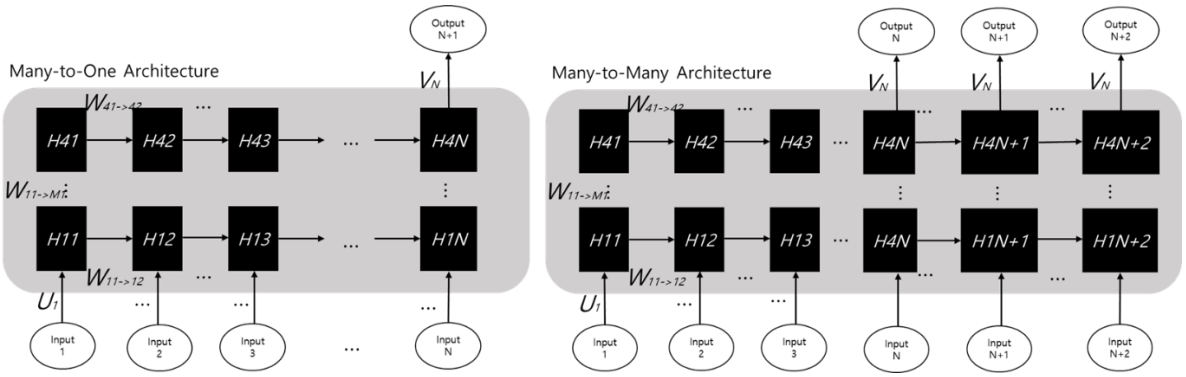


Figure 63. Many-to-One Architecture (Left) and Many-to-Many Architecture (Right)

6.4. Summary

The ML model is not a "Black Box," but it does not mean that the ML model infers the outputs by understanding the physical phenomena in the reactor based on thermal-hydraulics. In the simulation code, the SSF is calculated based on the conservation equation and the boundary conditions, and related information, whereas, the SSF is inferred by the ML model, which is trained by training data. The GOTHIC code supports the ML-based SSFIM to diagnose the plant states in real time by providing the physical variable data to train the ML algorithm. In conclusion, the ML-based SSFIM is the knowledge-based data driven model.

To develop the robust model, the normalized value of inputs is important in training the ML algorithm. In the case study, the mass flow rate at pump #1's flow path is the most powerful feature because the length of the change range during a transient is larger than any other variables in a normalized scale because of reverse flow. On the other hand, the temperature is sensitive to noise. Thus, to design the robust model in which the temperature is also included in the input data, categorizing according to the locations is required for determining the upper bound and lower bound in the normalization process.

The ML-based SSFIM has good inference ability unless the extrapolated range is far from the trained range. In transient B, within ± 20% of pump speed to the trained range is an acceptable field to be covered by the ML-based model to successfully infer the SSF. However, ± 40% of pump speed and totally different scenarios are unacceptable fields for the model. To prepare for severe cases, both the range and various types of scenarios are considerable factors in data generation process. Also, the head event identification before selecting the diagnosis module from the DMS is highly recommended.

By using the RNN's capability in information extension, other models in the AI-guided diagnosis system would be supported in various ways. The ML-based inference model would be expected to be utilized to provide a virtual environment to classify the abnormal status in the reactor. Moreover, the RNN would be used as the fundamental algorithm to predict the plant damage state in the future. Surely, there are many challenges in developing the models, but the experiences from the development of SSFIM would be helpful to resolve them.

CHAPTER 7. CONCLUSION

The Artificial Intelligence (AI) guided diagnosis, the primary stage of the Nearly Autonomous Management and Control (NAMAC) system, aims to deliver the plant damage states to the prognosis stage in real-time. The outcome from the diagnosis should be accompanied by accuracy and reliability because it provides the initial condition to both the prognosis and operator's action strategy. To identify the plant damage states, in the thesis, the Safety Significant Factor (SSF) is introduced to represent the physical plant states as the inferred target from the inputs of measurable physical variables in the Machine Learning (ML) based SSF Inference Model (SSFIM). The SSFIM is a ML-based, simulation-informed data-driven, and real-time diagnosis model to be used as a part of the AI-guided diagnosis model.

This chapter concludes the research work by summarizing the notable findings and discussions from the case studies, emphasizing the contributions, and suggesting future work related to this work.

7.1. Summary

The SSFIM based on the Recurrent Neural Network (RNN) with Gated Recurrent Units (GRU) successfully infer the Fuel Centerline (FCL) and Cladding (CL) temperatures from a few measurable physical variable data. The RNN is confirmed as an appropriate algorithm for dealing with the long-term time series data during transient, and the importance of the learning rate scheme to train the ML algorithm is enhanced. By using the algorithm and scheduler, the ML-based SSFIM shows the successful model performance within 1% which is accuracy as good as the thermocouple in the FCL even though the random noises are included in the dataset. The model performance depends on not the number of inputs but the type of variables. The most powerful variable is the mass flow rate at pump #1's flow path because its normalized value for the ML algorithm has the widest range length owing to the reverse flow.

The RNN proves its capability by allowing the ML-based inference model to extend the diagnosis range in the aspect of both coverage and time span. The adapted ML-based SSFIM infers temperature distribution in the reactor system without using temperature data. Also, the RNN's capability enables the model to be the faster-than-real-time diagnosis model by successfully

predicting the one second later SSF. Both case studies' results motivate the diagnosis field to be broadened.

Through the estimation of inference ability, the model's availability, limitations, and solutions to compensate for the shortcomings are evaluated. The ML-based model is able to infer the SSF by using the extrapolation in an acceptable field. But, in the case of an unacceptable field and the different field extrapolation, both the head event's range and types should be considered to cover the extreme scenarios in the training dataset from the deterministic thermal-hydraulic code.

The ML model does not calculate the SSF by using the thermal-hydraulics in the reactor system but infer the SSF by training the algorithm with the data from the best-estimate code. The simulation code has been used to analyze the safety and operating analysis, but data generation to support the ML-based model is added as one of the purposes. In conclusion, the ML-based SSFIM is a knowledge-based data-driven ML-based Inference model

7.2. Contribution

The major contributions of the thesis are as follows:

1) AI-guided diagnosis model is suggested as a primary stage of the NAMAC system. The NAMAC system is expected to transfer the human-centric plant control system to an AI-guided control system for reducing human error and recommending optimized action after safety challenges in the plant. Since the diagnosis is the first and fundamental stage in the system, informative data should be forwarded to following stages, such as prognosis and strategy planning, for applying the initial conditions. The AI-guided diagnosis transforms the raw incoming dataset from the actual reactor to physical plant damage state, dominant event, and Structures, Systems, and Components (SSC) availability. Digital twin is introduced to store information, data, and diagnosis modules to be used in the NAMAC system's operation. According to the transient conditions, a proper diagnosis module is selected by the operator's decision that is supported by the diagnosis engine arrangement workflow. The diagnosis engine calls the modules from the DT-Diagnosis (DT-D) class to carry out the objects. This high modular engine with operational workflow is proposed in the research work.

2) The robust ML-based SSFIM is developed to be a part of the AI-guided diagnosis model. This thesis focuses on developing the ML-based model to identify the physical plant damage state in the AI-guided diagnosis model. By using the only one physical variable, the SSF is reasonably inferred despite the 5% of random noise from the measurement. Moreover, the ML-based inference model proves its capability by mapping the temperature distribution in the reactor system to support another diagnosis module of which the objective is identifying the dominant event for the AI-guided diagnosis model.

3) The ML-based model is analyzed to provide the remarks in development. The RNN algorithm recognizes not the time-dependent physical variables but the trends of the input variables. In other words, data preprocessing is a critical step in the development of the ML model. There are several suggestions on how to improve the ML-based SSFIM's performance.
   - ✓ Reduce the sensor noise from the measurement devices.
   - ✓ Subdivide the physical variables in accordance with characteristics for the normalization process.
   - ✓ Generate a large amount of training data to cover the wide range and various types of scenarios.
   - ✓ Identify the head event before selecting the diagnosis modules of the SSFIM or develop the integrated model trained on all types of the scenarios.

4) The ML-based model is not limited to infer the plant damage states, but it offers diverse potentials to another ML field. As mentioned above, mapping the unobservable physical phenomena in the reactor system would provide the virtual environment which is used to classify the dominant event in transient. Also, the RNN would be useful ML algorithm for the prognosis.

7.3. Recommendations for Future Work

As discussed in previous chapters and assumptions, future tasks would be classified into two parts:

1) Improvement of the AI-guided Diagnosis Model

- In chapter 3, we have assumed that other technical functions of the AI-guided diagnosis model are fulfilled. Since one of the functions is accomplished, identification of the dominant event will be achieved. The ML-based inference model would support the testing of the AI-guided diagnosis model.
- Another assumption is that incoming datasets are intact and correct. However, the False Sensory Data Detection Model (FSDDM) is introduced in the initial configuration of the diagnosis engine. Development of the FSDDM would be necessary to improve the AI-guided diagnosis model.

2) Improvement of the ML-based SSFIM
   - Only the random error is assumed as the measurement error. To infer the SSF despite the false signals, a more robust model should be developed. It can start to answer the following questions:
     - ✓ How many correct physical variables are required when a false signal is included in the input dataset?
     - ✓ What type of bias is the most aggressive false signal in which the model can still infer the SSF?
   - As mentioned above, sensitivity study for normalization would improve the model performance in inferring the SSF.

With these recommendations, an advanced and improved AI-guided diagnosis model will be developed as a part of the NAMAC system.

REFERENCES

[1] R. S. Denning, Understanding the Nature of Nuclear Power Plant Risk, ICAPP'12, 2012.

[2] N. Dinh, Development of a Nearly Autonomous Management and Control System for Advanced Reactor, Technical Volume, 2018.

[3] T. Aven, On the use of conservatism in risk assessment, Reliability Engineering and System Safety, 2015.

[4] IAEA, Approaches and Tools for Severe Accident Analysis for Nuclear Power Plants, Safety Reports Series No. 56, 2008.

[5] IAEA, Considerations on the Application of the IAEA Safety Requirements for the Design of Nuclear Power Plants, IAEA TECDOC SERIES, IAEA-TECDOC-1791, 2016.

[6] IAEA, Deterministic Safety Analysis for Nuclear Power Plants, IAEA Safety Standards for protecting people and the environment, Specific Safety Guide No. SSG-2, 2009.

[7] Taleb, N., The Black Swan, 2007.

[8] Rumsfeld, D., Known Unknown: A Memoir, 2011.

[9] Martin, T., mTRACK - Monitoring Time-varying Relations in Approximately Categorized Knowledge, Proceedings of the 2009 IEEE Symposium on Computational Intelligence in Security and Defense Applications (CISDA 2009), 2009.

[10] G. Borghini, J. Imbert, and A. Golfetti, Skill, Rule and Knowledge-based Behaviors Detection during Realistic ATM Simulations by Means of ATCOs' Brain Activity, 2015.

[11] D. Embrey, Understanding Human Behavior and Error, Human Reliability Associates, 2005.

[12] M. Berry et al, Machine Learning and Understanding for Intelligent Extreme Scale Scientific Computing and Discovery, DOE Workshop Report, 2015.

[13] S. Haykin, Neural Networks and Learning Machines, 2009.

[14] Y. LeCun, Y. Bengio, and G. Hinton, Deep Learning, 2015.

[15] H. Lee et al, Convolutional Deep Belief Networks for Scalable Unsupervised Learning of Hierarchical Representation, 2009.

[16] W. Liu et al, IMAGENET Large Scale Visual Recognition Challenge (ILSVRC), 2015.

[17] G. Hinton et al, Deep Neural Networks for Acoustic Modeling in Speech Recognition, 2012.

[18] D. Silver et al, Mastering the Game of Go without Human Knowledge, DeepMind, 2017.

[19] W. Zaremba, Recurrent Neural Network Regularization, 2015.

[20] T. Mikolov et al, Extensions of Recurrent Neural Network Language Model, 2011.

[21] S. Liu et al, A Recursive Recurrent Neural Network for Statistical Machine Translation, 2014.

[22] N. Maknickiene, and A. Maknickas, Application of Neural Network for Forecasting of Exchange Rate and Forex Trading, 2012.

[23] Ulrich K. T. and Eppinger S. D., Product Design and Development, Chapter 9: Product Architecture, 1994.

[24] M. Khatib-Rahbar et al, Accident Diagnostic, Analysis and Management (ADAM) System Applications to Severe Accident Management, 2001.

[25] K. Ahn, and S. Park, Development of a Risk-Informed Accident Diagnosis and Prognosis System to Support Severe Accident Management, 2009.

[26] M. Saghafi, and M. Ghofrani, Accident Management Support Tools in Nuclear Power Plants: A Post-Fukushima Review, 2016.

[27] T. V. Santosh et al, Symptom Based Diagnostic System for Nuclear Power Plant Operation using Artificial Neural Network, 2003.

[28] T. V. Santosh et al, Application of Artificial Neural Network to Nuclear Power Plant Transient Diagnosis, 2007.

[29] T. V. Santosh et al, Diagnostic System for Identification of Accident Scenarios in Nuclear Power Plants using Artificial Neural Networks, 2009.

[30] T. V. Santosh et al, A Diagnostic System for Identifying Accident Conditions in a Nuclear Reactor, 2011.

[31] K. Mo et al, A Dynamic Neural Network Aggregation Model for Transient Diagnosis in Nuclear Power Plants, 2007.

[32] A. Ayodeji, Knowledge base operator support system for nuclear power plant fault diagnosis, 2018.

[33] S. Yoon et al, EBR-II Data Digitization, INL/EXT-14-32937, 2014.

[34] D. Roverso, Plant Diagnostics by Transient Classification: The ALADDIN Approach, 2002.

[35] H. A. Saeed et al, Novel fault diagnosis scheme utilizing deep learning networks, 2019.

[36] K. Yoo et al, Smart support system for diagnosing severe accidents in nuclear power plants, 2018.

[37] J. Lee et al, Use of Dynamic Event Trees and Deep Learning for Real-Time Emergency Planning in Power Plant Operation, 2019.

[38] N. Dinh, NE 591: Nuclear Safety Class Lecture Note: Nuclear Reactor Safety – Reactor Accident, 2015.

[39] D. J. Hill et al, Experimental Breeder Reactor II (EBR-II) Level 1 Probabilistic Risk Assessment, Nuclear Science and Engineering Division, ANL-NSE-2, 2018.

[40] J. W. Lane and T. L. George, Benchmark of GOTHIC to EBR-II SHRT-17 and SHRT-45R Tests, 2019.

[41] J. M. Link et al, EBR-II Simplified GOTHIC Model, Zachry Group, 2019.

[42] P. Balestra and M. Avramova, EBR II - Experimental Breeder Reactor design overview, 2018.

[43] R. Kohavi, A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection, International Joint Conference on Artificial Intelligence (IKCAI), 1995.

[44] S. Arlot, and A. Celisse, A Survey of Cross-validation Procedures for Model Selection, 2009.

[45] S. Raschka, Model Evaluation, Model Selection, and Algorithm Selection in Machine Learning, 2018.

[46] United States Environmental Protection Agency et al, Multi-Agency Radiological Laboratory Analytical Protocols Manual (MARLAP), Chapter 19 Measurement Uncertainty, NUREG-1576, 2004.

[47] I. Loshchilov and F. Hutter, SGDR: Stochastic Gradient Descent with Warm Restarts, ICLR, 2017.

[48] L. N. Smith, Cyclical Learning Rates for Training Neural Networks, 2017.

[49] R. Veerasamy et al, Validation of QSAR Models – Strategies and Importance, International Journal of Drug and Discovery, 2011.

[50] D.H. Wolpert and W. G. Macready, No Free Lunch Theorems for Optimization, 1997.

[51] Voelsing K. et al, GOTHIC Thermal Hydraulic Analysis Package User Manual, Electric Power Research Institute, 2016.

[52] Voelsing K. et al, GOTHIC Thermal Hydraulic Analysis Package Technical Manual, Electric Power Research Institute, 2016.

[53] Evans J.D. et al, Straightforward Statistics for the Behavioral Sciences, 1996.

[54] Kim Y. et al, Operator Support System with Diagnosis of Abnormal Status by Convolution Neural Networks, 2019.

[55] Pinheiro L. and Dras M., Stock Market Prediction with Deep Learning: A Character-based Neural Language, 2017.

[56] Mohan S. et al, Stock Price Prediction Using News Sentiment Analysis, 2019 IEEE Fifth International Conference on Big Data Computing Service and Applications, 2019.

[57] K. Patel, Introduction to Neural Network: Deep Learning for NLP, ICON 2017, 2017.

[58] N. Srivastava et al, Dropout: A Simple Way to Prevent Neural Networks from Overfitting, 2014.

[59] K. Cho et al, Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation, 2014.

[60] J. Chung et al, Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling, 2014.

[61] S. Hochreiter, and J. Schmidhuber, Long Short-term Memory, Neural Computation 9, 2017.

[62] R. Rana et al, Gated Recurrent Unit (GRU) for Emotion Classification from Noisy Speech, 2016.

[63] International Atomic Energy Agency, IAEA Safety Glossary, Terminology Used in Nuclear Safety and Radiation Protection 2018 Edition, 2018

[64] Convention on Early Notification of a Nuclear Accident, INFCIRC/335, IAEA, Vienna (1986).

[65] International Atomic Energy Agency, The International Nuclear and Radiological Event Scale, User's Manual, 2008

[66] P. Slovic, B. Fischhoff, and S. Lichtenstein, "Rating the Risks," Environment 21,1979.

[67] C. Perrow, Normal Accident - Living with High Risk Technologies, 1994

[68] J. Downer, The unknowable ceilings of safety: Three ways that nuclear accidents escape the calculus of risk assessments, The ethics of Nuclear Energy, 2015

[69] J. Downer, Anatomy of a Disaster: Why Some Accidents Are Unavoidable, centre for analysis of risk and regulation, 2010

[70] J. A. Rijpma, Complexity, Tight-Coupling and Reliability: Connecting Normal Accidents Theory and High Reliability Theory, 1997

[71] G. Vayssier, Basic Principles of Accident Management, IAEA Training Workshop on Severe Accident Management Guideline Development using the IAEA SAMG-D Toolkit, 2017

[72] R. Harter, Technical Support Guidelines – Tools to Support Implementation of Accident Management Guidance at US BWRs, Workshop on Understanding the Role of Severe Accident Management Guidelines, 2016

[73] R. Harter, Lecture 11 Organizational Aspects of SAMG Implementation, IAEA Workshop on Development of SAMGs using the IAEA's SAMG Development Toolkit, 2017

[74] R. Gronberg, "AI could one day control nuclear reactors; NC State researchers could make it happen": The News & Observer, 2018

[75] L. Fridman, MIT 6.S904: Deep Learning for Self-Driving Cars, Lecture 3: Deep Reinforcement Learning , 2018

[76] L. J. Savage, The Theory of Statistical Decision, Journal of the American Statistical Association, Vol 46, No.253, 1951

[77] C. Miller et al, Recommendations for Enhancing Reactor Safety in the 21st Century, The Near-Term-Task-Force Review of Insights from the Fukushima Dai-ichi Accident, 2011

[78] V. M. McCree, Proposed Plans for Resolving Open Fukushima Tier 2 and 3 Recommendations, SECY-15-0137, 2015

[79] V. M. McCree, Enhancements to Severe Accident Training for NRC Staff and Severe Accident Management Guidelines for Resident Inspectors, Proposed Resolution Plan for Tier 3 Recommendation 12.2, 2015

[80] N. Hiranuma, Overview of Guidelines on Technical Safety Review Services for Accident Management, IAEA Workshop on the Development of SAMGs using the IAEA's SAMG Development Toolkit, 2017

[81] L. Lin et al, NAMAC Q2 Quarterly Technical Report, 2019

[82] L. Lin et al, Development and Assessment of Advanced Algorithms, NAMAC Q5 Review Meeting, 2020

[83] U. Krogmann, From Automation to Autonomy – Trends Towards Autonomous Combat System, 1999

[84] Y. Kim et al, Challenges for Autonomous Water-cooled Small Modular Reactor ATOM Development, 8th International Conference on Simulation Methods in Nuclear Science & Engineering, 2018

[85] A. Abdelhameed et al, Feasibility of passive autonomous frequency control operation in a Soluble-Boron-Free small PWR, 2018

[86] D. Lee et al, Autonomous operation algorithm for safety systems of nuclear power plants by using long-short term memory and function-based hierarchical framework, 2018

[87] S. M. Cetiner et al, Supervisory Control System for Multi-Modular Advanced Reactors,

ORNL/TM-2016/693, Oak Ridge National Laboratory, 2016

[88] B. R. Upadhyaya et al, Autonomous control of space reactor systems, DE-FG07-04ID14589/UTNE-06, 2007

[89] S. Park et al, SAMEX: A severe accident management support expert, 2010

[90] A. Alfonsi et al, Dynamic Event Tree Analysis Through RAVEN, 2013

[91] T. Sumner et al, Benchmark Specifications and Data Requirements for EBR-II Shutdown Heat Removal Tests SHRT-17 and SHRT-45R, ANL-ARC-226 Rev.1, 2012.

[92] T. Sumner et al, Simulations of the EBR-II Tests SHRT-17 and SHRT-45R, NURETH-16, Argonne National Laboratory, 2015

[93] N. Yue et al, Thermal-hydraulic analysis of EBR-II Shutdown Heat Removal Tests SHRT-17 and SHRT-45R, Progress in Nuclear Energy 85 (2015) 682-693, 2015

[94] D. Kelly and C. Smith, Bayesian Inference for Probabilistic Risk Assessment, 2011.

[95] M. Knochenhauer et al, Using Bayesian Belief Network (BBN) Modeling for Rapid Source Term Prediction – RASTEP Phase 1, 2012.

[96] K. Murphy, Dynamic Bayesian Networks: Representation, Inference and Learning, 2002

[97] K. Groth et al, Building and using Dynamic Risk-informed Diagnosis Procedures for Severe Accidents, 2017

[98] M. Darling et al, Intelligent Modeling for Nuclear Power Plant Accident Management, 2018

[99] M. R. Denman et al, Advance Liquid Metal Reactor Discrete Dynamic Event Tree/Bayesian Network Analysis and Incident Management Guidelines (Risk Management for Sodium Fast Reactors), Sandia National Laboratory, 2015

[100] D. Silver, Advanced Topics Reinforcement Learning Lecture, Lecture 1. Introduction to Reinforcement Learning, 2015

[101] D. Silver, Advanced Topics Reinforcement Learning Lecture, Lecture 2. Markov Decision Processes, 2015

APPENDICES

APPENDIX A. RECURRENT NEURAL NETWORK MODULES

In this appendix, the details of deep neural network and advanced modules of the RNN are described.

A.1. Deep Neural Network

Since the DNN has multiple hidden layers, there are more numbers of model parameters, which would be updated by backpropagation, than single hidden layer neural network (see figure 64). Deep learning has evolved by overcoming its weaknesses through cutting-edge techniques. Gradient descent is not available in the DNN, because differentiation is not possible owing to multi-hidden layers. Therefore, backpropagation is used to update the weights in the network. However, as the number of layers increase, vanishing gradient, in which calculated error dissipates during backpropagation, happens. Vanishing gradient is caused by the activation function, for which differential value is small (maximum value=0.25), such as the tanh or sigmoid function [57]. The Rectified Linear Unit (ReLU) function is introduced to assure weight update even in the deep network. Figure 65 shows the ReLU function, that is described by following equation.

$$R(x) = \begin{cases} 0 & for\ x < 0 \\ x & for\ x \geq 0 \end{cases} \tag{30}$$

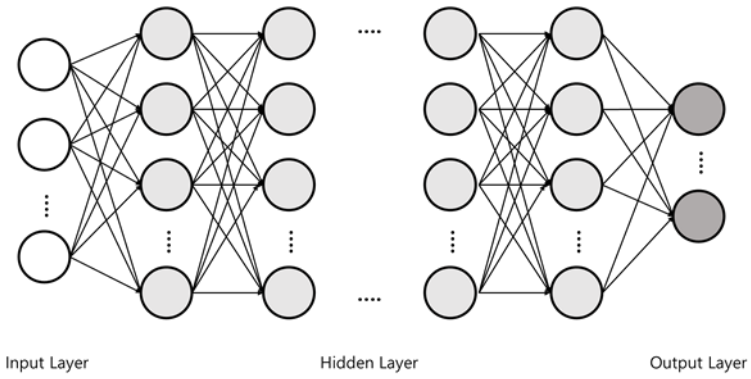


Figure 64. Deep Neural Network

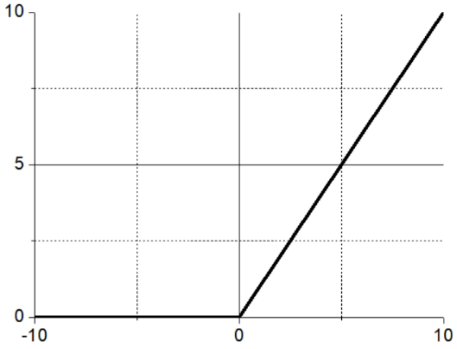


Figure 65. ReLU Function

Another problem of the DNN is over-fitting, which causes the decline of inference capability. The dropout algorithm is one of the regularizations used to reduce over-fitting by omitting several neurons in the network [58]. By deleting connections in accordance with the dropout probability, the DNN is enforced to not memorize specifics in the datasets. As illustrated in the figure 66, the activations of the dropped neuron is zero. The DNN is growing rapidly by virtue of computing power based on the Graphics Processing Unit (GPU), advanced tools, and development environment.

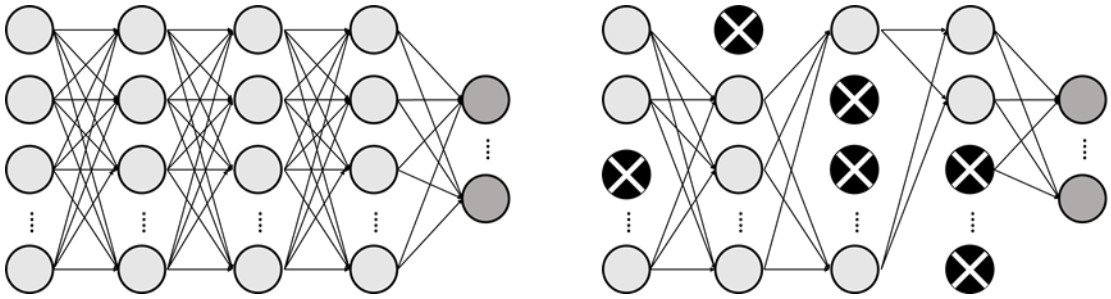


Figure 66. Dropout Model: A Standard Neural Net (Left) and An Example of Dropout Method in the Network (Right)

A.2. LSTM and GRU Modules

Since the RNN can be counted as DNN by converting into the unfolded time step, updating the parameters (U, V, and W), which are shared in the RNN, may result in the vanishing gradient problem during backpropagation. Hence, besides using the ReLU function, an especial unit such

as the LSTM unit module or GRU module is installed within a hidden layer to fundamentally resolve the problem [59]. The units are activated to enhance the capability of RNN by remembering the distant past. Figure 67 shows the formation of the cell units where various built-in gates function to update the parameters with hidden states in the network. Also, figure 68 illustrates the LSTM module and GRU module which will be introduced next [60].

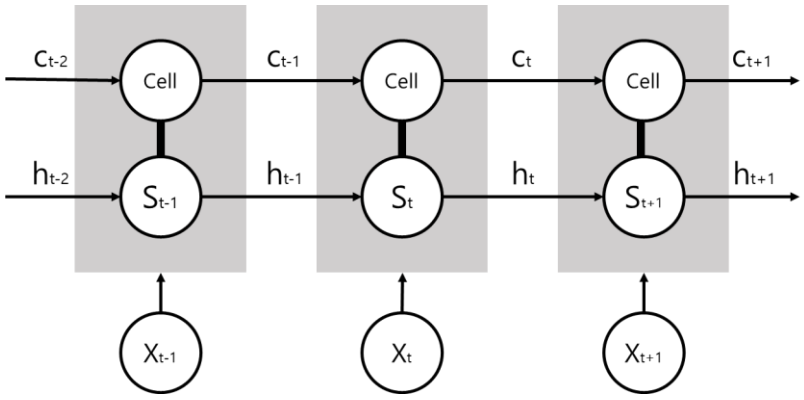


Figure 67. Gated Unit Installation in Recurrent Neural Network

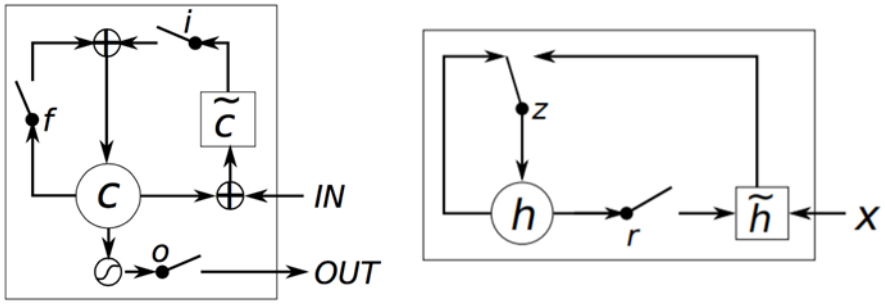


Figure 68. LSTM Module (Left) and GRU Module (Right)

The constitution of the LSTM is designed to decide how much the RNN should accept from a new input and how much the network should remember from the previous hidden state [61]. The LSTM, which changes the RNN inner nodes into a memory cell, comprises several gates and cells (Chung et al, 2014): an input gate (i), an output gate (o), a forget gate (f), a new memory cell ($\tilde{c}$), a final memory cell (c), and a final hidden state (h). The sigmoid function, of which the range is from 0 to 1, is used in the equations of gates (σ) for activation states. The final memory

cell (*c*) updates the weights according to the importance of the information, which heads toward the new memory cell, by obstructing the previous memory to 'forget' in the forget gate (*f*) or taking new 'input' in the input gate (*i*). The hidden state (*h*) is updated by the final memory cell (*c*) with the output gate (*o)*. Through the processes, which are repeated to adjust the parameters by reducing the backpropagation error, the network learns to predict the output from the input datasets. The equations for calculation in each gate and cell are as follows:

$$i_t = \sigma\left(W^{(i)}x_t + U^{(i)}h_{t-1}\right) \quad (31)$$

$$f_t = \sigma\left(W^{(i)}x_t + U^{(i)}h_{t-1}\right) \quad (32)$$

$$o_t = \sigma\left(W^{(o)}x_t + U^{(o)}h_{t-1}\right) \quad (33)$$

$$\widetilde{c_t} = tanh\left(W^{(c)}x_t + U^{(c)}h_{t-1}\right) \quad (34)$$

$$c_t = f \circ c_{t-1} + i_t \circ \widetilde{c_t} \quad (35)$$

$$h_t = o_t \circ tanh(c_t) \quad (36)$$

The GRU consists of two gates [59]: a reset gate (*r*) and an update gate (*z*) and two hidden layers: a candidate activation ($\tilde{h}$) and the activation (*h*). As the separated cell does not exist in the GRU unit, the function of the gate is extended to cover the role of the memory unit. In the module, the reset gate is able to block the previous hidden state and 'reset' with new input data, then updating the hidden state (*h*), in which the previous state and new input data are considered simultaneously, is controlled through the result of the update gate. The following are the equations for calculation in GRU module:

$$z_t = \sigma\left(W^{(z)}x_t + U^{(z)}h_{t-1}\right) \quad (37)$$

$$\widetilde{h_t} = tanh(Wx_t + r_t \circ Uh_{t-1}) \quad (38)$$

$$h_t = z_t \circ h_{t-1} + (1 - z_t) \circ \widetilde{h_t} \quad (39)$$

The LSTM unit module and the GRU module are competitively utilized in the RNN. Both pass the revised information to the next state to remember the important data for preventing the vanishing gradient. Theoretically, the LSTM features better in memorizing than the GRU on the long sequence data analysis [62]. However, as we can see in figure 68 above, the GRU is a relatively recently developed module that is a simplified version of the LSTM. Hence, the GRU is

able to train more efficiently or faster than the LSTM thanks to fewer parameters that need to be updated in the less complex structure.

APPENDIX B. NAMAC SYSTEM

This appendix describes the backgrounds of the NAMAC system, such as features of a nuclear accident, the instrumentation and control (I&C) system, recent research trends for developing autonomous control systems, and other technical components for the NAMAC system except AI-guided diagnosis.

B.1. Background

A nuclear accident is defined as any unintended event involving facilities from which a radiation release occurs or is likely to occur and leads to significant consequences for people and the environment [63 - 65]. According to the IAEA definition, the nuclear accident indicates a severe accident, however, the NAMAC system aims to operate to cover the full range of plant states, from normal operation to severe accident management. Although the thesis targets a transient as stated in the research scope, in this section, the backgrounds for the NAMAC system are introduced based on the severe accident management since the severe accident is the most serious case.

B.1.1. Nuclear Accident

A nuclear accident, which is featured as a rare but extreme event, is difficult to explain when asked how such immense consequences could have arisen owing to its large uncertainty and involuntary nature [66]. Moreover, since an accident in the reactor system happens with non-linearity and carries a threshold of sudden transition, the extent of the event can be widened by small but critical changes. For instance, in the case of a severe accident, core collapse that leads to core damage is a threshold effect in which there is a small increase of core temperature around the melting point. Hence, identification of true potentials of an accident is a substantial issue in managing the risk.

Through Normal Accident theory and Epistemic Accident theory, characteristics of a nuclear accident can be described. Normal Accident theory is defined as a non-critical failure, such as when a trivial initial event causes a catastrophic accident through some unexpected paths within the system [67]. A normal accident occurs by combinations of ordinary problems. But a normal

accident is unpredictable and unavoidable because of the complexity in the progression, therefore, it is not heuristic [68].

An epistemic accident arises from errors in establishing knowledge of the real world [69]. An epistemic accident is also unpredictable and unavoidable due to imperfect knowledge, theories, and assumptions. Typically, the accident type occurs in a highly innovative system that is upgraded by an engineers' techniques to overcome the limits of existing knowledge. An epistemic accident is heuristic in the system.

Nuclear accidents contain features of both a normal accident and an epistemic accident simultaneously. It is rarely predictable owing to both normal accident reasons and epistemic accident reason. A nuclear power plant is composed of multiple complex and tightly-coupled systems, which includes a reactor system, a coolant system, and a safety system. The engineering safety system, designed complicatedly, connected tightly, and in which the redundant components are installed, improves the reliability of the system operation. Hence, initial failures, which interact rapidly with other components in the system, are likely to increase the consequences. However, an established knowledge or design can be challenged from external hazards that can lead to a severe accident. In other words, a nuclear accident can initialize not only from transient or DBA, but also from an external event, such as floods, tsunamis, and seismic events. To manage the severe accident risk, the operator should recognize a current situation immediately for expeditious decision making.

One of the major problems of a nuclear accident is that the accident progression consists of irreparable conditions. For the response in the AOO or BDBA regime, the operator would take action against accident progression in anticipation of the function of safety barriers, such as fuel pellets, fuel cladding, reactor pressure vessel (RPV), and the containment building. However, in severe accident management, because the core and core structures are already damaged, the operator is pressed for time. The operator must also consider other unexpected events that may decisively threaten the safety barriers. For instance, a hydrogen explosion causes containment failure even though the conditions for its occurrence have large uncertainty. Therefore, the consequences would immensely increase over time during accident progression. In conclusion, the speed of transition in a nuclear accident's progression is debatable, but a severe accident entails rapidly changing situations that lead to the worst outcome. It puts a lot of pressure on operators to take action, so they have to make a decision as fast as possible.

In terms of mitigative action for accident management, a nuclear accident can challenge the decision-making paradigm. As mentioned above, a nuclear power plant is made up of complex tightly-coupled systems with high-technology designs. These sophisticated designs are the fruit of innovation in long term projects and have been developed steadily for the betterment of efficiency and safety [70]. Even though there is less likelihood of recurrence of a 'specific' nuclear accident in a high-technology system aspect, owing to human reliability aspect, an accident could recur. An operator support/training system should be developed to lessen human error. After all, a nuclear accident is not heuristic.

Table 38. Comparison among Normal Accidents, Epistemic Accidents, and Nuclear Accidents

Normal Accidents	Epistemic Accidents	Nuclear Accidents
Unforeseeable	Unforeseeable	Unforeseeable
More likely in tightly-coupled, complex systems	More likely in highly innovative systems	More likely in tightly-coupled, complex systems
Unlikely to reoccur	Likely to reoccur	Unlikely to reoccur in the system but likely to reoccur on human error
Do not challenge design paradigms	Challenge design paradigms	Challenge decision making paradigms
Not heuristic	Heuristic	Not heuristic

The ultimate goal of severe accident management is minimizing the consequence of radioactive release from the plant to the environment. Due to complex processes and large uncertainty, severe accident risk management is accomplished by optimal utilization of resources. In this respect, enhancing the support/training system is emphasized to mitigate the consequences while the accident is evolving.

B.1.2. Emergency Management

Both Emergency Operation Procedures (EOP) and Severe Accident Management Guidelines (SAMGs), of which implementation is supported by Technical Support Center (TSC) staff suggest a suitable response action depending on emergency conditions in the nuclear power plant system. Typically, the priority of EOP is the prevention of core damage with a focus on

reactivity control and core cooling by using engineered safety systems. Also, the EOP consists of response actions in precise and strict procedures, the results of which are obvious for risk analysis. Meanwhile, SAMGs are performed to maintain the containment integrity for mitigating any possible release of radioactive material during the BDBA regime after core damage. On account of a large uncertainty of the consequence in severe accident progression, SAMGs propose flexible response actions as guidelines. Decision making authority is transferred from the main control room (MCR) to TSC according to the transition from the EOP domain to the SAMGs domain.

SAMGs are established by considering the vulnerability of the safety barriers, the capability of the safety system components and potential drawbacks of both the system and operator ability. In SAMGs, to execute the strategy to mitigate the accident consequence, operators consider several subjects, such as parameters in the plant, available resources, appropriate actions, and potential negative consequences [71]. Parameters indicate a physical variable in the plant, for instance, the temperature in the steam line, pressure in the reactor vessel, and flow rate of inlet/outlet loop. Appropriate actions are based on accessible resources, which include valve, pump, and water, with logical rules to avoid an improper result. Examining potential negative consequences, which are involved in the prediction realm, is imperative to determine the response action.

TSGs are developed to assist EOP and Severe Accident Guidelines (SAGs) by engineering and technical support activities [72]. It identifies both limitations of the system functional capability and utilization of the available safety system for risk management which signifies the plant damage assessment, prognosis and instrument uncertainties. The structure of TSGs consists of Control Parameter Assessment Guidelines (CPAG), Plant Status Assessment Guidelines (PSAG), Function Status Assessment Guidelines (FSAG), and EOP/SAG Action Assessment Guidelines (EAAG). Through CPAG, PSAG and FSAG, TSC workers grasp the plant damage states and available safety system conditions. Then, EAAG suggests an adequate system operation with reasonable timing for the decision maker of Emergency Director. Figure 69 and figure 70 illustrate the responsibilities for accident management after Emergency Response Organization (ERO) activation with TSC organization and the role of EOP, SAMGs and TSGs on the accident progression, respectively [73].

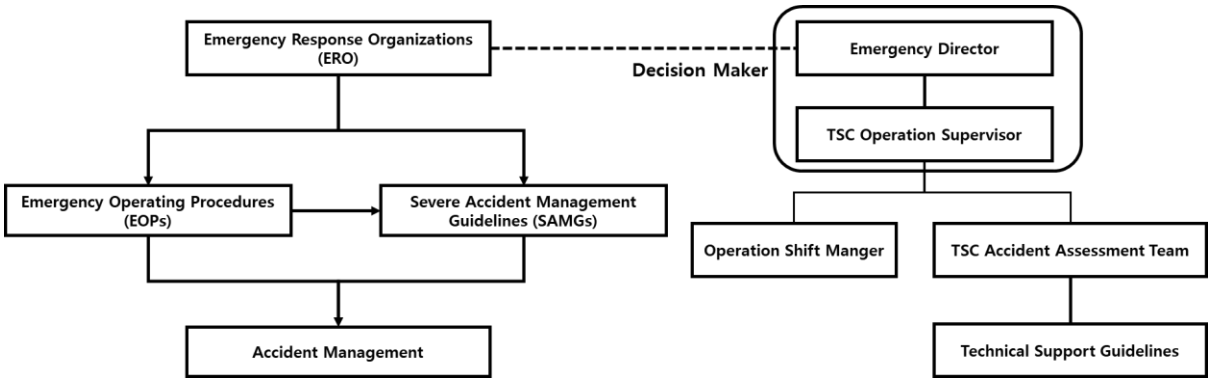


Figure 69. Accident Management with TSC organization

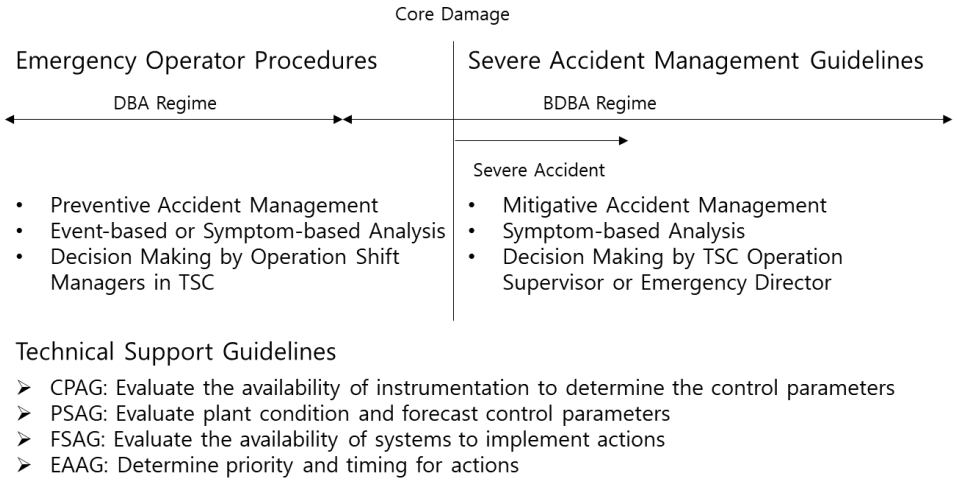


Figure 70. EOP, SAMGs, and TSGs

From the engineering perspective, TSGs should encompass not only epistemic understanding but also computational aids. To evaluate the control parameter, plant status, and function status, both a knowledge-based approach and a technique-based approach are ascertained in TSGs. However, in the case of EAAG, TSGs show only established knowledge such as logical flow charts to implement the actions with system criteria in safety barrier control strategies. To support EOP or SAMGs, which are symptom-based analyses used to prevent or mitigate the consequences, computational aid is inevitable.

B.1.3. Training on Nuclear Accident

After the Fukushima accident, hands-on training on a nuclear accident was emphasized for the effectiveness of emergency action programs through the Near-Term Task Force (NTTF)

evaluation. The Task Force recommends strengthening onsite emergency response strategy capabilities by training and doing exercises on SAMGs (Near-Term Task Force Recommendation 8.4) [77]. Even though existing strategies contain the training program to stress the importance of response to an emergency, a progressive training system is a remarkable feature for the risk management and control system.

Regular training exercises should be implemented to raise the level of emergency response capability quality pursuant to the rules about the SAMG training program. The regulation in 10 CFR 50.47 (b) (14) states that periodic exercises are required to evaluate major portions of emergency response capability, and periodic drills are conducted to develop key skills. Also, through 10 CFR 50.47 (b) (15), radiological emergency response training is stipulated to assist in an emergency [77]. NUREG-0654 delineate detailed matters for the enforcement of these regulation to correct deficiencies which are attested by a result of periodic training exercises.

The Nuclear Regulatory Commission's (NRC's) NTTF proposes recommendations, which are categorized into three tiers according to length of response terms, based on lessons learned from the Fukushima Daiichi accident. Studies and evaluations about the longer-term action are mainly subsumed under Tier 3 recommendations [78]. NTTF Recommendation 12.2 specifies that US NRC furthers the training system on severe accident managements. This recommendation consists of current progress and planned activities associated with both short-term and long-term enhancements. In particular, enhancement of training based on the State-of-Art Reactor Consequence Analysis (SOARCA) study and follow-up research is recommended to develop the upgraded training program on severe accident management [79].

SAMGs' training program carries the rules and skills which are necessary elements to execute the mitigative strategies. Accident management training may be conducted through several methods: computer-based training, classroom training, table-top drills/simulator drills, and ERO drills or exercises [73]. The program requires training on TSC's activities to support SAMGs via upgraded TSG. However, the suggested methods have implicit drawbacks to accomplish the program. For example, ERO drills or exercises are known as the most effective and practical method, but it is costly to prepare and drill.

SAMG's training system is limited by severe accident conditions and decision making for operators [73,80]. SAMG's training approach is based on the existing knowledge or procedures, and the program can be limited for severe accident scenarios. Moreover, since a severe accident is

not heuristic, the information related to severe accident phenomena may be rarely updated in the training program. Thus, skill-based training, which includes emergency response to abnormal and complicated symptom-based analyses, is hardly performed in an SAMG's training program.

The NAMAC system prefers a simulation-based approach to appraise the EOP or SAMGs owing to practical constraint of the accident case realization. A next-generation simulator which reflects NAMAC philosophy should be based on both established knowledge and updated skills from experience or studies. Existing deterministic simulation code such as GOTHIC can calculate the physical variables in the plant from formulas by well-founded knowledge under given assumptions. However, a next-generation simulator demands competent diagnosis, appropriate prediction, and adequate response by skill-based training. Alternative engineering techniques are necessary to satisfy both approaches to develop a state-of-the-art simulator. For the NAMAC system, the reinforcement learning (RL) is suggested to provide the most adequate response, which has the maximized reward or minimized penalty function among given knowledge-based strategies by interaction between the operator and digital environment. Therefore, the skill-based training for action strategy is enacted by using the AI techniques.

The research suggests the ML algorithm to diagnose the plant damage states in transient. Failures of some components in tightly coupled systems of the nuclear power plant lead to abnormal operational states or accident conditions. Even though multifarious programs and activities are provided to educate and train the licensed operator through SAMG's training, human diagnosis has its own inherent uncertainty. The AI-guided diagnosis model will be able to replace human activities as an alternative method. The ML model, which can be deemed simulation-based, trains with physical variable data because the data are generated by best-estimate deterministic simulation. Therefore, the simulation data-driven approach is utilized to satisfy not only the ML training but also knowledge-based approach.

B.1.4. I&C system

A nuclear power plant, which comprises of a reactor system tightly coupled with cooling and safety systems, is safely operated by a well-coordinated I&C system. The operator uses the I&C system for 1) monitoring the physical processes within each system, 2) understanding the plant operation status, 3) preventing potential problems that might cause the plant damage, and 4) mitigating the consequences of the accident by using a safety system in an emergency. Recently,

the I&C system is mostly digitalized and includes a protection automation function that is activated under operator action to improve plant reliability and safety. Above and beyond that, an advanced research project developing an autonomous control system, which entails both plant operational state identification and action strategy assessment by using AI technology, has come under the spotlight [74].

AI application has enabled the formation of intelligent control systems that are capable of making rational and efficient decisions using data analysis and assimilation of general information. For instance, a self-driving car controls the wheel and pedals by itself to run safely by considering surroundings such as roads, stationary objects, and moving vehicles or pedestrians. In order to execute an optimized effector control strategy, there are several steps: 1) the surrounding features are extracted from data from the sensors, 2) knowledge is obtained through the machine learning techniques from information, and 3) reasoning and planning are performed to achieve a goal by using deep learning with a reward function [75]. Similarly, in nuclear power plant operation, AI techniques can be applied to the I&C system for risk management with the following steps: 1) incoming physical features are used from the installed signal instrumentation in the plant, 2) the ML model converts the physical feature data into information about plant damage states through the AI-guided diagnosis, 3) the simulation-guided prognosis with initial conditions from the diagnosis predicts the consequences of operational states or accident conditions, 4) an action strategy is evaluated to recommend the optimized one to the operator by using the RL, and 5) the operator determines the action strategy for the safety system operation to prevent or mitigate the consequences (see figure 71). Since the AI application to control the system reduces human error or misjudgment that can be induced by extreme pressure in an emergency, an advanced management and control system is necessary for comprehensive and cautious operation.

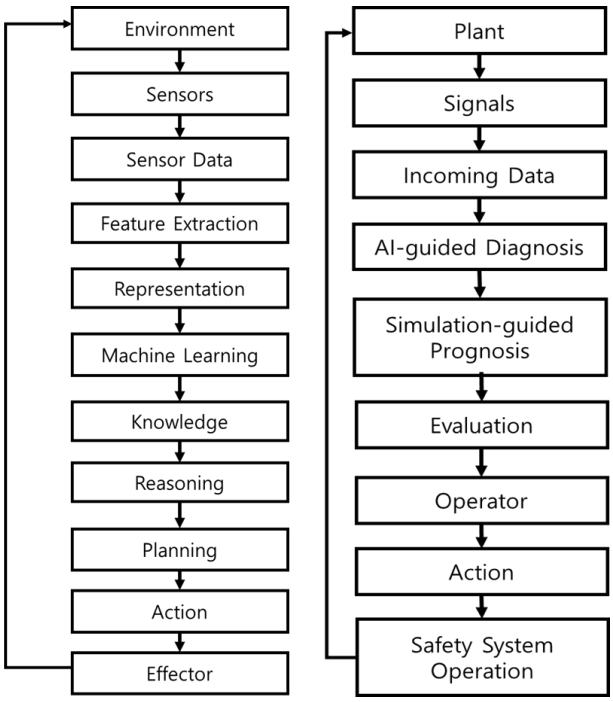


Figure 71. Schematic Flows of Self-Driving Cars and AI Application to I&C system

B.1.5. NAMAC System in Emergency Management

One of the main elements in the NAMAC system is knowledge-based strategies with considering features of a specific plant to achieve a success path which induces “no core damage” or “no containment failure.” In transient or DBA sphere, preventing core damage would be the first priority for the safety system operation, whereas, during a nuclear accident, the operator should make a least regret decision. A least regret decision is the optimized action that aims to minimize the worst regret of a decision maker [76]. The approach is the same as the severe accident management, where the ultimate goal is mitigation of the consequences by TSC’s efforts. Figure 72 shows the NAMAC’s position for emergency management to support TSC staff. From this perspective, the NAMAC system provides optimized strategy, such as sequence or extent of actions, in corresponding with an emergency based on not only knowledge (EOP/SAMGs) but also skill-based training by using the RL. Also, inversely, the system operation would be helpful to develop and validate the EOP/SAMGs for advanced reactors [2].

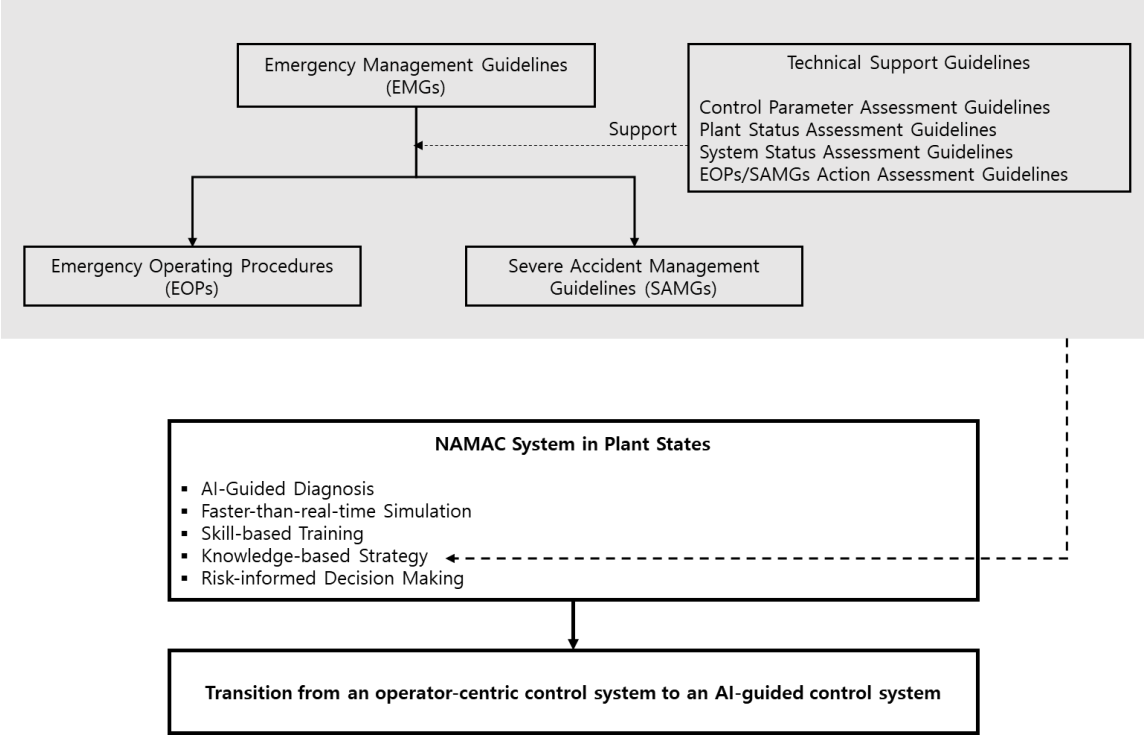


Figure 72. NAMAC System in Emergency Management

B.1.6. NAMAC System Workflow

NAMAC system architecture consists of three layers: the knowledge base layer, the developmental layer, and the operational layer. Figure 73 illustrates the NAMAC system architecture with its components [81]. Each layer has its objective in which 1) data to train the digital twins are secured in the knowledge base layer, 2) digital twins to fill out the digital twin hub are trained and tested in the developmental layer, and 3) the NAMAC system is tested to confirm the operational ability in the NAMAC operational layer. Finally, the NAMAC system is evaluated by monitoring the discrepancy between the predicted consequence and actual application consequence by the operator's action with preference structure. The discrepancy checker provides evidence to improve the digital twins' model performance. By applying for the architecture in developing AI-guided diagnosis system, the research focuses on training and testing digital twins in the developmental layer with knowledge base description, including: 1) issue space is demonstrated by transient scenario assumptions, and 2) RAVEN-GOTHIC is used as a simulation tool to store the data for digital twin training and testing. Discrepancies between actual

consequence and predicted consequence should be monitored in the operational layer to improve the NAMAC system's performance based on the model performance of the digital twins.

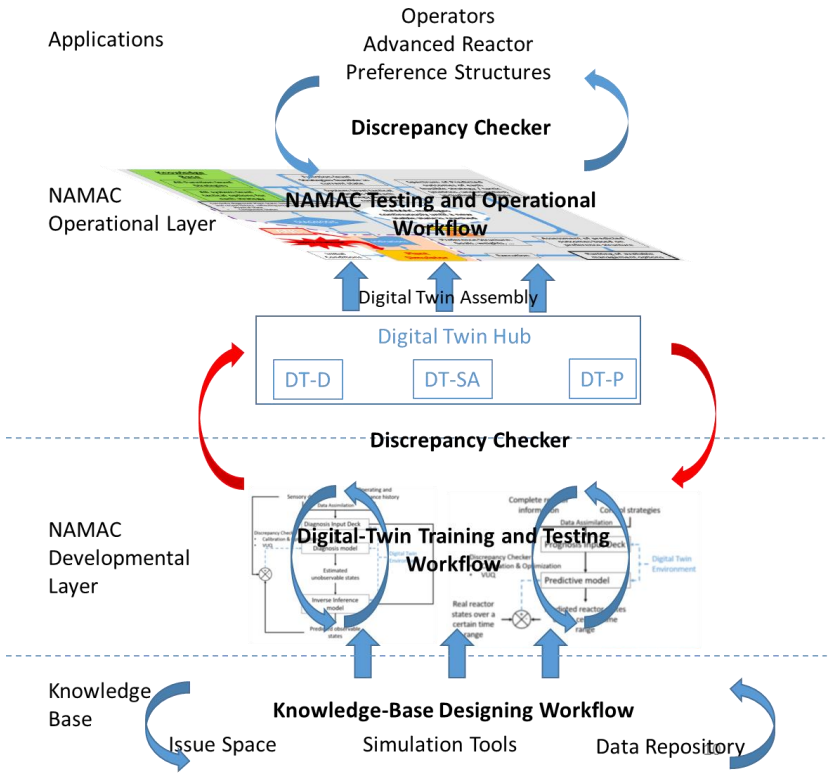


Figure 73. NAMAC System Architecture, Reprinted from 'NAMAC Q2 Quarterly Technical Report' by L. Lin et al, 2019

In the top layer of the system architecture, the NAMAC system consists of various types of digital twins with specific functions to achieve each objective. Figure 74 illustrates the system workflow which involves digital twins for diagnosis, prognosis, strategy inventory, and strategy assessment [82]. The adequate digital twins are provided from the digital twin hub, in which trained and tested digital twins are stored, by considering the constraints on corresponding transient or accident conditions. The constraints are determined from the application status including the operator's decision from preference structures. There are two discrepancy checkers: one for supporting the digital twin for strategy inventory and the other for determining the SCRAM commands. Finally, the recommendations from the digital twin of strategy assessment are delivered to the operators.

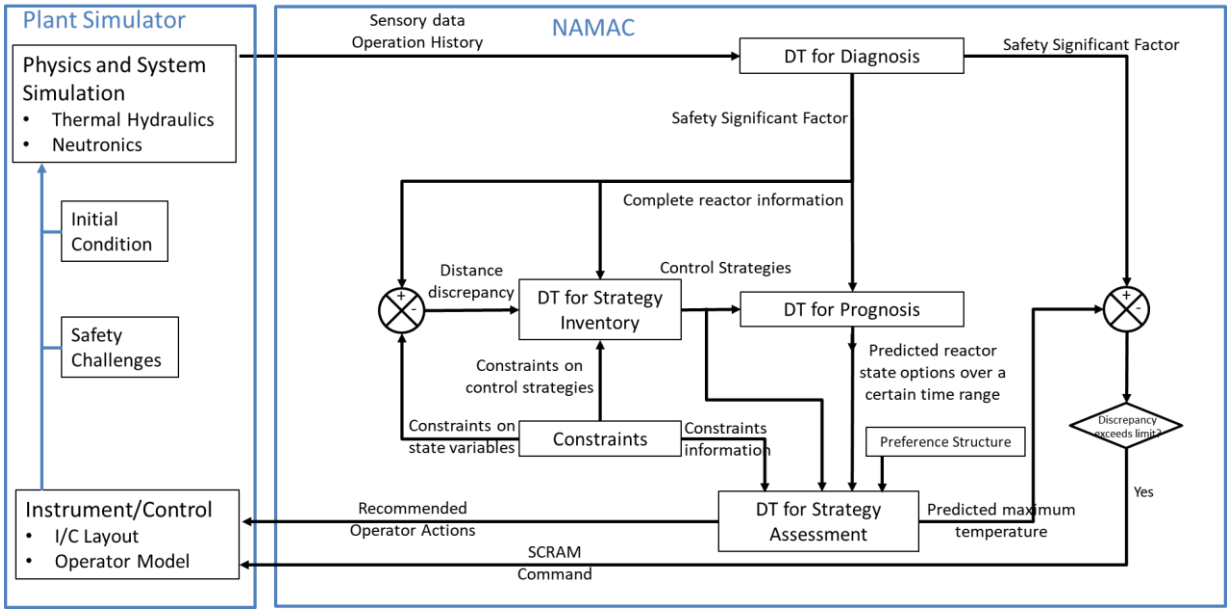


Figure 74. NAMAC System Workflow, Reprinted from 'Development and Assessment of Advanced Algorithms' by L. Lin et al, 2020

B.2. Current Studies for Advanced Control and Management System

Generally, the difference between automatic and autonomous systems is the degree of human involvement in responding. For developing the autonomous system, Krogmann (1999) enhanced the interaction between real environment and the system to achieve predesignated goals by goal-directed behavior [83]. He considered that computational and machine intelligence technique are the prerequisites for a transition from the conventional automation to an autonomous system. The Autonomous Transportable On-demand reactor Module (ATOM) is conceptually designed to eliminate the Fukushima-type accident by combining passive system and autonomous pressurized water reactor (PWR) type small modular reactor (SMR) [84]. It includes a GAIA (Genuinely Autonomous Intelligence for ATOM) system based on deep learning with advanced neural network to obtain operation mode-dependent autonomous strategies. To enhance the autonomous operation, passive and autonomous frequency control operation (PAFO) is developed to control the reactor power by the passive reactor feedback to the changes of the coolant temperature [85]. For the ATOM system, LSTM is used to develop an autonomous control algorithm in which the state of components is predicted from physical variables and component states to control the safety functions [86]. From the Oak Ridge National Laboratory, the supervised control system (SCS) is developed as an autonomous decision-making control system by using

both probabilistic and deterministic approaches from the analysis of the plant status [87]. The system includes a probabilistic decision-making model based on coupled event trees and fault trees, a performance-based system model, and utility theory algorithm to generate a decision. The reactor autonomous control system for a space mission has three hierarchies in its architecture: 'program manager' for decision-making at the top level, 'control coordinator' at the middle level, and 'fault diagnosis, fault tolerant control, and abnormal event control' at the bottom level in which the reactor system interfaces with the control system [88]. The system utilizes the Principal Component Analysis (PCA) for fault detection or sensor failure and a genetic algorithm for the model predictive control. To support the operator during a severe accident, the Severe Accident Management Support Expert (SAMEX) system was developed to provide technical support for emergency response [89]. However, the SAMEX system is not operated by the ML models but by a data base or simulator modules.

In conclusion, the autonomous control system is being studied by using a variety of advanced AI techniques with its improvements. The NAMAC system is distinguished from other autonomous control systems in several aspects: 1) discrete steps with specific objectives, 2) skill-based training for decision-making by using the RL, and 3) a nearly autonomous system that the operator intervenes only in a final decision.

B.3. Technical Components

There are several major technical components in the NAMAC project besides the RNN for the AI-guided diagnosis: 1) Raven-GOTHIC is for data generation, 2) Dynamic Bayesian Network (DBN) for prognosis with probabilistic approach, and 3) RL for evaluation of the action strategy or policies for decision making. In this section, a basic concept of these technical components is introduced.

B.3.1. RAVEN-GOTHIC Simulation

Reactor Analysis and Virtual control Environment (RAVEN) is a multi-purpose framework for simulating the operation action and plant states by using the control logic with the event tree-based analysis [90]. The RAVEN is coupled with deterministic thermal hydraulic code to perform the Dynamic PRA (DPRA) tasks by creating a number of accident scenarios. In the

research, the GOTHIC is used to compute the physical phenomena during the transient due to LOF for training and test data generation.

For the validation of the simplified EBR-II model, the benchmark study was implemented by comparison with SHRT-17 and SHRT-45R program data [40]. The SHRT-17 simulated a protected LOF caused by primary and intermediate pumps' trips with reactor scram. Also, the primary system auxiliary coolant pump was not operated owing to battery power off. Whereas, the SHRT-45R simulated an unprotected LOF caused by primary and intermediate pumps' trips with reactor scram failure because of plant protection system failure, but the battery power supply was continued for the auxiliary pump [91]. In both cases, increased temperatures in the reactor were not threatening. In virtue of natural circulation by inertia of the coolant and thermal expansion, the temperature in the reactor stayed in acceptable level in the SHRT-17 test. Likewise, the natural phenomena and inherent reactivity feedback to terminate the fission process enable the reactor to keep the temperature staying at an acceptable level. As we can see in figure 75, the GOTHIC model simulation data demonstrates the general agreement with SHRT program data in the active core outlet temperature (SHRT-17) and core power (SHRT-45R) [92, 93]. Through the benchmark study, GOTHIC code proves its capability by comparison between simulation results and experiment data of the Shutdown Heat Removal Test in LOF accident.

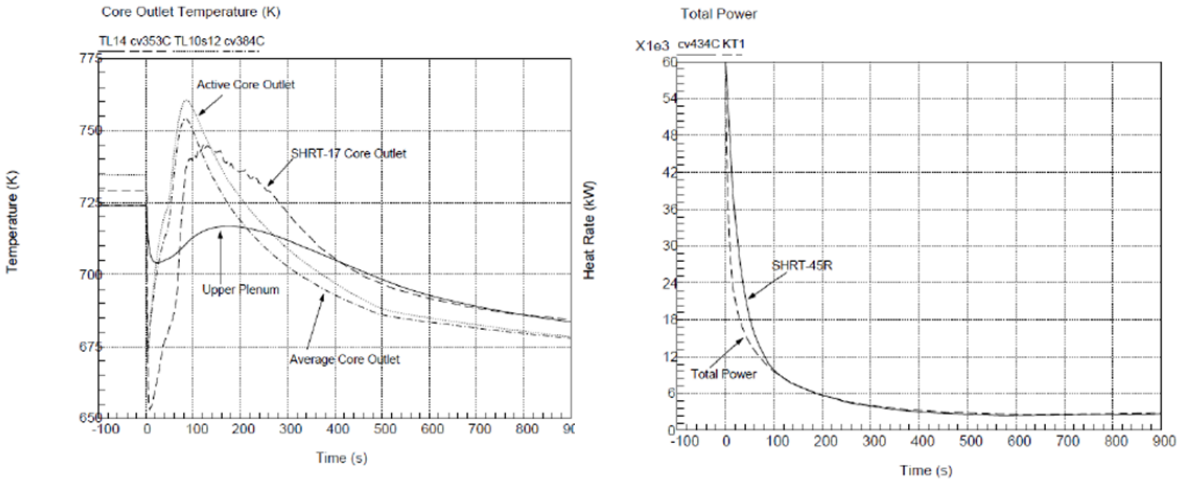


Figure 75. SHRT-17 Core Outlet Temperature (Left) and SHRT-45R Core Power (Right), Reprinted from 'BENCHMARK OF GOTHIC TO EBR-II SHRT-17 AND SHRT-45R TESTS (p. 10 and p. 21)' by J.W Lane et al, 2019, Zachry Nuclear Engineering, Inc.

B.3.2. Dynamic Bayesian Network

The Bayesian Network (BN) is one type of probabilistic graphical models that can be used to represent knowledge and conditional dependencies about an uncertain realm. It is useful for capturing the causal relationships in a complex system. It also allows prediction via Bayesian inference [94]. It is composed of nodes and edges, which refer to random variables and probabilistic relationships between the nodes, respectively. This relationship, given by conditional probability is calculated using stochastic methods. The BN is based on Bayes' rule that posterior probability ($P(A|B)$) is computed when given the prior probability ($P(A)$) and likelihood ($P(B|A)$) that B is materialized when A is true:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \tag{40}$$

where A and B are events and $P(B) \neq 0$.

For risk management, the BN is useful in predicting the accident progression in the complicated system, which includes a reactor system, coolant system and safety system [95].

The BN has two main advantages: ease in visualization and effectiveness in prediction. The event tree is a representative method of a logical model used to carry out the risk analysis by operation states of the safety related system. The consequences are induced from an initial event and passed through several head events step-by-step to describe accident progressions. When comparing to the event tree, the BN illustrates accident features with simpler form. In the event tree, a safety related system is presented as a head event and the branches identify the conditions of head events such as a success or failure, whereas, in the BN, the safety engineering system operation states or the physical states are assigned by states in each node and the causality is specified through the edge. The BN is an economical approach to analyze the accident progression because one combination of node and edge is able to contain two or more branches.

Secondly, the BN is powerful at predicting consequences with its probability when the evidence is set. Since the condition of the safety system or relevant components varies in an accident's progression, a consequence can be changed during an accident. After observable physical variables and available system strategies are determined in the nodes according to the diagnosis result, the probability of the damage states is forecasted by using the BN inference.

The dynamic Bayesian Network (DBN) is defined to by the BN, which considers the sequential data processing to model time-dependent influence [96]. DBN encourages the

calculation of posterior probability over time in each node by assigning the evidence in each time step. Recently, DBN has been used to develop an accident diagnostic support model that aims at inferring the reactor state with a subset of available information to establish Safely Managing Accident Reactor Transients (SMART) Procedure [97 - 99]. Primary tasks to build the DBN are identifying the nodes and edges. A DBN Node would consist of three elements: SSC states, physical properties, and accident scenarios. First of all, an appropriate number of nodes should be prepared to avoid unnecessary computational complexity due to an excessive amount of nodes including meaningless parameter nodes. In the SMART procedure, pertinence of plant parameters has been estimated by Kullback-Leibler (K-L) divergence and cross-validation with analysis of F-scores. Secondly, data processing is required to set up the edges in the DBN. For the prognosis process with probabilistic approach in the NAMAC system, currently, the DBN is used for developing the prognosis model.

B.3.3. Reinforcement Learning

The RL is motivated by psychology of behavior so as to determine an optimized action strategy in a given environment in order to gain the maximized reward [100]. When an agent recognizes the current state through observation, it performs an action according to the state. Then the environment provides rewards and the next states (see Figure 76). There are several elements in the RL algorithm: 1) policy which chooses the action in the state and is the subject for learning, 2) reward that state gives some values to the agent in accordance with the action, and 3) value function which indicates a gross predicted reward. RL uses both exploitation for making the best decision in a given current situation and exploration for gathering more information by random choice to make the optimized decision in the long run. Usually the RL is used to robot legged locomotion, self-driving cars, marketing strategy plan, and so on.

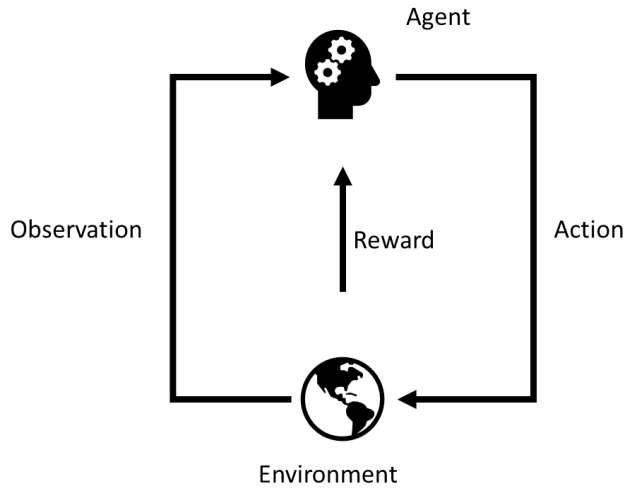


Figure 76. Basic Idea of Reinforcement Learning

As figure 77 shows below, the reward ($R(s)$) and transition probability (P) are considered in Markov Decision Process (MDP) dynamics to calculate the value function ($V^{\pi}(s)$) of policy (π) [101]. The Bellman Equation is a basic equation used to sum up the immediate reward and future reward expectation:

$$\begin{aligned} V^{\pi}(s) &= \mathbb{E}_{\pi}[R(s_0) + \gamma R(s_1) + \gamma^2 R(s_2) + \cdots | s_0 = s] \\ &= \mathbb{E}_{\pi}[R(s_0) + \gamma V^{\pi}(s_{t+1}) | s_0 = s] \end{aligned} \tag{41}$$

where, s is a set of states, such as physical states or symptoms; *R* indicates a reward function; and γ means discount factor in which the value of reward would decrease because it is reasonable to prefer current rewards to later uncertain rewards. The action-value function ($Q^{\pi}(s,a)$) calculates expected return value starting from state *s*, with action *a* and policy (π).

$$Q^{\pi}(s,a) = \mathbb{E}_{\pi}[R(S_0, A_0) + \gamma V^{\pi}(S_{t+1}, A_{t+1}) | S_0 = s, A_0 = a] \tag{42}$$

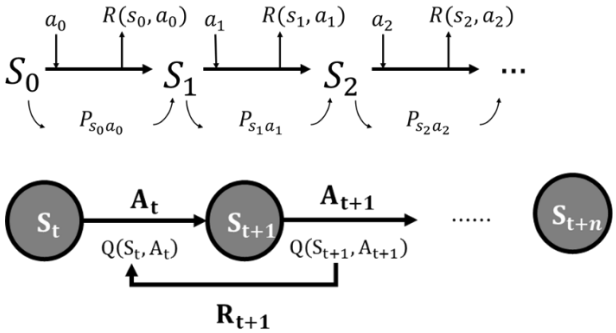


Figure 77. Markov Decision Process Dynamics

The RL training is implemented to update the policy that induces the optimized action through policy iteration. In the RL training process (see figure 78), policy, which is initialized randomly, is repeatedly updated until convergence:

$$\pi(s) = arg \max_{a \in A} \gamma \sum_{s' \in S} P_{sa(s)}(s') V^*(s), V = V^{\pi}$$

This is a preprocess to apply to prognosis with a specific action execution. Thus, the DBN in combination with the RL is suggested to find an optimal sequence of decisions [2]. To complete the RL model for risk-informed decision making, the issues following must be solved:

- ✓ Reward issue: What is the condition to get a reward? How can we set up the reward with discount factor and penalty during accident progression?
- ✓ State issue: What becomes of the state or environment that reflects the risk index for the risk-informed decision making?
- ✓ Algorithm type issue: What kind of algorithm type would be appropriate? There are some suggestions such as policy gradient, Deep Q-learning, and so on.

The RL model to provide an optimal policy for the NAMAC system is currently under development.

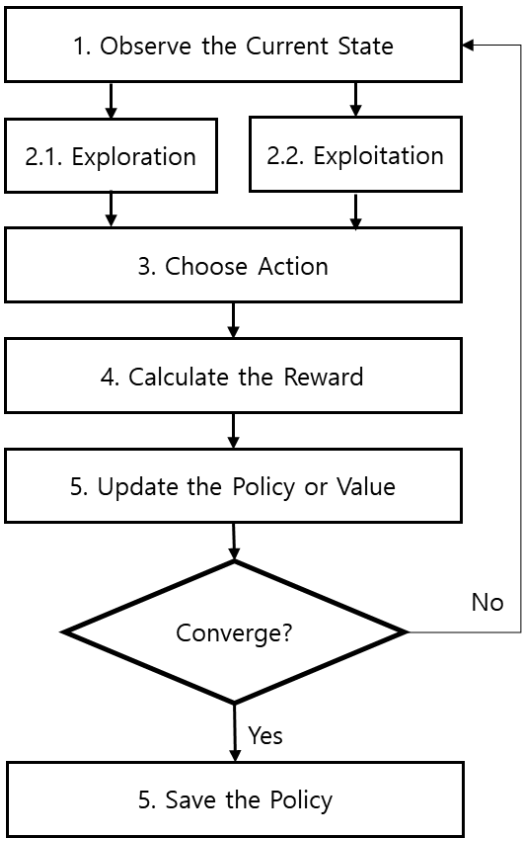


Figure 78. Reinforcement Learning Algorithm Training Process

APPENDIX C. TESTING RESULTS

In this appendix, the results from the SSFIM are shown according to the models.

Table 39. Type 1 Test Result in Transient A

Model Name	Validation Loss	Type 1 Test Data	
		RMSE in FCL	MAPE (%)
A_t9	8.726E+00	2.945E+01	2.927E+00
A_t14	8.580E-02	1.779E+00	1.387E-01
A_fl1	4.450E-02	1.505E+00	1.437E-01
A_fl6	8.654E-02	2.106E+00	1.946E-01
A_fl11	4.420E-02	1.569E+00	1.512E-01
A_pr8	4.510E-02	1.503E+00	1.519E-01
A_t914	5.600E-02	1.555E+00	1.282E-01
A_t9_fl1	4.250E-02	1.526E+00	1.593E-01
A_t9_fl6	5.960E-02	3.214E+00	4.245E-01
A_t9_fl11	4.230E-02	1.655E+00	1.514E-01
A_t9_pr8	4.180E-02	1.456E+00	1.264E-01
A_t14_fl1	4.280E-02	1.479E+00	1.270E-01
A_t14_fl6	2.476E-01	1.608E+00	1.812E-01
A_t14_fl11	4.300E-02	1.548E+00	1.401E-01
A_pr8_fl1	4.630E-02	1.484E+00	1.412E-01
A_fl16	3.900E-02	1.515E+00	1.446E-01
A_t91416	5.100E-02	1.871E+00	1.799E-01
A_t914_fl1	4.170E-02	1.497E+00	1.421E-01
A_t914_fl6	5.230E-02	1.514E+00	1.703E-01
A_t914_fl11	3.510E-02	1.554E+00	1.518E-01
A_t914_pr8	3.810E-02	1.531E+00	1.471E-01
A_t9_fl16	4.370E-02	1.530E+00	1.503E-01
A_t9_pr8_fl1	4.330E-02	1.497E+00	1.494E-01
A_t14_fl16	3.533E-02	1.523E+00	1.502E-01
A_pr8_fl16	4.260E-02	1.485E+00	1.509E-01
A_fl1611	4.410E-02	1.522E+00	1.415E-01
A_T4	5.250E-02	1.845E+00	1.665E-01
A_t914_fl16	4.560E-02	1.530E+00	1.532E-01
A_t914_pr8_fl16	4.280E-02	1.519E+00	1.546E-01
A_T4_FL3	3.690E-02	1.748E+00	1.684E-01
A_T8	5.140E-02	1.811E+00	1.500E-01
A_T4_pr8_FL3	4.010E-02	1.535E+00	1.536E-01
A_T8_FL3	3.600E-02	1.703E+00	1.687E-01
A_T8_pr8_FL3	4.110E-02	1.509E+00	1.504E-01

Table 40. Type 1 Test Result in Transient B

Model Name	Validation Loss	Type 1 Test Data	
		RMSE in FCL	MAPE (%)
B_t9	2.138E+00	9.659E+00	1.309E+00
B_t14	8.900E-03	1.174E+00	1.345E-01
B_fl1	9.200E-03	9.755E-01	1.057E-01
B_fl6	1.290E-02	1.001E+00	1.001E-01
B_fl11	8.600E-03	8.032E-01	9.275E-02
B_pr8	8.100E-03	6.512E-01	7.382E-02
B_t914	8.300E-03	1.050E+00	1.236E-01
B_t9_fl1	6.100E-03	8.807E-01	1.002E-01
B_t9_fl6	9.700E-03	8.788E-01	9.445E-02
B_t9_fl11	7.600E-03	6.719E-01	6.638E-02
B_t9_pr8	7.400E-03	7.332E-01	8.614E-02
B_t14_fl1	6.500E-03	7.868E-01	9.120E-02
B_t14_fl6	9.500E-03	8.812E-01	9.906E-02
B_t14_fl11	7.200E-03	6.878E-01	7.291E-02
B_pr8_fl1	6.800E-03	6.823E-01	7.867E-02
B_fl16	1.030E-02	9.487E-01	9.915E-02
B_t91416	9.200E-03	1.242E+00	1.889E-01
B_t914_fl1	7.900E-03	8.233E-01	9.779E-02
B_t914_fl6	3.000E-02	1.958E+00	1.862E-01
B_t914_fl11	7.000E-03	7.549E-01	8.579E-02
B_t914_pr8	9.100E-03	8.420E-01	1.002E-01
B_t9_fl16	9.800E-03	8.874E-01	9.827E-02
B_t9_pr8_fl1	7.800E-03	7.334E-01	8.964E-02
B_t14_fl16	5.600E-03	7.844E-01	9.238E-02
B_pr8_fl16	8.100E-03	6.656E-01	6.813E-02
B_fl1611	8.000E-03	6.740E-01	6.867E-02
B_T4	8.700E-03	8.942E-01	1.091E-01
B_t914_fl16	9.600E-03	8.384E-01	9.856E-02
B_t914_pr8_fl16	7.100E-03	7.607E-01	9.201E-02
B_T4_FL3	7.500E-03	7.599E-01	8.878E-02
B_T8	7.800E-03	1.035E+00	1.224E-01
B_T4_pr8_FL3	6.900E-03	6.866E-01	7.204E-02
B_T8_FL3	8.000E-03	9.077E-01	1.143E-01
B_T8_pr8_FL3	8.200E-03	8.119E-01	1.035E-01

Table 41. Type 1 Test Result in Transient C

Model Name	Validation Loss	Type 1 Test Data	
		RMSE in FCL	MAPE (%)
C_t9	2.634E-01	1.561E+01	1.721E+00
C_t14	2.070E-02	3.469E+00	3.822E-01
C_fl1	1.170E-02	1.667E+00	1.752E-01
C_fl6	1.560E-02	3.166E+00	2.436E-01
C_fl11	1.050E-02	1.711E+00	2.000E-01
C_pr8	9.400E-03	1.767E+00	2.018E-01
C_t914	2.140E-02	3.414E+00	3.722E-01
C_t9_fl1	1.280E-02	1.934E+00	2.085E-01
C_t9_fl6	9.900E-03	4.697E+00	3.052E-01
C_t9_fl11	1.000E-02	1.856E+00	2.060E-01
C_t9_pr8	1.000E-02	1.772E+00	2.153E-01
C_t14_fl1	1.240E-02	2.141E+00	2.315E-01
C_t14_fl6	1.780E-02	9.548E+00	7.137E-01
C_t14_fl11	9.300E-03	1.956E+00	2.225E-01
C_pr8_fl1	9.900E-03	1.699E+00	1.953E-01
C_fl16	9.500E-03	1.737E+00	1.994E-01
C_t91416	1.370E-02	2.445E+00	2.699E-01
C_t914_fl1	1.200E-02	1.801E+00	1.824E-01
C_t914_fl6	1.180E-02	2.077E+00	2.533E-01
C_t914_fl11	9.800E-03	2.203E+00	2.476E-01
C_t914_pr8	9.300E-03	2.008E+00	2.431E-01
C_t9_fl16	9.100E-03	1.728E+00	1.985E-01
C_t9_pr8_fl1	1.060E-02	1.672E+00	1.886E-01
C_t14_fl16	9.400E-03	1.786E+00	2.045E-01
C_pr8_fl16	1.030E-02	1.669E+00	1.924E-01
C_fl1611	1.090E-02	1.816E+00	1.991E-01
C_T4	1.350E-02	3.312E+00	3.667E-01
C_t914_fl16	1.200E-02	2.674E+00	2.825E-01
C_t914_pr8_fl16	1.050E-02	1.868E+00	2.319E-01
C_T4_FL3	8.400E-03	1.938E+00	2.306E-01
C_T8	1.390E-02	3.018E+00	3.475E-01
C_T4_pr8_FL3	9.900E-03	1.749E+00	2.087E-01
C_T8_FL3	1.060E-02	2.085E+00	2.491E-01
C_T8_pr8_FL3	1.040E-02	1.757E+00	1.998E-01

Table 42. Type 2 and 3 Test Results in Transient A

Model Name	Type 2 Test Data		Type 3 Test Data	
	RMSE in FCL	MAPE (%)	RMSE in FCL	MAPE (%)
A_t9	2.947E+01	2.931E+00	2.946E+01	2.929E+00
A_t14	6.313E+01	5.518E+00	5.526E+01	3.872E+00
A_fl1	2.324E+00	2.705E-01	8.735E+00	9.281E-01
A_fl6	2.927E+00	3.719E-01	2.872E+00	2.974E-01
A_fl11	3.803E+00	4.137E-01	1.501E+01	1.591E+00
A_pr8	3.250E+00	3.669E-01	5.160E+01	6.248E+00
A_t914	1.048E+02	1.289E+01	1.124E+02	1.132E+01
A_t9_fl1	2.543E+00	3.003E-01	8.347E+00	9.277E-01
A_t9_fl6	3.441E+00	4.078E-01	9.571E+00	6.246E-01
A_t9_fl11	4.400E+00	4.674E-01	1.549E+01	1.618E+00
A_t9_pr8	3.776E+00	4.309E-01	4.220E+01	4.847E+00
A_t14_fl1	2.656E+00	3.118E-01	1.088E+01	1.197E+00
A_t14_fl6	7.133E+00	7.652E-01	6.725E+00	7.563E-01
A_t14_fl11	3.791E+00	4.311E-01	1.224E+01	1.237E+00
A_pr8_fl1	2.476E+00	2.910E-01	2.757E+01	2.724E+00
A_fl16	3.245E+00	3.720E-01	1.105E+01	1.178E+00
A_t91416	8.692E+01	9.508E+00	4.225E+01	4.136E+00
A_t91421	9.875E+01	9.270E+00	5.815E+01	4.740E+00
A_t914_fl1	2.996E+00	3.514E-01	1.087E+01	1.191E+00
A_t914_fl6	1.542E+01	1.647E+00	1.862E+01	2.120E+00
A_t914_fl11	4.880E+00	5.507E-01	1.556E+01	1.609E+00
A_t914_pr8	8.916E+00	9.785E-01	4.337E+01	4.800E+00
A_t9_fl16	2.567E+00	3.022E-01	9.007E+00	9.711E-01
A_t9_pr8_fl1	2.295E+00	2.677E-01	2.259E+01	2.355E+00
A_t14_fl16	4.550E+00	5.075E-01	1.478E+01	1.489E+00
A_pr8_fl16	2.562E+00	2.997E-01	2.763E+01	2.762E+00
A_fl1611	2.396E+00	2.745E-01	8.193E+00	9.211E-01
A_T4	7.968E+01	8.288E+00	3.610E+01	3.161E+00
A_t914_fl16	2.832E+00	3.234E-01	9.839E+00	1.063E+00
A_t914_pr8_fl16	2.823E+00	3.297E-01	2.771E+01	2.702E+00
A_T4_FL3	4.613E+00	5.288E-01	1.480E+01	1.513E+00
A_T8	5.367E+01	4.760E+00	3.385E+01	2.836E+00
A_T4_pr8_FL3	2.529E+00	2.935E-01	1.622E+01	1.757E+00
A_T8_FL3	4.730E+00	5.379E-01	1.492E+01	1.571E+00
A_T8_pr8_FL3	2.986E+00	3.470E-01	2.005E+01	1.927E+00

Table 43. Type 2 and 3 Test Results in Transient B

Model Name	Type 2 Test Data		Type 3 Test Data	
	RMSE in FCL	MAPE (%)	RMSE in FCL	MAPE (%)
B_t9	9.661E+00	1.308E+00	9.650E+00	1.308E+00
B_t14	3.754E+01	4.296E+00	1.937E+01	2.082E+00
B_fl1	1.434E+00	1.799E-01	4.942E+00	5.744E-01
B_fl6	4.099E+00	4.363E-01	1.967E+01	1.964E+00
B_fl11	2.427E+00	3.002E-01	1.065E+01	1.247E+00
B_pr8	1.906E+00	2.254E-01	3.245E+01	3.887E+00
B_t914	1.070E+02	1.109E+01	8.240E+01	6.566E+00
B_t9_fl1	6.073E+00	4.428E-01	8.927E+00	9.119E-01
B_t9_fl6	6.806E+00	5.606E-01	2.545E+01	2.379E+00
B_t9_fl11	2.703E+00	3.299E-01	1.208E+01	1.462E+00
B_t9_pr8	2.168E+00	2.681E-01	3.240E+01	3.851E+00
B_t14_fl1	4.305E+00	3.909E-01	6.321E+00	6.860E-01
B_t14_fl6	6.120E+00	7.247E-01	3.421E+01	3.964E+00
B_t14_fl11	3.601E+00	4.320E-01	1.308E+01	1.580E+00
B_pr8_fl1	1.887E+00	2.251E-01	2.950E+01	3.574E+00
B_fl16	1.761E+00	1.981E-01	7.047E+00	8.498E-01
B_t91416	3.187E+01	3.805E+00	1.700E+01	1.921E+00
B_t91421	1.662E+01	1.851E+00	7.707E+00	8.773E-01
B_t914_fl1	3.782E+00	3.149E-01	6.342E+00	7.679E-01
B_t914_fl6	2.078E+00	2.084E-01	2.916E+00	3.649E-01
B_t914_fl11	3.384E+00	4.112E-01	1.171E+01	1.410E+00
B_t914_pr8	2.565E+00	3.226E-01	3.554E+01	4.326E+00
B_t9_fl16	1.971E+00	1.980E-01	3.442E+00	4.043E-01
B_t9_pr8_fl1	2.820E+00	3.288E-01	1.288E+02	1.313E+01
B_t14_fl16	6.915E+00	7.462E-01	2.836E+01	2.965E+00
B_pr8_fl16	2.079E+00	2.635E-01	2.362E+01	2.761E+00
B_fl1611	2.508E+00	3.130E-01	1.182E+01	1.409E+00
B_T4	6.271E+01	7.355E+00	1.781E+01	2.047E+00
B_t914_fl16	4.711E+00	4.198E-01	6.931E+00	6.898E-01
B_t914_pr8_fl16	4.131E+00	5.090E-01	2.631E+01	3.192E+00
B_T4_FL3	5.450E+00	6.470E-01	1.530E+01	1.813E+00
B_T8	3.898E+01	4.716E+00	2.332E+01	2.508E+00
B_T4_pr8_FL3	4.873E+00	4.895E-01	1.587E+01	1.920E+00
B_T8_FL3	4.611E+00	5.590E-01	1.365E+01	1.627E+00
B_T8_pr8_FL3	2.540E+00	3.088E-01	1.684E+01	1.935E+00

Table 44. Type 2 and 3 Test Results in Transient C

Model Name	Type 2 Test Data		Type 3 Test Data	
	RMSE in FCL	MAPE (%)	RMSE in FCL	MAPE (%)
C_t9	1.566E+01	1.725E+00	1.560E+01	1.719E+00
C_t14	7.747E+01	8.478E+00	4.305E+01	4.606E+00
C_fl1	1.843E+00	2.103E-01	3.594E+00	4.001E-01
C_fl6	4.824E+00	3.280E-01	8.195E+00	7.018E-01
C_fl11	2.426E+00	2.852E-01	8.852E+00	1.021E+00
C_pr8	3.068E+00	3.734E-01	3.343E+01	3.680E+00
C_t914	5.863E+01	6.572E+00	4.510E+01	4.821E+00
C_t9_fl1	2.345E+00	2.502E-01	3.791E+00	4.104E-01
C_t9_fl6	6.204E+00	4.249E-01	8.867E+00	8.541E-01
C_t9_fl11	2.658E+00	3.204E-01	8.877E+00	1.020E+00
C_t9_pr8	2.880E+00	3.490E-01	3.680E+01	4.196E+00
C_t14_fl1	2.697E+00	2.922E-01	4.846E+00	4.745E-01
C_t14_fl6	5.801E+00	4.087E-01	8.656E+00	7.671E-01
C_t14_fl11	3.548E+00	4.189E-01	8.580E+00	9.864E-01
C_pr8_fl1	3.455E+00	4.195E-01	3.363E+01	3.704E+00
C_fl16	3.080E+00	3.725E-01	1.319E+01	1.454E+00
C_t91416	4.428E+01	4.953E+00	2.021E+01	2.148E+00
C_t91421	1.924E+01	2.246E+00	1.914E+01	2.238E+00
C_t914_fl1	2.929E+00	2.476E-01	3.881E+00	3.666E-01
C_t914_fl6	3.055E+00	3.034E-01	9.480E+00	8.727E-01
C_t914_fl11	3.896E+00	4.840E-01	8.485E+00	9.739E-01
C_t914_pr8	3.883E+00	4.674E-01	4.266E+01	4.825E+00
C_t9_fl16	2.475E+00	3.026E-01	6.930E+00	7.712E-01
C_t9_pr8_fl1	4.970E+00	6.058E-01	3.074E+01	3.574E+00
C_t14_fl16	4.463E+00	5.074E-01	1.153E+01	1.264E+00
C_pr8_fl16	2.341E+00	2.832E-01	2.339E+01	2.585E+00
C_fl1611	2.308E+00	2.737E-01	7.944E+00	9.167E-01
C_T4	6.362E+01	7.108E+00	2.455E+01	2.674E+00
C_t914_fl16	3.696E+00	4.241E-01	8.532E+00	9.752E-01
C_t914_pr8_fl16	3.271E+00	3.850E-01	2.346E+01	2.661E+00
C_T4_FL3	3.856E+00	4.577E-01	8.172E+00	9.381E-01
C_T8	4.765E+01	5.302E+00	2.430E+01	2.712E+00
C_T4_pr8_FL3	3.235E+00	3.833E-01	2.645E+01	2.844E+00
C_T8_FL3	3.318E+00	4.055E-01	7.466E+00	8.683E-01
C_T8_pr8_FL3	2.857E+00	3.411E-01	2.032E+01	2.224E+00

Table 45. Mapping Inference Results

MAPE (%)	B_fl1 Model			B_t9_fl16 Model		
	Type 1	Type 2	Type 3	Type 1	Type 2	Type 3
TL1s45	1.063E-02	1.062E-02	3.518E-01	1.044E-02	1.229E-02	1.539E-02
TL2s2	7.755E-02	7.637E-02	2.401E+00	7.692E-02	8.460E-02	9.767E-02
TL3s2	7.827E-02	7.649E-02	2.561E+00	7.877E-02	8.682E-02	9.960E-02
TL4s4	7.853E-02	7.702E-02	2.431E+00	7.835E-02	8.641E-02	9.911E-02
TL5s2	7.794E-02	7.762E-02	2.646E+00	7.684E-02	8.675E-02	1.087E-01
TL6s2	7.806E-02	7.773E-02	2.636E+00	7.696E-02	8.675E-02	1.081E-01
TL7s4	7.840E-02	7.809E-02	2.604E+00	7.735E-02	8.679E-02	1.068E-01
TL8s1	7.716E-02	7.625E-02	2.606E+00	7.601E-02	8.467E-02	1.011E-01
TL9s1	7.598E-02	7.650E-02	2.122E+00	-	-	-
TL10s1	7.716E-02	7.625E-02	2.601E+00	7.601E-02	8.463E-02	1.010E-01
TL10s6	1.435E-01	2.246E-01	8.424E+00	1.318E-01	1.009E+00	1.912E+00
TL10s12	1.809E-01	3.067E-01	1.074E+01	1.678E-01	1.455E+00	2.770E+00
TL11s1	7.710E-02	7.618E-02	2.607E+00	7.595E-02	8.446E-02	1.006E-01
TL11s5	1.567E-01	2.573E-01	9.377E+00	1.439E-01	1.178E+00	2.229E+00
TL11s9	1.627E-01	2.844E-01	1.015E+01	1.482E-01	1.338E+00	2.548E+00
TL12s1	8.653E-02	8.774E-02	1.263E+00	8.539E-02	1.187E-01	1.990E-01
TL12s4	1.194E-01	1.258E-01	9.028E+00	1.207E-01	4.381E-01	7.927E-01
TL12s6	1.609E-01	1.608E-01	1.229E+01	1.596E-01	4.620E-01	8.080E-01
TL13s1	7.580E-02	7.631E-02	1.873E+00	7.379E-02	7.968E-02	9.157E-02
TL13s4	1.536E-01	1.630E-01	9.227E+00	1.546E-01	5.265E-01	9.513E-01
TL13s8	2.991E-01	2.934E-01	1.792E+01	2.981E-01	6.025E-01	1.019E+00
TL14s1	1.967E-01	2.797E-01	5.776E+00	1.998E-01	1.233E+00	2.331E+00
TL15s3	2.201E-01	2.934E-01	5.705E+00	2.282E-01	1.208E+00	2.270E+00
TL16s5	2.924E-01	3.044E-01	6.238E+00	2.937E-01	5.154E-01	9.000E-01
TL17s1	4.045E-04	4.229E-04	9.060E-03	4.519E-04	6.289E-04	1.040E-03
TL18s2	4.216E-04	4.382E-04	8.890E-03	4.723E-04	6.455E-04	1.060E-03
TL19s5	2.259E-01	2.515E-01	5.692E+00	2.263E-01	5.903E-01	1.113E+00
TL20s1	2.444E-01	2.538E-01	5.926E+00	2.398E-01	4.925E-01	8.377E-01
TL21s1	2.453E-01	2.546E-01	5.969E+00	2.406E-01	4.928E-01	8.380E-01
Avg	1.294E-01	1.516E-01	5.213E+00	1.213E-01	4.045E-01	8.053E-01