ABSTRACT

LIU YANG. Development of a Data-Driven Analysis Framework for Boiling Problems with Multiphase-CFD Solver (Under the direction of Dr. Nam Dinh).

Flow boiling is a highly efficient heat transfer regime, which is used for thermal management in various engineered systems. Among the modeling tools for boiling, the Multiphase Computational Fluid Dynamics (MCFD) solver based on Eulerian-Eulerian two-fluid model has demonstrated its potential in solving boiling problems for industrial applications. On the other hand, in two-fluid model, closure relations are needed to make the two-fluid conservation equations solvable. Such relations, usually empirical or semi-empirical correlations, bring model form uncertainty and model parameter uncertainty to the MCFD solver. A still open issue for MCFD is that such uncertainties can be significant and are still not well quantified, thus undermining the predictive capability of the solver.

This dissertation presents a data-driven analysis framework to address this open issue. The framework aims to leverage state of the art statistical methods and the increasingly affluent boiling data, from both high-resolution experimental measurements and high-fidelity simulations, to 1). perform validation and uncertainty quantification (VUQ) for the MCFD solver based on all available datasets; 2). develop data-driven closure relations based on deep neural networks for the MCFD solver that has the better predictive capability. Three major products are developed within the framework.

First, a boiling data processing and storage procedure is developed for high-resolution experiments and high-fidelity simulations. The extracted data are stored in a structured manner to ensure the flexibility for multipurpose usage. Second, a comprehensive validation and uncertainty quantification (VUQ) procedure is developed for the MCFD solver. The procedure quantifies the uncertainties of MCFD solver predictions using Bayesian inference; then calculates validation metrics that quantitatively measuring the agreement between experimental measurement and obtained prediction uncertainties. Last, a study of new boiling closure relation development based on deep learning is performed. The deep feedforward networks trained by high fidelity boiling simulation data are found to be capable of predicting wall boiling heat transfer behavior with good accuracy.

Development of a Data-Driven Analysis Framework for Boiling Problems with Multiphase-CFD Solver

by
Yang Liu

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

2018

APPROVED BY:

_____	_____
Dr. Nam Dinh Committee Chair	Dr. Ralph Smith
_____	_____
Dr. Joseph Doster	Dr. Igor Bolotnov

DEDICATION

To my parents

BIOGRAPHY

Yang Liu was born in China, Hunan province. He was educated through high school in his hometown. He obtained his college education at Tsinghua University in Beijing, from where he received his bachelor's degree in nuclear engineering in 2010. After that, he obtained his master's degree in nuclear engineering from China Institute of Atomic Energy in 2013.

He began his graduate study in nuclear engineering at North Carolina State University in 2014, under the supervision of Professor Nam Dinh. His research focuses on applying data-driven methodology to boiling problems, including boiling images processing, validation and uncertainty quantification for Multiphase-CFD solver, and data-driven modeling for boiling heat transfer using deep learning algorithms.

ACKNOWLEDGMENTS

The work presented in this dissertation is the outcome of a challenging yet rewarding intellectual experience. First, I would like to thank my advisor, Professor Nam Dinh, for his encouragement, support, and guidance during the course of this research. He lighted up the path towards the idea of data-driven approach for me. He also pushed me to become a more disciplined thinker. I'm truly grateful for the freedom he gave me to explore the specific projects in which I found interest. He has been and will always be a role model for me.

I would also like to express my gratitude to my dissertation committee, Professor Ralph Smith, Professor Joseph Doster, and Professor Igor Bolotnov. They have contributed many insightful perspectives for which I deeply value as a crucial part of my graduate experience.

I owe a great debt of gratitude to Carolyn Coyle and Professor Jacopo Buongiorno (Massachusetts Institute of Technology) who provided their high-quality boiling data, and to Dr. Yohei Sato and Dr. Bojan Niceno (Paul Scherrer Institute) who provided their high-fidelity simulation results. Without the data they provided, this data-driven analysis framework would become a castle in the air.

My work would not have been possible without the financial support from the Consortium for Advanced Simulation of Light Water Reactors (CASL).

Finally, I want to thank my friends, Chih-Wei Chang, Han Bao, Linyu Lin and many others. Without them, my life in Raleigh would not have been as much fun and memorable.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

NOMENCLATURE

a	effective bubble area factor
A_a	interfacial area concentration, 1/m
A_b	effective bubble area fraction
C_d	drag coefficient
C_l	lift coefficient
C_{wl}	wall lubrication coefficient
C_p	specific heat at constant pressure, J/(kg·K)
C_{vm}	virtual mass coefficient
D_d	bubble departure diameter, m
D_S	bubble diameter in bulk flow, m
d_1	departure diameter constant, m/rad
e	bubble growth waiting time factor
E	turbulence wall function Log law offset
f_d	bubble departure frequency, 1/s
$\boldsymbol{g}$	Gravity vector, m/s^2
h	specific enthalpy, J/kg
h_{fg}	latent heat of evaporation, J/kg
h_l	forced convective heat transfer coefficient, W/(m^2·K)
h_{lg}	interfacial heat transfer coefficient, W/(m^2·K)
$\boldsymbol{M}$	interfacial force, N/m^3
N_a	nucleation site density, 1/m^2
N_{avg}	averaged Nucleation site density coefficient, 1/m^2
$\boldsymbol{n_w}$	unit vector normal to the wall
Pr^t	turbulent Prandtl number
p	pressure, Pa
q_{wall}	wall heat flux, W/m^2
q_{Ev}	evaporation heat flux, W/m^2
q_{Qu}	quenching heat flux, W/m^2
q_{Fc}	forced convective heat flux, W/m^2
T	temperature, K
t	time, s
$\boldsymbol{U}$	velocity, m/s
y_w	near wall distance, m

y^+	dimensionless wall distance
A_a	interfacial area concentration, 1/m
A_b	effective bubble area fraction
C_d	drag coefficient
C_l	lift coefficient
C_{wl}	wall lubrication coefficient
C_p	specific heat at constant pressure, J/(kg·K)
C_{vm}	virtual mass coefficient
D_d	bubble departure diameter, m
D_S	bubble diameter in bulk flow, m
f_d	bubble departure frequency, 1/s
$\boldsymbol{g}$	gravity vector, m/s^2
h	specific enthalpy, J/kg
h_{fg}	latent heat of evaporation, J/kg
h_l	forced convective heat transfer coefficient, W/(m^2·K)
h_{lg}	interfacial heat transfer coefficient, W/(m^2·K)
$\boldsymbol{M}$	interfacial force, N/m^3

Greek symbols

α	void fraction
Γ_{ki}	evaporation/condensation rate per volume, kg/(m^3·s)
θ	contact angle, rad
σ	surface tension, kg/s^2
σ^t	turbulent dispersion coefficient
κ	von Karman constant
λ	thermal conductivity, W/(m·K)
μ	dynamic viscosity, Pa·s
μ_{con}	contact angle scaler, 1/rad
υ	kinematic viscosity, m^2/s
ρ	density, kg/m^3
τ	stress tensor, kg/(m·s^2)
Φ	source term of interfacial area concentration, 1/m^3

Subscript

g	gas phase

r	relative motion
sat	saturation
sub	subcooling
sup	superheat
l	liquid phase

Superscript

t	turbulence
$*$	non-dimensional

CHAPTER 1. INTRODUCTION

1.1 Background

Two-phase flow and boiling heat transfer occurs in many situations and are used for thermal management in various engineered systems with high energy density, from power electronics to heat exchangers in power plants and nuclear reactors. Essentially, two-phase and boiling heat transfer is a complex multiphysics process, which involves different interactions between heated solid surface, liquid, and vapor, including nucleation, evaporation, condensation, interfacial mass/heat/momentum exchange, and interface topological change (such as bubble breakup or coalescence). Correspondingly, there are various quantities of interest (QoIs) of a given boiling system, such as flow pattern, pressure drop, wall heat transfer, void fraction distribution, phasic temperature and velocity distribution, etc. In nuclear engineering, understanding the relevant phenomena and accurately predicting the QoIs involve in two-phase flow and boiling heat transfer is crucial for the design of an efficient and safe reactor.

The measurement of many phenomena related to two-phase flow and boiling heat transfer is very challenging with current experimental techniques. Moreover, the experimental study of two-phase flow and boiling at prototypical reactor scale is technically challenging and expensive. Thus in current practices, the design and safety analysis of reactor thermal hydraulics systems are highly dependent on the scientific modeling and simulation.

The modeling of two-phase flow and boiling heat transfer for engineering purpose can be characterized by two aspects: the dimension and the treatment of interfacial interaction. The system code, such as TRACE [1] and COBRA-TF [2] deals with one-dimension cross-sectional averaged problems, while multiphase computational fluid dynamics (MCFD) code deals with multi-dimensional problems. The treatment of interfacial interaction has two types: the mixture model and the Eulerian-Eulerian two-fluid model.

The mixture model treats two phases as a mixture whose property is averaged from vapor and liquid. Thus only one set of conservative equations for mass, momentum, and

energy is needed. The mixture model is further divided, based on their treatment of mechanical non-equilibrium (the relative velocity between the two phases). The homogeneous equilibrium model assumes there is no relative velocity between the phases. The slip factor model uses empirical correlations for the slip ratio (which is defined by the ratio of vapor velocity to liquid velocity). The drift-flux model uses kinematic constitutive equations to describe the relative flow. Theoretically, all three models can be applied to both one-dimension cross-sectional averaged problem and multi-dimensional problems. In practice, however, these models are mainly used for one-dimension cross-sectional averaged problems, such as in system analysis and engineering calculations.

The two-fluid model treats two phases by separate sets of field conservation equations. For one-dimension cross-sectional averaged formulation, the interfacial interactions are modeled in a correspondingly coarse manner. Notably, the effect of local interactions, e.g., wall boiling heat transfer, is coarsened as a source term that affects the cross-sectional averaged parameters of flow dynamics. In contrast, the multi-dimensional formulation requires a much larger set of closure relations needed to provide detailed modeling of interfacial interactions and wall heat transfer.

There are two common features of those modeling approaches: 1). the interface between two phases is averaged and is not resolved in the simulation; 2). closure relations are required to make the conservation equations of the model solvable. Among those approaches, the MCFD solver based on Eulerian-Eulerian two-fluid model has been regarded as the most promising tool, especially for applications with complex geometries such as reactor fuel rod bundles. The main reason is MCFD has the capability to describe phenomena with local detail, while still retain relative computational efficiency. Based on this reason, MCFD attracts increasingly interests over recent two decades. Simulations are developed based on it, from adiabatic bubbly flow and boiling simulation [3, 4] to critical heat flux prediction [5, 6].

On the other hand, it should be recognized that fundamental disconnection still exists between performing simulations using MCFD solver and applying MCFD solver for industrial applications. The main issue is the uncertainties of the MCFD solver are not well

quantified, especially for the uncertainty introduced by the closure relations. As pointed out by Roy and Oberkampf [7]:

Information on the magnitude, composition, and sources of uncertainty in simulations is critical in the decision-making process for natural and engineered systems. Without forthrightly estimating and clearly presenting the total uncertainty in a prediction, decision makers will be ill advised, possibly resulting in inadequate safety, reliability, or performance of the system. Consequently, decision makers could unknowingly put at risk their customers, the public, or the environment.

Compare to the single-phase CFD, MCFD has a significant uncertainty source from the closure relations. Those closure relations are introduced to describe the unresolved phenomena in the two-fluid model, including boiling process and the interfacial interaction. There are two issues for applying those closure relations in a MCFD solver. Firstly, most of those closure relations are empirical or semi-empirical correlations with empirical parameters whose values significantly influence the results of the solver, yet the values of those parameters can vary significantly between different practices. Secondly, the closure relations are proposed in a manner that one closure relation deals with one physical phenomenon, a group of closure relations is then assembled for the whole system. Such "divide-and-conquer" approach neglects the possible interactions between different closure relations.

Many efforts have been made to address these two issues. These efforts can be characterized into two interrelated categories. The first category is to evaluate the uncertainty of closure relations through uncertainty quantification (UQ) and evaluate the influence of the uncertainties on the solver predictions through validation. A most straightforward and widely used approach is to select different closure relations, in many cases also randomly sampling the corresponding empirical parameters, and run multiple simulations, then graphically compare the simulation results with the experimental measurement [8-10]. Based on these comparisons, a set of "optimized" parameters and closure relations are selected for future usage. This approach can produce reasonably good results with a limited data support. However, it is heavily dependent on the researcher's experience and is generally providing *ad hoc* result.

A more comprehensive framework for the validation and uncertainty quantification (VUQ) of a solver was formulated in the late 1990s [11]. Later, an improved version was developed [7]. This framework includes the following steps: construction of validation hierarchy, design of validation experiments, UQ in computations, and validation metrics. The fundamental idea of this framework is phenomena decomposition, which is similar to the "divide-and-conquer" approach for the closure development in MCFD solver. It decomposes a complex system into several progressively simpler tiers. Each tier represents a series of sub-systems or phenomena of the complete system; an example is given in Figure 1. A new type of experiment termed validation experiment needs to be conducted according to the decomposition which should provide detailed measurements for all the inputs and outputs of each component, including comprehensive uncertainty analysis of these measurements. The most updated criteria of validation experiment are evaluated in [12].

This framework provides a detailed guidance with solid theoretical background for the uncertainty study and validation of a computational model, thus can be applied to the MCFD solver. However, based on the author's best knowledge, there is no real application that follows the complete process of this framework. One of the major difficulty is that most of the currently available datasets are not suitable for this framework since those datasets were measured from traditional experiments, which cannot meet the standard of the validation experiment.

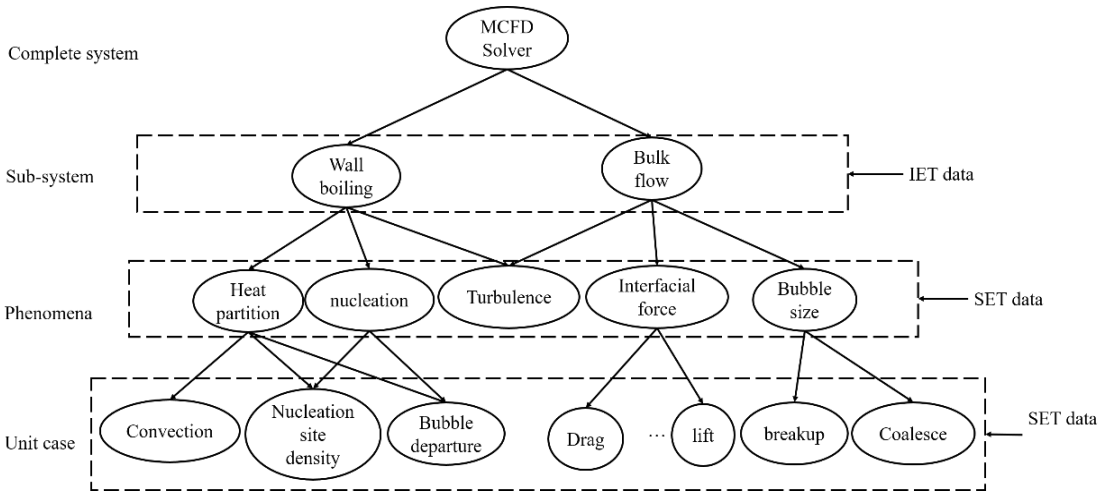

Figure 1. Validation hierarchy for MCFD solver based on AIAA guidance.

The second category is to develop new mechanistic closure relations that aim to better describe the underlying physics of the two-phase flow and boiling phenomena. With closure relation that resolves the physics, the uncertainty of it can be significantly reduced. Among the efforts of this category, developing new wall boiling closure relation that consider the detailed physical mechanism for wall boiling heat transfer is an active research topic. New mechanistic closure relations that focus on the bubble dynamics [13], or the bubble sliding effects [14, 15] have been developed. Those new closure relations were validated against a few experimental datasets and demonstrated better agreement, compared to closure relations that heavily rely on empirical correlations. Another active research topic is the closure relation for interfacial area transport which aims to describe the evolution of interfacial structure across different two-phase regimes. Since firstly proposed by Kocamustafaogullari and Ishii [16], various works on interfacial area transport equation (IATE) have been developed for different phenomena, such as wall nucleation [17] and bulk condensation [18], as well as for different geometries, such as round pipe [19] and rectangular channel [20].

The major limitation of this approach stems from a fact: the quantitative measurement of physical processes relevant to two-phase flow and boiling, such as nucleation and bubble deformation, are still very challenging. Thus a fully understanding of the underlying physics of these phenomena is still not possible. Inevitably, artificial concepts and empirical parameters are included in these mechanistic correlations, and those parameters are tuned, just like the efforts of the first category, to make the closure relation match the data. In this sense, the predictive capability of those mechanistic closure relations is heavily relied on, and thus limited by, the individual researcher's experience and knowledge. In many cases, those proposed closure relations demonstrate a good agreement on measurements under certain conditions, but fail to match measurements under other conditions [21]. The applicability range of those closure relations is thus limited.

To summarize, the significant uncertainty of the closure relations of the MCFD solver hampers the predictive capability of it, thus causing a disconnect between running simulations with MCFD solver to making practical decisions based on it. Current efforts to address the uncertainty issue, including the VUQ strategy to quantify and validate the

uncertainty of solver predictions and the development of mechanistic closure relations, both have their limitations. In this sense, novel insights are required to address this uncertainty issue.

1.2 Motivation

It worth noting that some recent developments in multiple research fields have demonstrated potential impact on the study of two-phase flow and boiling, and MCFD solver can benefit from those developments.

First, the development of experimental technology makes the high-resolution measurement for two-phase flow and boiling possible. The particle tracking velocimetry (PTV) has been used for measuring whole field distribution of QoIs of the two-phase flow and boiling system. In the representative work [22-24], promising results are demonstrated for the quantitative measurement of the detailed bubble dynamics and flow velocity fields. Moreover, high-speed infrared (IR) camera has been applied to measure the detailed boiling process. The pioneering work is the UCSB-BETA experiment [25, 26], which measured the detailed nucleation and the corresponding nucleate boiling heat transfer of pool boiling and thin liquid film boiling. Another representative work is [27], which couples the high-speed IR camera and high-speed camera to estimate the wall heat flux partitioning in nucleate boiling heat transfer. Information extracted from those high-resolution experiments can significantly increase the current available two-phase flow and boiling database, which is usually from measurements on limited points of the whole domain.

Second, the recent progress of computational power and numerical algorithms makes the first-principle simulation of the two-phase flow and boiling heat transfer system possible. In these simulations, the interface is fully resolved through interface capturing or interface tracking methods. These simulations provide high-fidelity results which can be treated as “virtual experimental data”. Moreover, many important phenomena which cannot be measured in experiments can be derived based on the simulation results. One of the representative works [28, 29] conducted the direct numerical simulation (DNS), which use the level set method to track the interface, for the adiabatic bubbly flow system. The obtained results were used to analyze the bubble-induced turbulence and the interfacial

forces. Other works [30, 31] simulated the high heat flux pool boiling, with interface tracking method (ITM) based on color function and large eddy simulation for turbulence. The details of nucleate boiling, including micro-layer dynamics, can be captured. From the simulation results, the heat flux partitioning and the ratio of vapor-to-liquid area over the heat transfer surface were calculated. Moreover, the feasibility of using high-fidelity computational model to quantify the uncertainty of low-fidelity model has been demonstrated within statistical framework [32]. Within this "high-to-low" framework, the high-fidelity simulation results can be used to quantify the uncertainty of the MCFD solver.

Third, based on the author's best knowledge, the value of both high-fidelity simulation and high-resolution experiment are still not fully exploited. One major reason is the traditional data analysis method used in the engineering community is not capable to handle the "big data" concept. It should also be noted that the data mining technique based on machine learning algorithms have already made progresses in many research fields which demonstrates it is a promising tool for the information extraction of the high resolution experimental measurements. Open source packages such as scikit-learn or commercial package Matlab can be conveniently used for such purposes.

Last, with the development of algorithms and computational power, the deep neural network (DNN) demonstrates its power and draws increasing attention with a series of successful applications in several topics [33]. Recently, there are wide applications of DNN on several areas including computer vision [34], natural language processing [35], and playing the sophisticated Go game [36]. For the physics problem, DNN also demonstrates its potential to predict high resolution QoIs with the local input features. Ling et al. [37] use DNNs with the input of invariant tensor basis to predict the anisotropy tensor in turbulent flow. The DNN is trained by single-phase turbulent flow DNS results. The trained DNN demonstrated significant improvement compared to baseline RANS eddy viscosity models. Inspired by this, Kutz [38] predicts the DNNs would play a significant role in turbulence modeling in next decade. In MCFD solver, a neural network based closure to represent the interfacial force is studied [39] that learn from DNS data. The promising results indicate DNN is a potential tool for the data-driven closure relation development.

However, according to the authors' best knowledge, there is still no application of DNN for boiling related problems.

To summarize, recent developments in experimental technology and computational power have created increasingly affluent data for two-phase flow and boiling heat transfer problem. Whereas the state-of-the-art statistical and machine learning methods can be applied to leverage these new data sources. Inspired by these developments, a novel data-driven framework to address the uncertainty issue of the MCFD solver can be developed.

1.3 Dissertation overview and outline

The significant uncertainty within the closure relations of the MFD solver hampers the predictive capability of it. The traditional two approaches to address this issue are: conducting model selection and parameter tuning based on expert judgement; developing new mechanistic closure relations that better describe the underlying physics of the relevant phenomena. However, these two approaches demonstrate only limited applicability on this issue.

In this dissertation, a new framework to address this uncertainty issue is proposed. Directly driven by data, this framework aims to address the uncertainty of MCFD solver also through two approaches. In contrast to the traditional efforts, these two approaches are based on and driven by data. The first approach is to perform a comprehensive VUQ on MCFD solver with available database. This VUQ work is based on Bayesian inference which learns from data. The second approach is to develop data-driven closure relations based on deep neural networks, which are benefited from the "big-data" and good mathematical properties of DNN and thus have lower uncertainty and better predictive capability compared to traditional empirical correlations. Such a data-driven framework is possible today due to technological advancements on following topics:

- High resolution experimental measurement for boiling process
- High fidelity simulation for boiling process with detailed local features
- State-of-the-art machine learning and statistical algorithms

This chapter has discussed the background, motivation, and achievements of the dissertation. The remainder of the dissertation is structured in the following manner.

Chapter 2 provides an overview of the data-driven framework, as well as the explanation of the essential concepts that closely relevant to this dissertation.

Chapter 3 discusses the data processing and storage procedure. The data processing examples for experimental measurements and simulation results are discussed respectively. The concepts of virtual containers, which is proposed for storing data for multipurpose usage, is introduced.

Chapter 4 and 5 introduces a data-driven VUQ procedure for MCFD solver. In Chapter 4, a detailed introduction of Eulerian-Eulerian two-fluid model based MCFD solver is provided, including the characterization of its closure relations. Then a six-step VUQ procedure for MCFD is introduced with technical details. In Chapter 5, two case studies on two-phase flow and boiling heat transfer is conducted as demonstrations of the proposed VUQ procedure.

Chapter 6 introduces the data-driven modeling approach for boiling closure relation development. The fundamental ideas of deep learning are discussed in this chapter. An example using deep feedforward network is demonstrated.

Chapter 7 gives a detailed summary of contributions, conclusions, and recommendation for future works.

CHAPTER 2. OVERVIEW OF THE PROPOSED FRAMEWORK

The chapter provides an overview of the data-driven analysis framework. The framework consists of three components. The first one is data processing and storage procedure, which aims to convert the heterogamous data from high-resolution experiments and high-fidelity simulation to well-structured datasets. With the processed data, two types of application for the MCFD solver can be developed, one is the validation and uncertainty quantification (VUQ); the other is the data-driven modeling. Before introducing the framework, several essential concepts that closely relevant to the dissertation is introduced in Section 2.1.

2.1 Essential concepts

The section discusses the essential concepts that closely related to the main theme of this dissertation.

2.1.1 Verification

The definition of verification in the context of modeling and simulation is given by [40]:

Verification is the process of determining that a model implementation accurately represents the developer's conceptual description of the model and the solution to the model.

The verification can be further divided into two types: code verification and solution verification. The code verification is *the process of determining that the numerical algorithms are correctly implemented in the computer code and of identifying errors in the software* [40]. The code verification aims to identify and correct potential errors in the source code and numerical algorithms of the model. Two methods are widely used for the code verification: the method of exact solutions (MES) and the method of manufactured solutions (MMS).

The solution verification is *the process of determining the correctness of the input data, the numerical accuracy of the solution obtained, and the correctness of the output data for a particular simulation* [40]. The solution verification mainly deals with two types of errors. One is Human errors, which could be introduced in the preparation of inputs

needed by the simulation, or in the post-processing of output results from the simulation. The other is the numerical error stems from the discretization and iteration process during the simulation. Unlike code verification, solution verification needs to be performed for every simulation if it is significantly different from previous verified solutions.

2.1.2 Validation

There are several definitions of validation that have been used in different communities, one of the most well-known definition is included in [40] which defines validation as:

The process of determining the degree to which a model is an accurate representation of the real world from the perspective of the intended uses of the model.

Essentially speaking, verification is a mathematics issue while validation is a physics issue. In other words, the verification deals with the problem: is the model solves the equation correctly? Whereas the validation deals with the problem: is the model solves the correct equation?

In previous practices, a widely used validation approach is to graphically compare the model predictions with the experimental measurement. Such "graphical comparison", while provides a basic understanding of the model accuracy, cannot generate a quantitative measurement of the simulation-data agreement, and can hardly lead to a reasonable evaluation of the solver. To address this issue, validation metrics that aim to provide a quantitative measurement of the agreement between model predictions and experimental measurement are proposed [41-43]. In this dissertation, two validation metrics are applied to the VUQ of MCFD solver.

2.1.3 Sources of uncertainty

For a general computational model, there are three sources of uncertainty:

- *Model parameter uncertainty*. A computational model inevitably contains parameters that need to be specified before the model can be used for prediction. Those parameters, whether denoting certain input physical quantities such as inlet velocity or wall heat flux, or representing the empirical description of a closure relation, have uncertainties that influence the prediction. Such uncertainty can

come from the intrinsic variation of the physical process such as the fluctuation of inlet velocity, or the lack of knowledge about a certain phenomenon such as the empirical parameter in a closure relation. In this sense, those parameters are treated as random variables if the UQ is performed under the Bayesian framework. The uncertainty introduced by those parameters are termed as model parameter uncertainties, which needs to be quantified and then propagated through the model.

- *Model form uncertainty*. The model form uncertainty is also termed as model bias, model inadequacy, or model discrepancy in different references. It stems from the simple fact that no model is perfect. This occurs even for a model with no parameter uncertainty so that the true values of all parameters required for a model are known. With all those true parameter values, the obtained QoIs from the model still would not be their true values in the real world. Such discrepancy is embedded in the formulation of the model, which usually includes approximations and simplifications for certain complicated physical process, as well as ignorance of some physical interactions between different phenomena, especially for complicated multi-scale problems such as the multiphase flow and boiling. The model form uncertainty is generally problem dependent and more difficult to address as compared to the parameter uncertainty. The study of the model form uncertainty is a topic of active research.

- *Numerical errors and uncertainty*. The numerical errors mainly arise from the discretization process and map the continuum PDE to discrete equations, insufficient iterative convergence for solving the nonlinear equations, as well as the round-off of simulation results. Strictly speaking, the evaluation of numerical errors is not considered a work of validation, but a work of solution verification as discussed before.

2.1.4 Uncertainty quantification

The uncertainty quantification (UQ) process can be characterized as two different types: the forward UQ and the inverse UQ. The first type is based on the assumption that the model parameter uncertainties are already known. Thus the probability distributions of

the QoIs can be obtained by simply perturbing the parameter values according to their known distributions. This can be done with the Monte Carlo method with certain sampling strategies such as Latin Hypercube Sampling (LHS). In most practices of the forward UQ, the experimental data is not directly evolved. This process is usually applied to problems that only evolve measurable parameters with clear knowledge. The inverse UQ, on the other hand, is based on a more realistic assumption that we have limited knowledge of the parameters implemented in the model. Thus, the uncertainties of the parameters need to be inferred using experimental measurements. The Bayesian framework is a suitable statistical tool for such inference and has multiple applications since it was first introduced for computational models by [44]. The Bayesian framework assumes that parameters can be regarded as random variables, and have prior distributions based on current knowledge about it. With the experimental data, the likelihood function can be calculated. Combing this likelihood function with the prior distribution, the posterior distribution can be obtained. The likelihood term takes into account how probable the data is given the parameters of the model. Once the posterior distribution of the parameter is obtained, it can be propagated through the model to construct uncertainties of QoIs using forward UQ. In the inverse UQ, the model form uncertainty can be considered. For problems with empirical parameters that cannot be directly measured, inverse UQ needs to be applied.

2.1.5 Sensitivity analysis

The general objective of sensitivity analysis (SA) is to quantify the individual parameter's contribution towards the QoIs and determine how variations in parameters affect the QoIs. A solid SA should be able to provide a ranking of the parameters by their importance to the QoIs. With the uncertainty into consideration, SA should also be able to identify how the uncertainty of model predicted QoIs can be apportioned to the different sources of uncertainty of the model inputs.

2.1.6 Predictive capability

The most important role of modeling and simulation is to make predictions regarding untested conditions. In engineering, the ability of a model for prediction is termed predictive capability. In contrast to the model prediction in scientific study, which focus on finding the philosophical concept of "truth", the predictive capability concerns whether

the predictions made by the model is used to inform and improve the decision-making process. In other words, the predictive capability evaluates the adequacy of predictions made by a model to meet the accuracy requirement for QoIs. The predictive capability maturity method (PCMM) proposed by Oberkampf [40] provides a comprehensive guidance to evaluate the predictive capability of a model.

2.1.7 Data-driven modeling

Modeling a complex physical process that has not yet been fully understood is a difficult task. The classical approach is to formulate a model with some simplification assumptions. The model formulated in this way is consistent with the simplified physical process, but some parameters are left unknown until being empirically determined with the support of available data. While data also plays a role in this modeling approach, it is still mainly driven by the knowledge and experience of the researcher.

The data-driven modeling approach, on the other hand, is focused on the data about the physical process, aiming to find the connection between the input conditions of the process and the output QoIs of it. This approach does not require the explicit understanding of the physical process. The data-driven modeling is possible today due to 1). the significantly increased data availability from high-resolution experiments and high-fidelity simulations; 2). the recent breakthrough in machine learning especially the deep learning algorithm[33].

2.2 Overview of the proposed framework

The data-driven analysis framework for boiling problem proposed in this dissertation can be simply illustrated in Figure 2. It aims to quantify and validate the uncertainty of MCFD solver through a VUQ procedure, and reduce the uncertainty of MCFD solver through data-driven modeling of closure relation. There are three major components of the framework.

The first component is the data processing and storage. This dissertation mainly relies on data from high-resolution experiments and high-fidelity simulations. Most of these data are heterogeneous and cannot be directly applied to the analysis of MCFD solver. Thus a data processing procedure is needed to convert the heterogeneous data to the form that is

compatible with the MCFD solver. Moreover, considering the fact that one dataset can serve multiple purposes, the processed dataset should be properly stored to maximize the flexibility for multipurpose usage. This part of work will be discussed in Chapter.3.

The second component is VUQ of the MCFD solver, with the focus on the closure relations. The closure relations contribute significant uncertainties to the MCFD. The VUQ component of the framework uses the modular Bayesian approach to quantify both the parameter uncertainty and the model form uncertainty of the closure relations in MCFD solver, then propagate the obtained uncertainty through the solver to obtain the uncertainty of QoIs predicted by the solver. The obtained QoIs, with its full uncertainty distribution, are quantitatively validated against all available datasets. This part of work will be discussed in Chapter.4.

The third component is data-driven modeling for the closure relations of the MCFD solver. In contrast to the modeling approach based on expert's knowledge and experience, the data-driven modeling approach discussed in this dissertation focuses on the data of the boiling process, using the deep learning algorithm to identify the connection between the input flow and surface features of boiling process and the output QoIs of it. This part of work will be discussed in Chapter.5.

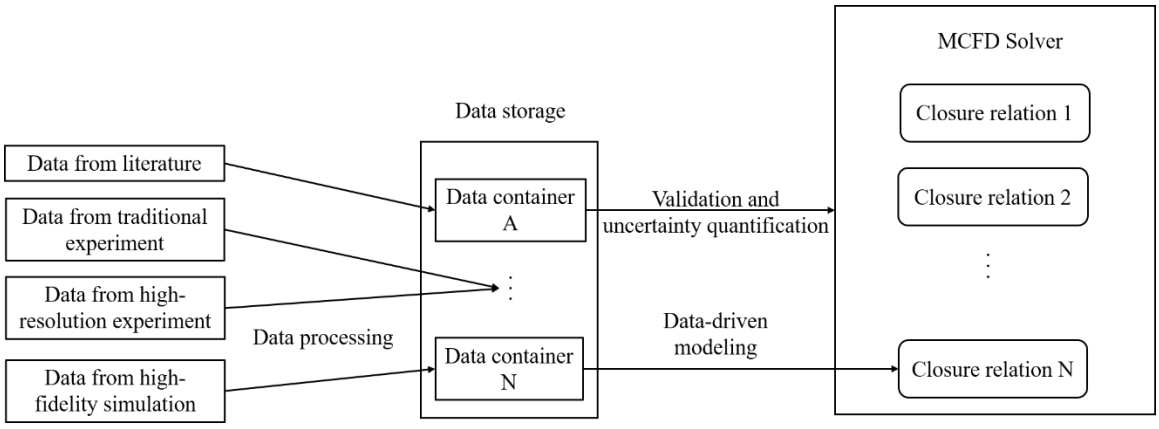

Figure 2. Overview of the data-driven analysis framework.

In this dissertation, two examples for data processing and storage are demonstrated. Based on the obtained data, two applications, one for VUQ and one for data-driven modeling, are developed as a demonstration of the last two components of the framework. It should be noted that further applications can be developed that couples these two

components together. For example, a researcher can first develop a data-driven closure relation, then incorporate this closure relation in the MCFD solver and perform VUQ with it.

CHAPTER 3. DATA PROCESSING AND STORAGE

This chapter discusses the data processing and storage procedure of the data-driven analysis framework. The nucleate boiling is a complex multiphysics process that involves interactions between heating surface, liquid, and vapor. Thus the data related to boiling also have complex structures. Moreover, different applications could be developed based on one dataset, whereas one application could also depend on several interrelated datasets. To maximize the convenience of data usage, the concept of "virtual container" is adopted for the storage of the obtained data.

In this chapter, the data processing procedure is applied to both high-resolution experiments and high-fidelity simulations. Multiple QoIs are extracted and stored in the virtual container. The purpose of this procedure is to convert the heterogeneous rich data to well organized datasets that can be conveniently used for quantifying or reducing the uncertainty of MCFD solver through various applications.

3.1 Processing boiling images from IR camera

The data measured by high-speed infrared (IR) camera in both pool boiling and subcooled flow boiling experiments are studied in this work. A typical IR experiment is demonstrated in Figure 3. These experiments used nano-meter-thick metal film deposited on a glass substrate for ohmic heating. This design ensures uniform heat flux distribution on heating surface. The transparency of glass ensures it is the heating surface's infrared that is captured by the camera. For subcooled flow boiling, the setup is similar with a different direction. The data from both the pool boiling experiments (UCSB-BETA) [25, 26] and subcooled flow boiling experiments (from MIT boiling experimental facility) [45] are processed in this work.

The raw data from IR camera is the counts of photons the sensor received. It can be easily converted to temperature data. However, the temperature contains only static information of the boiling process, it is very difficult to identify active nucleation sites and the corresponding nucleation boiling information from the temperature. A more desired data type is the heat flux distribution over the heating surface from which the nucleation information and the heat partitioning can be obtained. The transient heat conduction

equation over the substrate is solved to obtain the heat flux distribution over the heating surface. With the heat flux obtained, the nucleation information including the location of active nucleation sites, the heat partitioning, the bubble area fraction, etc. can be obtained. This is done through a parallel processing system as depicted in Figure 4. The detailed work would be discussed in Section 3.1.2.

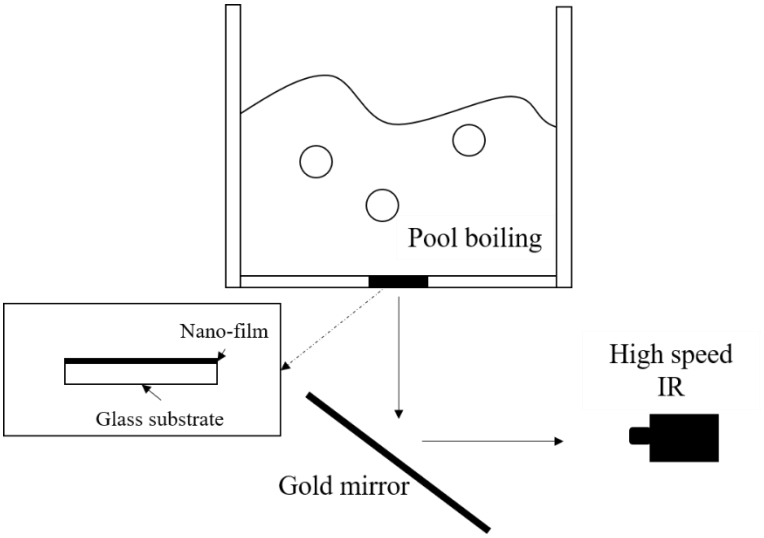

Figure 3. Schematic of measuring high-resolution boiling process with IR camera.

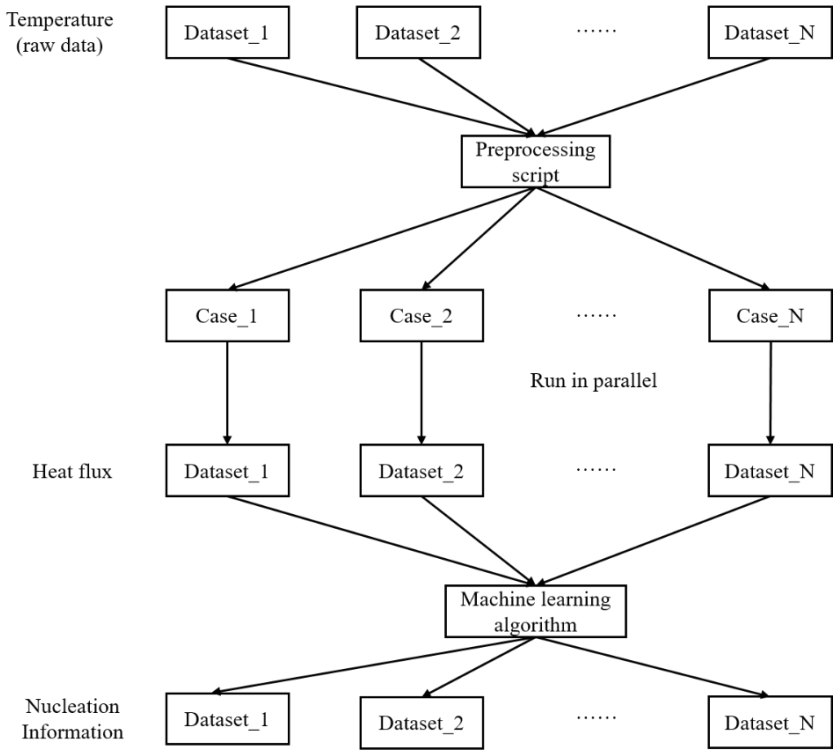

Figure 4. Automatic data processing for IR boiling images.

3.1.1 Heat flux distribution processing

For UCSB-BETA experiment, the heat flux is obtained by solving transient three-dimensional heat conduction equation for the glass substrate, as shown in Figure 5. A special type of boundary condition is developed to map the IR temperature data to the upper wall cells. The surrounding walls are assumed to be adiabatic, while the lower wall is assumed to be a mixed boundary type with constant heat transfer coefficient. Grid sensitivity is conducted to ensure the obtained results are mesh independent.

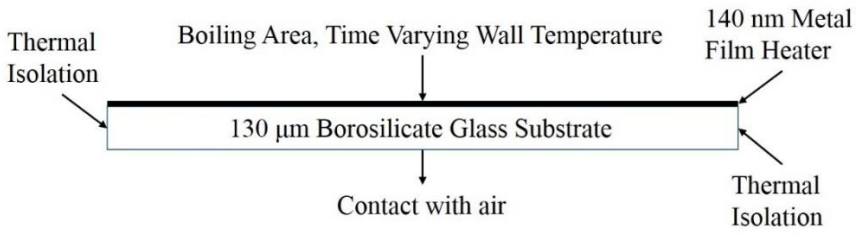

Figure 5. Scheme of computational domain for deriving heat flux distribution on heating surface.

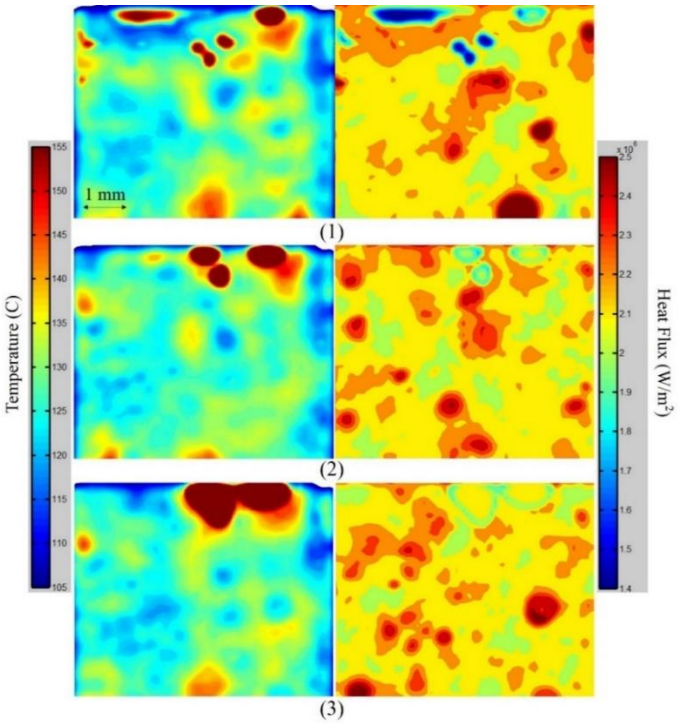

Figure 6. Three frames of (1) appearance of hot spot (t = 0ms); (2) Some hot spots being rewetted while others beginning merging (t = 60.3 ms); (3) Merged hot spot become irreversible and lead to burnout (t = 119.7 ms); $q = 2.15\ MW/m^2$, BETA experiment.

An example of obtained results are demonstrated in Figure 6. The development of the boiling crisis can be directly observed from these images.

Similar process is performed for the subcooled flow boiling experiments. Figure 7 demonstrates an example of the obtained temperature and heat flux distribution. As can be seen, the sliding effect can be clearly observed in the heat flux map.

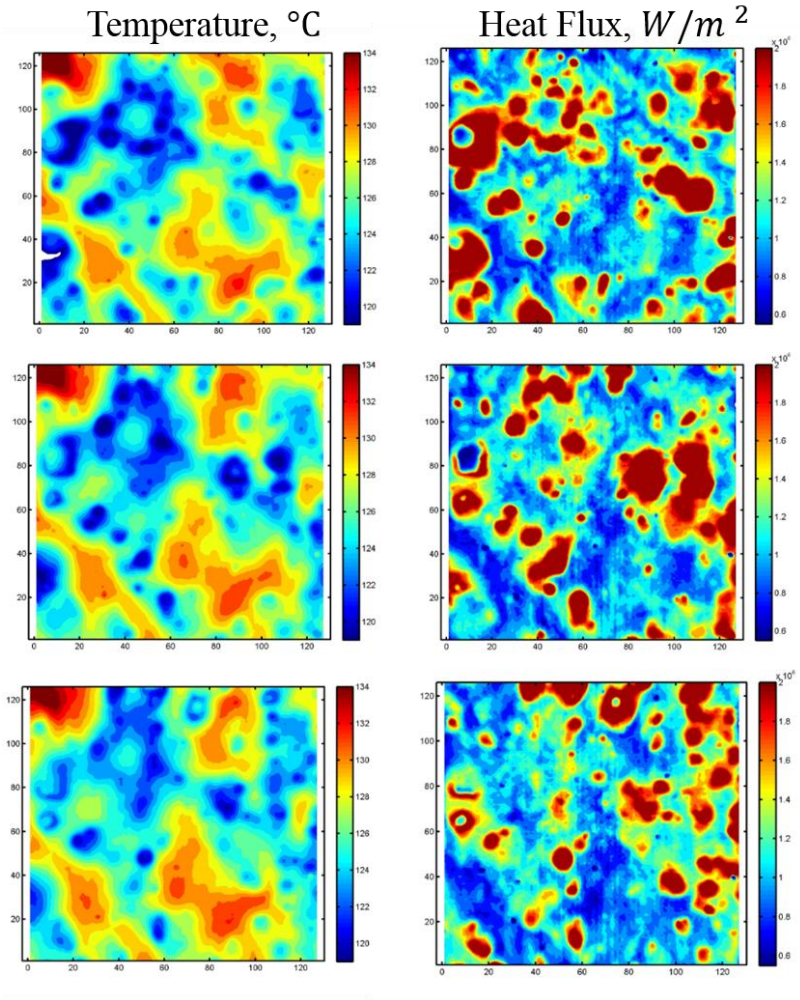

Figure 7. Temperature and corresponding heat flux distribution in flow boiling, mass flow rate = 500kg/m^2, heat flux = 1.4MW/m^2.

3.1.2 Nucleation information processing

As can be seen from Figure 6 and Figure 7, the nucleation sites can be clearly identified from the heat flux images. However, manually counting and recording the nucleation information would be very time consuming and tedious, the human error could also be introduced. In this work, the hierarchical clustering, an unsupervised machine

learning algorithm is applied to automatically identify and record the nucleation information.

The hierarchical clustering algorithm is applied to the obtained heat flux images to identify the active nucleation sites. There are two types of hierarchical clustering, the agglomerative and divisive. In this work, the agglomerative type is adopted, which is a "bottom up" approach. Each sample starts in its own cluster, in each iteration, pairs of clusters are merged until the stop criterion is satisfied.

The clustering process is dependent on measuring the dissimilarity, which is usually represented by distance, between sets of samples. Such measurement is specified by a chosen metric and the corresponding linkage criterion. The metric determines how to measure the dissimilarity, while the linkage criterion determines based on which property of the cluster to measure the dissimilarity. The commonly used metrics for hieratical clustering includes: Euclidean distance, Maximum distance, Mahalanobis distance, etc.

In this work, the Euclidean distance is chosen as the metric,

$$\|\boldsymbol{a}-\boldsymbol{b}\|_2=\sqrt{\sum_i (a_i-b_i)^2}\,, \tag{1}$$

where i is the dimension of the sample. In this work, the clustering is applied to pixels of an image, thus $i=2$.

A cluster consists of multiple samples; thus the metric alone cannot generate a unique distance between two clusters. The linkage criterion is introduced to resolve this issue and provide a unique measurement for the distance between two clusters. The maximum linkage criterion, for example, defines the distance between two clusters A and B as the maximum pairwise distance of samples inside A and B. The minimum linkage criterion, in contrast, defines the distance as the minimum pairwise distance of samples inside A and B.

In this work, the centroid linkage is chosen, which means the Euclidean distance between two clusters A and B is calculated by

$$d(A,B)=\|c_A-c_B\|_2\,, \tag{2}$$

where c_A and c_B are the centroids of clusters A and B, respectively.

Before running the clustering process, a value is specified to the linkage criterion, the merging process will stop when the distance between every pair of clusters exceed this value. This value needs to be adjusted for each case, since a too large value would falsely merge two nucleation sites into one, while a too small value would falsely divide one nucleation sites into several. In the practice, around 50 frames are randomly chosen from each case to manually examine if the clustering is correct or not, based on which the linkage value is adjusted.

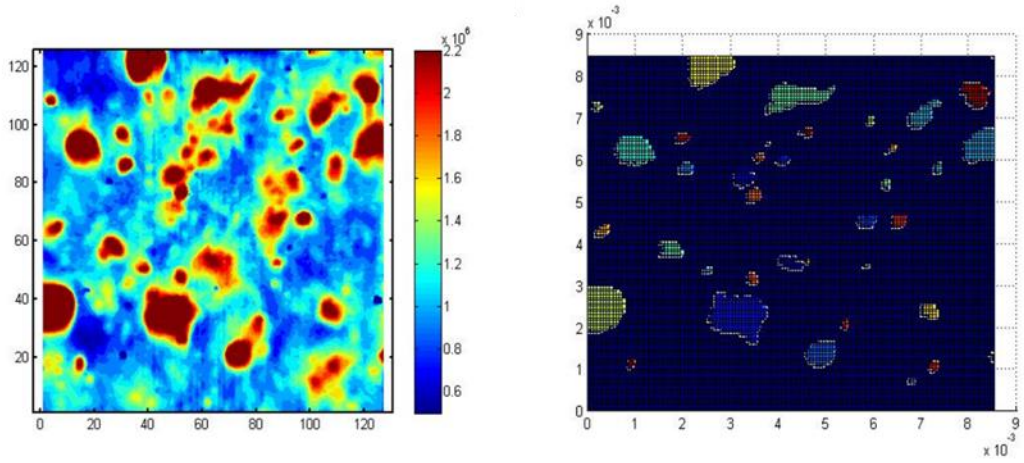

Figure 8. Example of hierarchical clustering for active nucleation sites identification.

An example of the hierarchical clustering for active nucleation sites identification is demonstrated in Figure 8, it can be found that even the very small incipient nucleation sites can be identified.

Once the nucleation sites are identified, the corresponding nucleation information can be obtained, including the nucleation site density, bubble area fraction, and evaporation heat flux, etc., one example is provided in Figure 9. These information are stored for future usage.

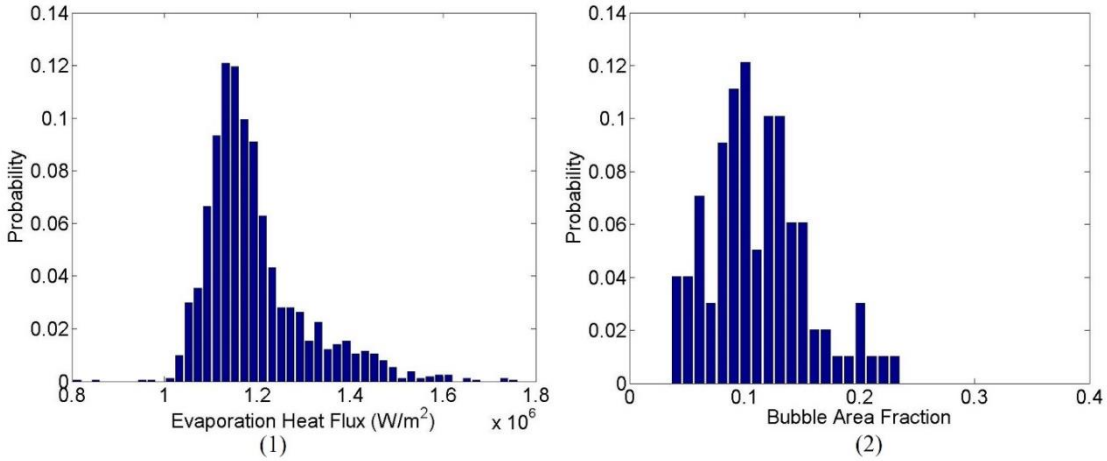

Figure 9. Evaporation Heat Flux and Bubble Area Fraction Distributions, $q = 1.16\ \mathrm{MW/m^2}$, BETA experiment, Heater Ti36A.

These obtained boiling data can serve multiple purpose. First, it provides direct observation to the detailed boiling process that provide better understanding of the flow boiling process and the boiling crisis. This could serve a foundation for new mechanistic closure relation development. Second, the obtained data can be further averaged over time and space to serve for the VUQ of wall boiling closure relation in MCFD solver. Last, coupling with the flow dynamics measurement, the data can be used for data-driven modeling. In this dissertation, only the second application is developed, but it should be noted that more applications can be development with these high-resolution data.

3.2 Processing high-fidelity simulation results

The high-fidelity simulation results are processed to serve for data-driven modeling purpose, as will be discussed in Chapter.5. The simulation studied multiple pool boiling scenarios with interface tracking method (ITM) [46]. The simulation is based on directly solving the incompressible Navier-Stokes equations with a sharp-interface, phase-change model proposed in [30]. The three conservative equations solved in this approach can be expressed as follows:

Mass:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \rho \boldsymbol{U} = 0\,, \tag{3}$$

Momentum:

$$\frac{\partial \rho \boldsymbol{U}}{\partial t} + \nabla \cdot (\rho \boldsymbol{U}\boldsymbol{U}) = -\nabla p + \nabla \cdot \{\mu(\nabla \boldsymbol{U} + (\nabla \boldsymbol{U})^{\boldsymbol{T}})\} + \boldsymbol{f}\,, \tag{4}$$

Energy:

$$\mathrm{C}_p \left(\frac{\partial T}{\partial t} + \boldsymbol{U} \cdot \nabla T\right) = \nabla \cdot (\lambda \nabla T) + Q\,, \tag{5}$$

In addition to the conservative equations, the color function ϕ is used to track the interface between vapor and liquid:

$$\frac{\partial \phi}{\partial t} + \nabla \cdot (\phi \boldsymbol{U}) = -\frac{1}{\rho_{\boldsymbol{l}}} \Gamma_{\boldsymbol{lv}}\,, \tag{6}$$

In the simulation, the nucleation site is prescribed in the whole heating surface, and the heat conduction in the solid wall is considered through conjugate heat transfer. With the boundary conditions specified, the solver is able to predict the detailed boiling process with high accuracy. The QoIs of boiling process include the wall superheat, evaporation heat transfer component, convective heat transfer component towards liquid, and near wall bubble concentration. Such QoIs are the outcome of complex interactions between different phenomena, including: convection, evaporation, conjugate heat transfer, buoyancy, and nucleation. Although it is impossible to develop an explicit correlation or a model to accurately account for such interaction and give a reasonable prediction for the QoIs, a DNN that takes those phenomena as input features can serve as a "black-box" model for the prediction of boiling QoIs and can be applied to untested conditions.

On the other hand, however, the ITM simulation is performed on very fine meshes, the results contain detailed interface information, as well as the fluctuation of physical quantities. For the two-fluid-model, such information is unnecessary. In this sense, the obtained ITM results need to go through certain average process before being used for training the DNN which is supposed to be compatible with two-fluid-model.

In this work, a combination of time and space average is processed for each physical quantities $f(\boldsymbol{x}, t)$ into a space and time averaged form $\langle f \rangle(\boldsymbol{x}, t)$ which can be described as follows:

$$\langle f \rangle(\boldsymbol{x}, t) = \frac{1}{\tau}\frac{1}{l^3}\int_{t-\tau}^{t}\int_{x_1-l/2}^{x_1+l/2}\int_{x_2-l/2}^{x_2+l/2}\int_{x_3-l/2}^{x_3+l/2} f(\boldsymbol{x}', t')\,dx_3'dx_2'dx_1'dt'\,, \tag{7}$$

where τ and l are the averaging time scale and averaging length scale, respectively. One interesting fact that worth noting ,as discussed in [47], is that this average process is mathematically equivalent to the convolution operation over a 4-dimensional matrix (three dimension in space, one dimension in time), with a kernel function $g(\boldsymbol{x})$:

$$\langle f \rangle(\boldsymbol{x}, t) = \int \int_{R^4} \int \int g(\boldsymbol{x} - \boldsymbol{x}')\ f(\boldsymbol{x}', t) d\boldsymbol{x}' \,. \tag{8}$$

For quantities that are only valid in the heating surface, i.e. the potential nucleation site density, nucleation activation temperature, wall superheat, and the heat transfer components, the average process is performed on 2-dimensional surface and time. Such operation is widely adapted in the DNN for extracting and preserving features of the data. It is assumed the averaged process will preserve the causal-relationship between input features and the boiling QoIs. Such assumption is reasonable if the ITM simulation already reached quasi-steady state before the data is extracted. Every physical quantity obtained from the ITM simulation are propagated through the process to generate an averaged version of it. The void fraction α is obtained by propagating color function ϕ through this process. One example of the average process is demonstrated in Figure 10, where the color function describing the bubble interface is averaged over time and space to generate the void fraction distribution over a slice plane.

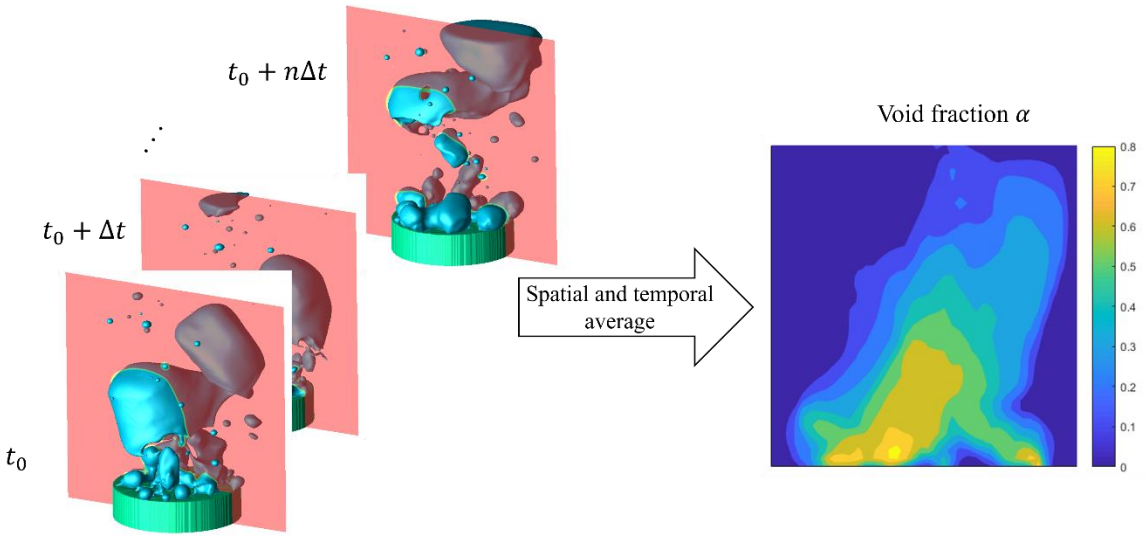

Figure 10. Demonstration of the average process, from bubble interface to void fraction.

After this average process, the raw data of hundreds gigabyte level would be reduced to hundreds megabyte level while still preserves all the important information.

3.3 "Virtual container" for data storage

The processed data should be properly stored for the future usage. There are different purposes to use these datasets, one researcher hope to investigate the relevant physical process from the data, another researcher wants to use the data to validate a model, while another researcher would like to develop a new model based on the data. In this sense, the data need to be stored in a flexible way to maximize the convenience for all purposes. Moreover, potential connections could exist between different datasets, thus the data storage should also be flexible to preserve such possible connection. Based on this, the concept of "virtual container" proposed in the Nuclear Energy Knowledgebase for Advanced Modeling and Simulation (NE-KAMS) [48] is adopted in this dissertation.

With this concept, datasets are stored in virtual containers according to the facility and the experimental condition. That is, no matter how many measurements was taken in one experiment, how many QoIs are measured and what their types are, it should be stored together in one container. This container should have a clear description about the information it stored and should provide access to all types of data it stored so other researchers that is not familiar with this experiment can still understand and use them. An example of virtual container storing the subcooled flow boiling data is given in Table 1. In this dissertation, the virtual container is stored in the *dataframe* format supported by Pandas, which is a python package.

The virtual container designed in this way ensures the data are well organized and easy to use. Moreover, the collaboration between two containers can also be achieved through this configuration. One possible scenario is illustrated in Figure 11. In this scenario, the interactions between two containers, which stored data of the same physical process, are characterized. First, the type I data from both containers are used for studying the physical process. Second, type II data of container A and type III data of container B are used together for data-driven modeling. Last, type III data of container A and type II data of container B are used for VUQ of the model.

Table 1. Example of an experimental data container

General information	
Source	MIT boiling experimental facility [45]
System configuration	
Geometry	Vertical flow in rectangular channel
Fluid materials	water liquid/vapor
Heater materials	ITO sapphire heater with synthetic CRUD
Test program	
Flow conditions	500 kg/m^2
Heat configurations	2um thick CRUD with 10um diameter chimneys on a 45um pitch
Heat flux	1400 kW/m^2
Data stored	
[D0] raw data	IR counts distribution
[D1] data type I	temperature/heat flux distribution
[D2] data type II	Nucleation information
[D3] data type III	Averaged heater temperature and heat transfer partition
Data characteristics	
Applicability	boiling model VUQ for flow boiling on low pressure
Quality	Good

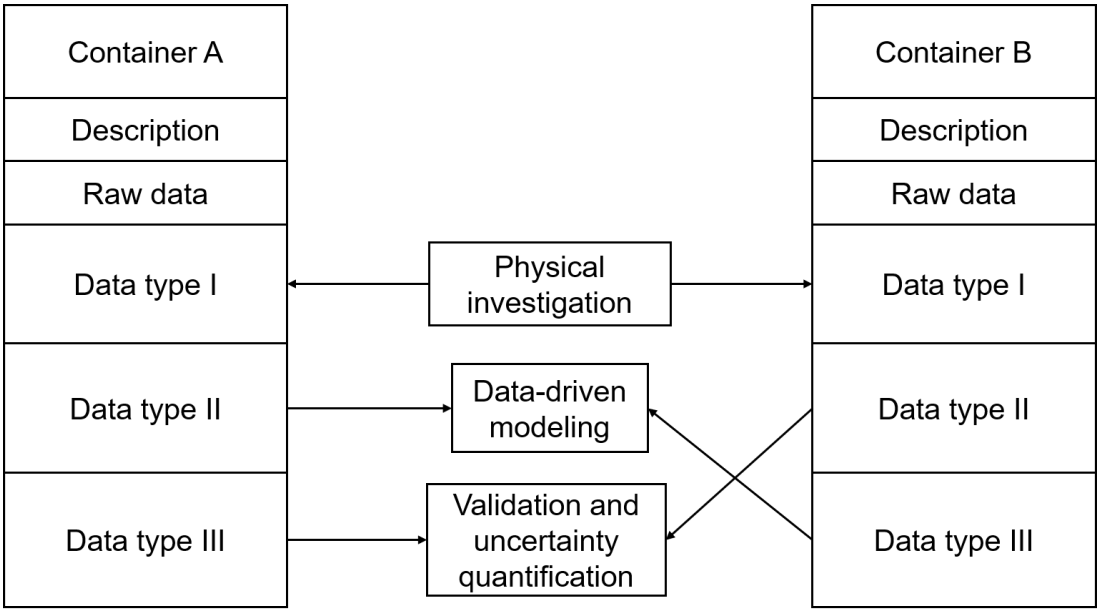

Figure 11. Collaboration between two virtual containers.

3.4 Summary remarks

In this chapter, the data processing and storage procedure is introduced with two examples. The purpose of this procedure is to convert the heterogeneous rich data to well organized datasets that can be conveniently used for quantifying or reducing the uncertainty of MCFD solver through various applications.

The hierarchical clustering algorithm is applied for the high-resolution IR boiling experiments. Active nucleation sites and the corresponding nucleation information can be automatically identified with the algorithm. Boiling related QoIs for MCFD solver are extracted. Similarly, the time and space average process is applied to high-fidelity ITM simulations. The extracted data are organized in the virtual container for further usage.

In the following two chapters, two different applications are developed driven by different datasets from traditional experiments, high-resolution experiments, and high-fidelity simulations. Each of the application reflects one usage of the data: from VUQ to data-driven modeling. It also should be noted that these two applications can be connected together: develop a data-driven closure relation first, then put the developed closure relation into the VUQ process. Data stored in virtual container can serve for such purpose with maximum flexibility.

CHAPTER 4. METHODOLOGY DEVELOPMENT FOR THE VALIDATION AND UNCERTAINTY QUANTIFICATION FOR MCFD SOLVER

In this chapter, a validation and uncertainty quantification (VUQ) procedure for the Eulerian-Eulerian two-fluid-model based multiphase-computational fluid dynamics solver (MCFD) is developed. The procedure aims to answer the question: how to evaluate if a MCFD solver adequately represents the underlying physics of a multiphase system of interest? The proposed procedure is based on total data-model integration (TDMI) approach that uses Bayesian method to inversely quantify the uncertainty of the solver predictions with the support of multiple experimental datasets. The framework consists of six steps with state-of-the-art statistical methods, including: 1). Solver evaluation and data collection; 2). Surrogate model construction; 3). Sensitivity Analysis; 4). Parameter selection; 5). Uncertainty quantification with Bayesian inference; and 6). Validation metrics calculation. Those steps are formulated in a modular manner and using non-intrusive methods. Such features ensure the applicability of the flexible procedure to different scenarios and modeling of multiphase flow and boiling heat transfer, as well as the extensibility of the procedure to support VUQ of different MCFD solvers.

4.1 Eulerian-Eulerian two-fluid model based MCFD solver

The fundamental idea of a two-fluid-model is to average the local instantaneous conservation equations, thus eliminating the need for tracking interfaces to achieve computational efficiency. The system of averaged conservation equations needs to be solved numerically, commonly using a finite-volume or finite-element method. The convergence and accuracy of the solution depend on numerical techniques and temporal and spatial resolutions needed to capture the dynamics and scales of governing physical processes.

4.1.1 Conservative equations

Generally speaking, the two-fluid model solver relies on solving three ensemble averaged conservative equations for mass, momentum and energy. The k-phasic mass conservation equation can be written as

$$\frac{\partial(\alpha_k \rho_k)}{\partial t} + \nabla \cdot (\alpha_k \rho_k \mathbf{U}_k) = \Gamma_{ki} - \Gamma_{ik} \,, \tag{9}$$

where the two terms on the left-hand side represent the rate of change and convection, the two terms on the right-hand side represent the rate of mass exchanges between phases due to condensation and evaporation.

The k-phasic momentum equation is given by

$$\frac{\partial(\alpha_k \rho_k \mathbf{U}_k)}{\partial t} + \nabla \cdot (\alpha_k \rho_k \mathbf{U}_k \mathbf{U}_k) = -\alpha_k \nabla p_k + \nabla \cdot [\alpha_k (\tau_k + \tau_k^t)] + \alpha_k \rho_k \mathbf{g} + \Gamma_{ki} \mathbf{U}_i - \Gamma_{ki} \mathbf{U}_k + \mathbf{M}_{ki}, \tag{10}$$

where *i* represents the interphase between two phases, $\mathbf{M}_{ki}$ represents the term of averaged interfacial momentum exchange, which can be modeled by a set of interfacial force closure relations.

The k-phasic energy conservation equation in terms of specific enthalpy can be given as

$$\frac{\partial(\alpha_k \rho_k h_k)}{\partial t} + \nabla \cdot (\alpha_k \rho_k h_k \mathbf{U}_k) = \nabla \cdot \left[\alpha_k \left(\lambda_k \nabla T_k - \frac{\mu_k}{\mathrm{Pr}_k^t} \nabla h_k\right)\right] + \alpha_k \frac{Dp}{Dt} + \Gamma_{ki} h_i - \Gamma_{ik} h_k + q_k \,, \tag{11}$$

where the terms on the right-hand side represent heat transfer in phase *k*, work done by pressure, enthalpy change due to evaporation and condensation, and heat flux from the wall. The wall boiling heat transfer is modeled by a set of closure relations.

4.1.2 Characterization of closure relations in MCFD

Solving a typical two-phase flow and boiling problem involves predicting the boiling process on the heated wall and flow and heat transfer process in the bulk flow. The boiling process involves complex multi-physics process that includes interaction between liquid and the wall surface, such as nucleation and bubble departure. Such process cannot be directly resolved in the MCFD solver. Thus, wall boiling closure relations are incorporated in the MCFD solver to predict the boiling process; some recent development also includes the capability for the prediction of the departure from nucleate boiling (DNB) [5, 15]. After nucleation and departure from the wall, the bubbles join the bulk flow and interacting with the liquid in the bulk flow. Resolving such interactions require a series of closure relations

for the interphase exchange of mass, momentum and energy. Interfacial force closure relations are proposed to describe the interphase momentum exchange, while interfacial condensation closure is necessary to describe the interphase mass and heat transfer for the subcooled flow boiling problem. Moreover, the size of bubbles has significant influence on those interphase exchanges, and closure relation is needed for determining the bubble size. Last, the turbulence can influence the interphase exchange, and the bubble dynamics in turn influences the turbulence, thus closure relation is also required to describe the bubble-induced turbulence. Thus the closure relations in a MCFD solver can be characterized into five categories: wall boiling, interfacial momentum exchange, interfacial mass/heat transfer, bubble size, and turbulence. There exist complex relationships between those closure relations and a typical structure of them is depicted in Figure 12.

The role of closure relation is the reflection of the "divide-and-conquer" philosophy that decompose a complex system into several sub-phenomena and models them with closure relations separately. It should also be noted that most of those closure relations are proposed for bubbly flow which assumes the continuous phase is liquid and the disperse phase is vapor or gas. Several investigations also assume such closure relations can be extended to droplet flow simulation where the continuous phase is vapor while disperse phase is liquid droplet. There are still gaps for the modeling of slug or churn flow using MCFD solver.

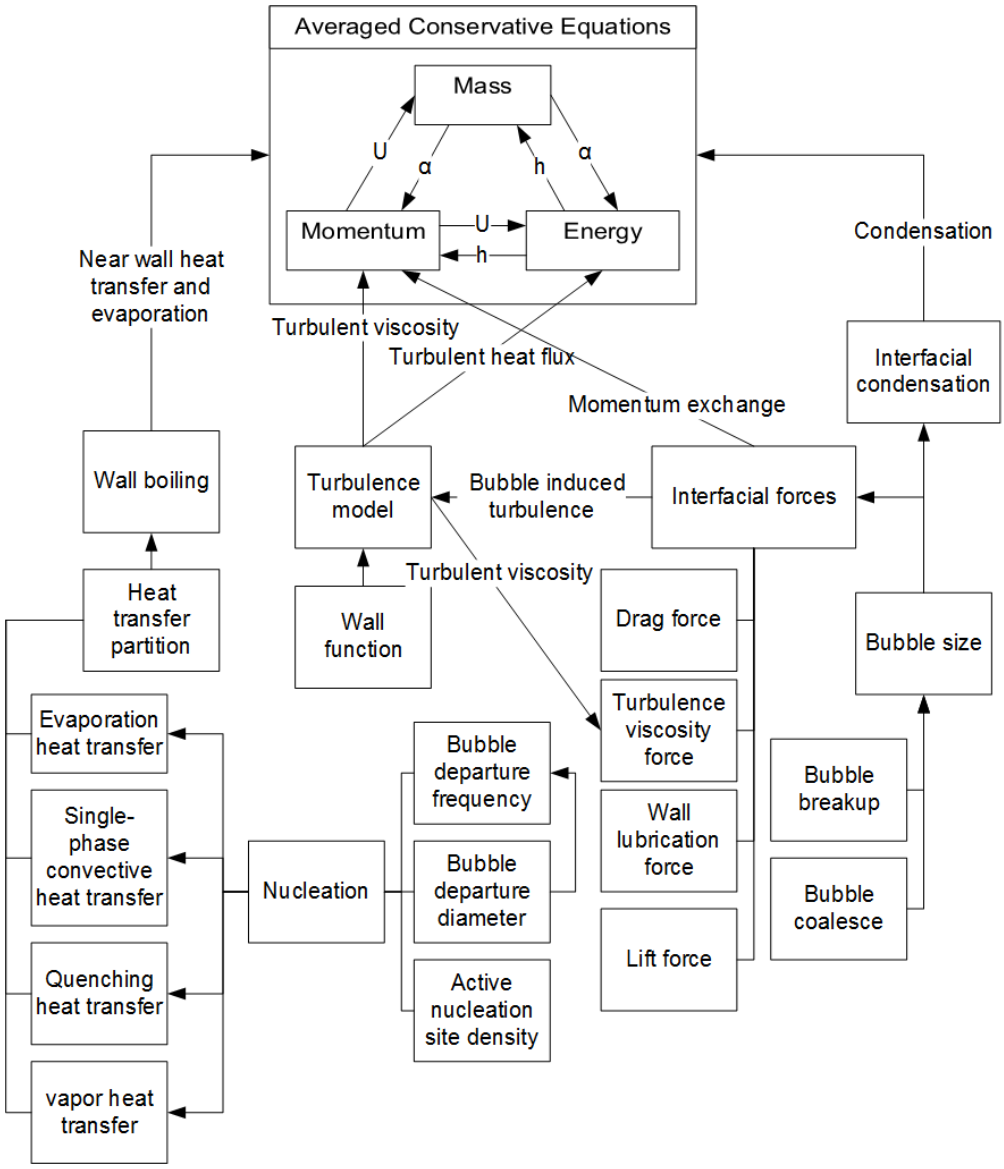

Figure 12. Closure relation structures in a typical MCFD solver.

Turbulence

The modeling of turbulence in MCFD solver is based on the Reynolds-averaged Navier–Stokes (RANS) model to obtain the turbulence viscosity ν^t. This type of turbulence model, which usually comes with wall functions, such as $k-\varepsilon$ model, $k-\omega$ model etc., already has been widely applied and tested in single phase flow problems. One additional closure in MCFD is to introduce a term that takes into account the bubble/droplet induced turbulence, such as the work by [49] and [28]. In most practices, only the turbulence of continuous phase - i.e. liquid phase in bubbly flow, vapor phase in droplet

flow -is modeled in this way, while the turbulent viscosity of dispersed phase is assumed to be linearly dependent on the v^t of continuous phase with a turbulence response coefficient C_t.

Interfacial momentum closure relations

The interfacial momentum exchange between two phases is represented by different types of interfacial forces. For a typical MCFD solver, five interfacial forces are modeled. The drag force is modeled to describe the resistance of relative motion between the two phases. The lift force is modeled to describe the force that exerts by continuous fluid flow past the bubble. The turbulent dispersion force is modeled to describe the effect of liquid turbulence on the bubble. The wall lubrication force is designed as an artificial force to move the bubble away from the wall to describe. The virtual mass force is modeled to describe the inertia of bubble acceleration or deceleration. Figure 13 illustrates these five interfacial forces.

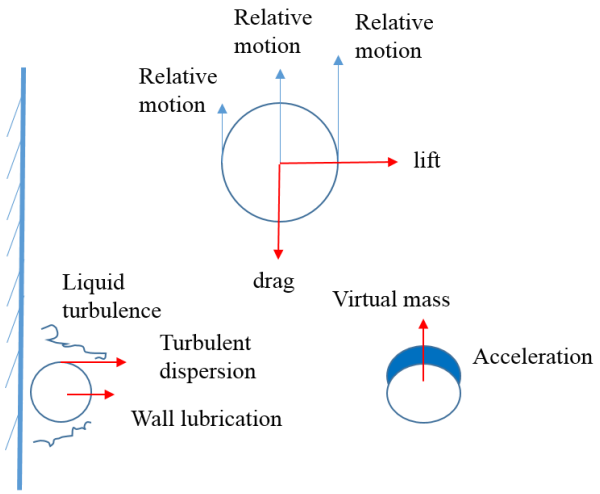

Figure 13. Schemes of interfacial forces.

Table 2 summarizes the expressions of those interfacial forces. Among these interfacial forces, the form of drag force and lift force can be analytically derived; thus their expression is quite consistent among different MCFD solvers. On the other hand, the force coefficients C_d and C_l are calculated from semi-empirical correlations. For drag force, several models can be used, such as the work done by [50] and [51], while for lift force, the correlation developed by [52] is widely used. It should also be noted that in some

practices, the force coefficients are also set to be constant for simplification purpose. The forms of other three forces is varied among different researchers due to the lack of solid theoretical support. Besides the expressions summarized in Table 2, there are other expressions can be used, such as the wall lubrication force model proposed by [53] turbulent dispersion force model proposed by [54].

Table 2. Expressions of interfacial forces

Force type	Expression
Drag force	$\mathbf{M}_g^D = -\frac{3}{4}\frac{C_d}{D_s}\rho_l\alpha\|\mathbf{U}_g - \mathbf{U}_l\|(\mathbf{U}_g - \mathbf{U}_l)$
Lift force	$\mathbf{M}_g^L = C_l\rho_l\alpha(\mathbf{U}_g - \mathbf{U}_l) \times (\nabla \times \mathbf{U}_g)$
Wall lubrication force [55]	$\mathbf{M}_g^{WL} = -f_{WL}(C_{wl}, y_w)\alpha\rho_l\frac{\|\mathbf{U}_r - (\mathbf{U}_r \cdot \mathbf{n}_w)\mathbf{n}_w\|^2}{D_s}\mathbf{n}_w$, $f_{WL}(C_{wl}, y_w) = \max\left(-0.2C_{wl} + (\frac{C_{wl}}{y_w})D_s, 0\right)$
Turbulent dispersion force [56]	$\mathbf{M}_g^{TD} = -\frac{3}{4}\frac{C_D}{D_s}\frac{v_l^t}{\sigma^t Pr_l^t}\rho_l\|\mathbf{U}_g - \mathbf{U}_l\|\nabla\alpha$
Virtual mass force [57]	$\mathbf{M}_g^{VM} = -C_{vm}\rho_l\alpha\left(\frac{D\mathbf{U}_g}{Dt} - \frac{D\mathbf{U}_l}{Dt}\right)$

Interfacial mass and heat transfer closure relations

Bubbles developed from nucleation depart from the wall and join the bulk flow. In subcooled flow boiling, the bubbles become surrounded by the subcooled liquid causing vapor condensation. The interfacial mass transfer related to condensation of vapor bubbles in the bulk coolant can be described as

$$\Gamma_{\mathrm{lg}} = \frac{h_{lg}(T_{sat} - T_l)A_a}{h_{fg}}, \tag{12}$$

where h_{lg} is the heat transfer coefficient between the two phases, which can be calculated using empirical or semi-empirical correlations, representative works have been done by [58] and [59].

Bubble size closure relations

The size of the bubble has significant influence on the interphase exchanges of mass, energy, and momentum. Initially, the bubble size is evaluated using empirical correlation derived from subcooled flow boiling. One example of the empirical correlation, as proposed by [60], is

$$D_s = \frac{D_{ref,1}(T_{sub} - T_{sub,2}) + D_{ref,2}(T_{sub,1} - T_{sub})}{T_{sub,1} - T_{sub,2}}, \quad (13)$$

where $T_{sub,1}, T_{sub,2}, D_{ref,1}, D_{ref,2}$ are empirical constants which have suggested values, but those values are often tuned in different applications. A more sophisticated development is to predict the bubble size distribution with the interfacial area transport equation [61]. The volumetric interfacial area concentration equation can be expressed as

$$\frac{\partial(A_a)}{\partial t} + \nabla \cdot (A_a \mathbf{U}_a) = \frac{2}{3}\frac{A_a}{\alpha}\left(\frac{\partial \alpha}{\partial t} + \nabla \cdot (\alpha \mathbf{U}_a)\right) + \Phi_{\mathrm{BB}} + \Phi_{\mathrm{BC}} + \Phi_{\mathrm{NUC}}, \quad (14)$$

in which the first term on the right-hand side refers to the contribution of phase change and expansion due to pressure-density change. Here Φ_{BB}, Φ_{BC}, and Φ_{NUC} represent the source and sink term induced by breakup, coalescence, and nucleation respectively. There are several semi-empirical correlations for those terms proposed by different researchers, reprehensive works include [62], and [63].

Another mechanistic approach for bubble size prediction is the multiple size group (MUSIG) model which deals with the non-uniform bubble size distribution by dividing the bubble size distribution into a finite number of groups. A more recent progress is the inhomogeneous MUSIG model by [64] which allows each bubble group to have its own velocity.

Wall boiling closure relations

Nucleation and growth of vapor bubbles serve as a mechanism for efficient cooling of the superheated fluid layer and hence heat removal from the heated wall. The wall boiling closure relation in MCFD solver is developed to provide a consistent treatment of phenomena that govern heat transfer in boiling. The general approach is termed “heat partitioning”, which decomposes the wall heat flux into several components representing

corresponding heat transfer mechanisms. The wall boiling model was first introduced by [65], which partitions the wall heat flux into three components: single phase forced convective heat transfer, quenching heat transfer, and evaporation heat transfer. This model is often called "Generation-I" model as many new refined models are developed based on it. It can be expressed as

$$q_{wall} = q_{Ev} + q_{Qu} + q_{Fc}\,. \tag{15}$$

The quenching heat transfer describes the heat transfer towards the liquid phase when the cool liquid replaces the detached bubbles from the wall. The quenching heat flux can be calculated with the expression proposed by [66]:

$$q_{Qu} = A_b \frac{2}{\sqrt{\pi}} f \sqrt{t_{wait} k_l \rho_l c_{p,l}} \left(T_{sup} - T_l\right), \tag{16}$$

where t_{wait} is the waiting time between the bubble departure and the appearance of a new bubble at a given nucleation site, which can be modeled by different empirical correlations.

The forced convective heat transfer happens in the area where no nucleation happens. The convective heat flux can be modeled as:

$$q_{Fc} = (1 - A_b) h_l \left(T_{sup} - T_l\right), \tag{17}$$

where$h_l$is the convective heat transfer coefficient which is usually modeled with semi-empirical correlations that take into account the near wall turbulence.

In the boiling process, a significant proportion of heat transfer is served for evaporation. The bubbles appear and grow on the active nucleation site until departure. Therefore, the evaporation heat flux is dependent on the nucleation site densityN_a, bubble departure diameter D_d, and bubble departure frequency f_d:

$$q_{Ev} = \frac{\pi}{6} D_d^3 \rho_v f_d N_a h_{fg}. \tag{18}$$

Selected empirical correlations for those nucleation phenomena are summarized in Table 3, Table 4, and Table 5, respectively. A more comprehensive review of those empirical correlations also can be found in [67]. It should also be noted that the evaporation heat transfer also serves as a void fraction source term in the mass conservation equation.

Table 3. Selected models for nucleation site density

Model	Empirical correlation for N_a, m^{-2}	Condition
Lemmert and Chawla [68]	$N_a = \left(aT_{\sup}\right)^b$, a = 210, b = 1.805	Pool boiling
Wang and Dhir [69]	$N_a = 5 \times 10^{-31}(1 - cos\theta)R_c^{-6.0}$	Pool boiling, p=1bar
Yang and Kim [70]	$N_a = N_{avg}\phi(\beta)exp(-CR_c)$	Pool boiling
Hibiki and Ishii [71]	$N_a = N_{avg}\left[1 - exp\left(-\frac{\theta^2}{8\mu_{con}^2}\right)\right]\left[exp\left(\frac{\lambda' g(\rho^+)}{R_c}\right) - 1\right]$	Pool and flow boiling, p ~ [1-198] bar

Table 4. Selected models for bubble departure diameter

Model	Empirical correlation for D_d, m	Condition
Cole and Rohsenow [72]	$D_d = 1.5 \times 10^{-4}\sqrt{\frac{\sigma}{g\Delta\rho}}\left(\frac{\rho_l C_{pl} T_{sat}}{\rho_g h_{fg}}\right)^{5/4}$	Pool nucleate boiling
Tolubinsky and Konstanchuk [73]	$D_d = \min[0.06exp(-\Delta T_{sub}/45), 0.14]$, mm	Subcooled flow boiling
Kocamustafaogullari [74]	$D_d = 1.27 \times 10^{-3}\left(\frac{\rho_l - \rho_g}{\rho_g}\right)^{0.9} d_{ref}$	Pool and flow boiling, p ~ 1-142 bar
Zeng et al.[75]	Mechanistic model bubble departure/lift-off based on force balance analysis	Pool and Flow boiling

Table 5. Selected models for bubble departure frequency

Model	Empirical correlation for f, s^{-1}	Condition
Cole[76]	$f_d = \sqrt{\frac{4g(\rho_l - \rho_g)}{3D_d\rho_l}}$	Pool nucleate boiling near CHF
Kocamustafaogullari and Ishii [16]	$f_d = \frac{1.18}{D_d}\left[\frac{\sigma g(\rho_l - \rho_g)}{\rho_l^2}\right]^{0.25}$	Subcooled _flow boiling
Podowski et al. [77]	Mechanistic model accounts for waiting time and bubble growth time	subcooled flow boiling

It is noted that in the "Generation-I" model, some of the important phenomena in flow boiling were not considered, such as the bubble sliding effect and the nucleation site interaction under high heat flux boiling. Some more recent efforts to resolve such issues have been made [13-15, 78]. The refined models resolve the underlying physics during nucleation and bubble growth. A common feature of those refined boiling models is the consideration of bubble sliding effect in flow boiling. The differences between those refined boiling models and the "Generation-I" model are summarized in Figure 14.

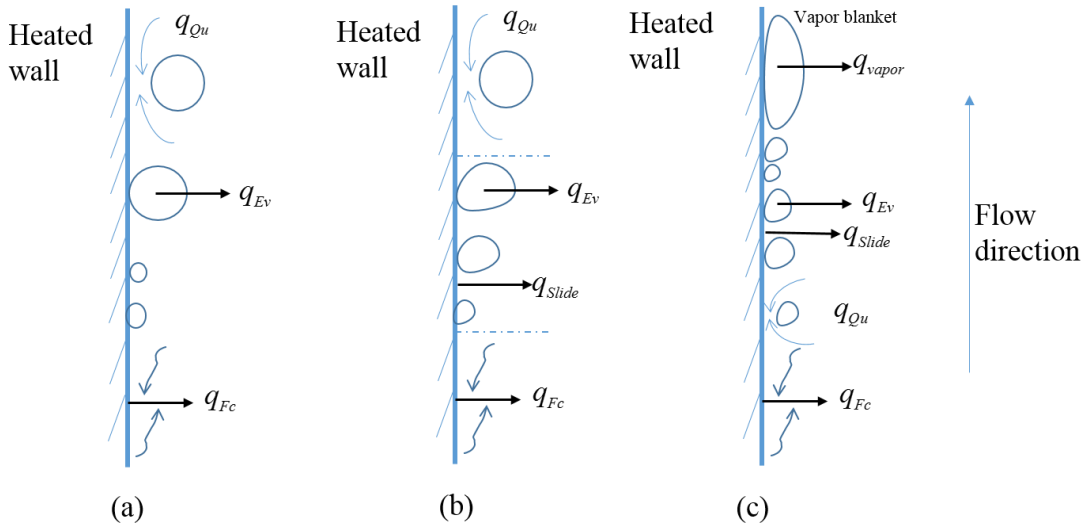

Figure 14. Illustration of heat partitioning in (a)." Generation-I" boiling model; (b). Refined boiling models; (c). Boiling model for DNB prediction.

To summarize, the MCFD solver is based on a system that consists of three conservative equations and a set of closures included in the equations. The solver discretizes and numerically solves this system. The empirical parameters in the closures contribute to a major source of uncertainty of the solver. Similar to the validation hierarchy, the role of closures in the system is also a reflection of the "divide-and-conquer" philosophy that decompose a complex system into several sub-phenomena and modeled them with closures separately. Effective as this approach is, it could underestimate or even ignore the interaction between different sub-phenomena. A consequence of this approach is the non-negligible model form uncertainty. In this sense, both the model parameter uncertainty and the model form uncertainty need to be considered in the VUQ of MCFD solver.

4.1.3 Characterization of uncertainties of MCFD solver

As discussed in Section 2.1, there are three sources of uncertainty for a general computational model: parameter uncertainty, model form uncertainty and numerical uncertainty. In MCFD solver, these uncertainties can be roughly characterized as follows.

One major source of uncertainty stems from the closure relations, which are largely empirical correlations, relying on historical data and published experiments; both the correlations and the data were generated decades ago, long before rigorous procedures for uncertainty analysis became required. Thus, the uncertainty comes from following issues:

- Common for legacy experiments is the lack of detail and accurate description of facility, measurement techniques, flow geometry, heater surface characteristics, inlet and boundary conditions. Consequently, it is not possible to evaluate the experimental data uncertainty, which is instrumental for determining uncertainty of models and model parameters. For safety analysis, conservative assumptions were usually invoked in selecting parameters of empirical correlations.
- Calibration of models involves a body of data, although not all data are born equal in their relevance to the conditions of interest. Value of information of data varies between experiments, tests (in an experiment), and types (even locations) of measurements (in a test).

- A major contribution to data uncertainty is the use of models outside the experimental domain, both interpolation and extrapolation. Even though various scaling techniques were devised and applied, complexity of two-phase flow and boiling heat transfer makes it difficult, if not impossible, to satisfy different scaling requirements.
- As a special case of scaling distortions, note that empirical correlations were mostly designed for steady or quasi-steady state, and fully-developed flow situations. However, there are many problems in two phase and boiling flow that are intrinsically unsteady or transient. Furthermore, developing flow is dominant in industrial systems. The effect is particularly notable for flow regime criteria.

Another group of uncertainty stems from basic assumptions that underlie the two-fluid modeling approach. This includes:

- The averaging procedure to obtain conservation equations, particularly the use of cross-sectional averaged variables in one-dimensional description of flow boiling. For different flow regimes, phase distributions across the channel can vary greatly and not well characterized for flow boiling in different transient scenarios.
- Another assumption is scale separation that decouples physics between "global" scale of fluid dynamics (given by the field equations' advection and diffusion terms) and "sub-grid-scale" local interactions (given by source terms). The source terms typically contain neither time nor spatial functional dependence. The scale separation allows source terms (e.g., interfacial exchanges, wall heat transfer, friction) in conservation equations be determined via local conditions, which – in one-dimensional model of flow boiling – are axially local and cross-sectional averaged.
- Interfacial exchange terms and wall terms are typically decomposed into components, which are considered independent, and whose effect on "global" fluid dynamics is additive. For example, the effect of wall heat transfer is averaged over the cross-section flow.

Note that the "divide-to-conquer" strategy popular in mechanistic modeling of various engineered systems. Their applicability (of scale separation and physics

decomposition) is limited by the complexity (non-linearly coupled/ multiscale nature) of two-phase flow, in general, and flow boiling in particular.

The numerical uncertainty stems from the numerical solution process of the two-fluid model's partial differential (conservation) equations. A major source of uncertainty results from the combined effect of discretization errors and errors due to approximation (integration) of closure relations over the numerical solution's finite difference or control volume. The errors increase with variability of different flow characteristics over the control volume.

The evaluation of numerical uncertainty of a MCFD solver is the same with other computational models, such as single phase CFD. The general tools including the manufactured solution [79], or the Richardson extrapolation [80]. Such evaluation process is considered as a topic of the verification. Thus, in this chapter, the numerical uncertainty is not explicitly analyzed. Rather, it is implicitly included in the model form uncertainty term. The validation and uncertainty quantification (VUQ) procedure discussed in the following section mainly deals with the parameter uncertainty and the model form uncertainty.

4.2 VUQ procedure for MCFD solver

As discussed in Section 1.1 the validation procedure for CFD based on the phenomena decomposition has been proposed for two decades [11]. Although the procedure has detailed guidance and solid theoretical background, limited validation practice has been conducted based on it. The major reason is the strict requirement of the validation data support, which current traditional experiments often cannot provide. Instead, a more convenient and straightforward validation paradigm has been adopted. This approach first identifies a set of closure relations that are considered to be important to the system of interest, then conduct parameter tuning on the closure relation based on available separate-effect test (SET) data. The tuned closure relations are then employed in the solver to obtain the QoIs and compare with the integrated-effect test (IET) data. This approach can produce reasonably good results with a limited data support. On the other hand, there are some shortcomings for this approach. The parameter tuning heavily depends on the researcher's experience and is generally providing *ad hoc* result. Also, the obtained results

cannot be updated with newly available data. Moreover, the possible interactions between the closure relations are neglected through this approach.

To overcome the shortcoming of the aforementioned traditional validation paradigm and to avoid the strict requirement for validation data, a new validation procedure based on the idea of total data-model integration (TDMI) is proposed. The TDMI approach treats closure relations, solver, and data in an integrated manner within the Bayesian procedure. Taking the MCFD solver as an example, for a given solver, the closure relation structure and the corresponding phenomena decomposition are already fixed. When using TDMI approach, the solver with its closures are treated as an integrated mathematical model which is like a "black-box" computational model with parameters serving as inputs and QoIs as outputs. This "black-box", along with available data, are employed in the Bayesian framework to quantify the uncertainty of both the influential parameters and the QoIs. This approach ensures the possible interaction between different closure relations to be considered, and the results can be automatically updated with newly available data. Another advantage of TDMI is its flexibility with data. It is capable to simultaneously take into account multiple QoIs measured under different conditions. Moreover, it can provide a reasonable VUQ result with limited available data, and can also give a more accurate result with better data support. The VUQ relationship between MCFD solver and data under TDMI approach is illustrated in Figure 15.

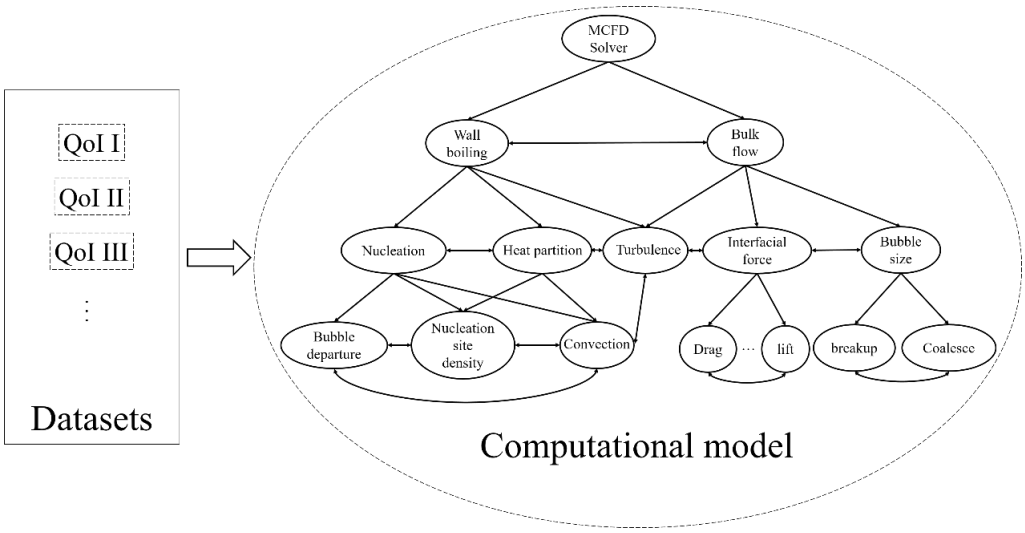

Figure 15. VUQ relationship of MCFD solver, closure relations, and data under TDMI approach.

It is obvious that in the TDMI approach, the validation is tightly coupled with the UQ process. The Bayesian method serves as the core method in this approach. The TDMI with the Bayesian method has been firstly demonstrated in [81] for the two-phase drift-flux model with synthetic data. The comparison between the traditional validation and TDMI based VUQ is illustrated in Figure 16.

The VUQ procedure is proposed as a six-step workflow. It is formed in a modular manner, which means each step of the workflow can be treated independently and the specific method applied in a step will not influence the following steps. For example, for the surrogate construction (step 2), one can choose a method that fits their problem best, such as Gaussian process, Stochastic collocation or support vector machine, and this choice will not influence the further steps of the work. Moreover, for each step, the non-intrusive method can be used which ensures the extendibility of the procedure to different MCFD solvers. The workflow of the procedure is summarized in Figure 17.

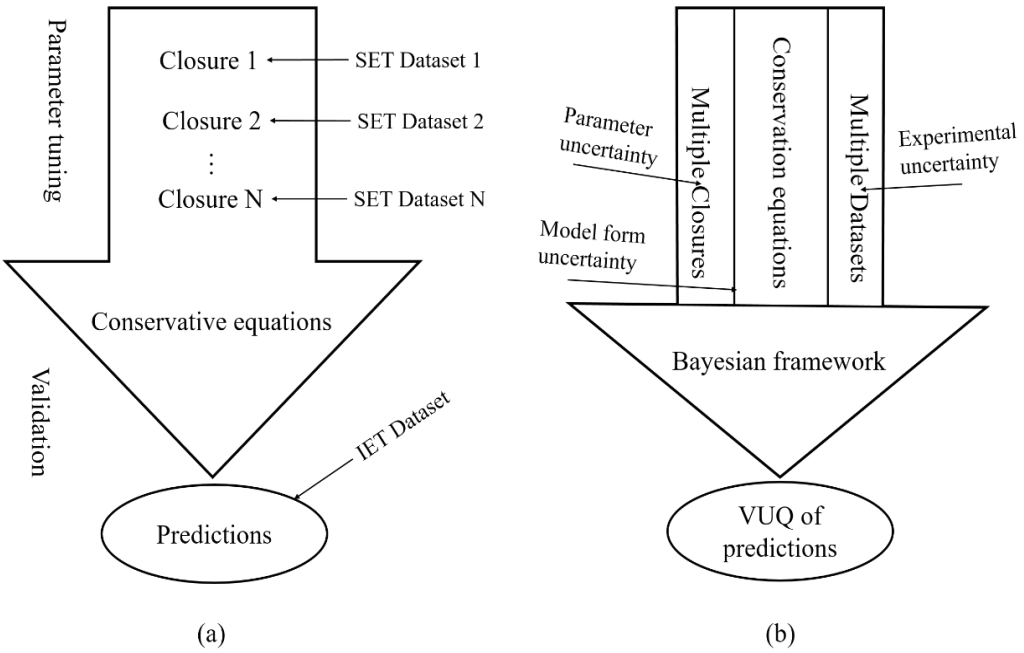

Figure 16. Two different validation paradigms: (a) Traditional validation; (b)VUQ based on total data-model integration.

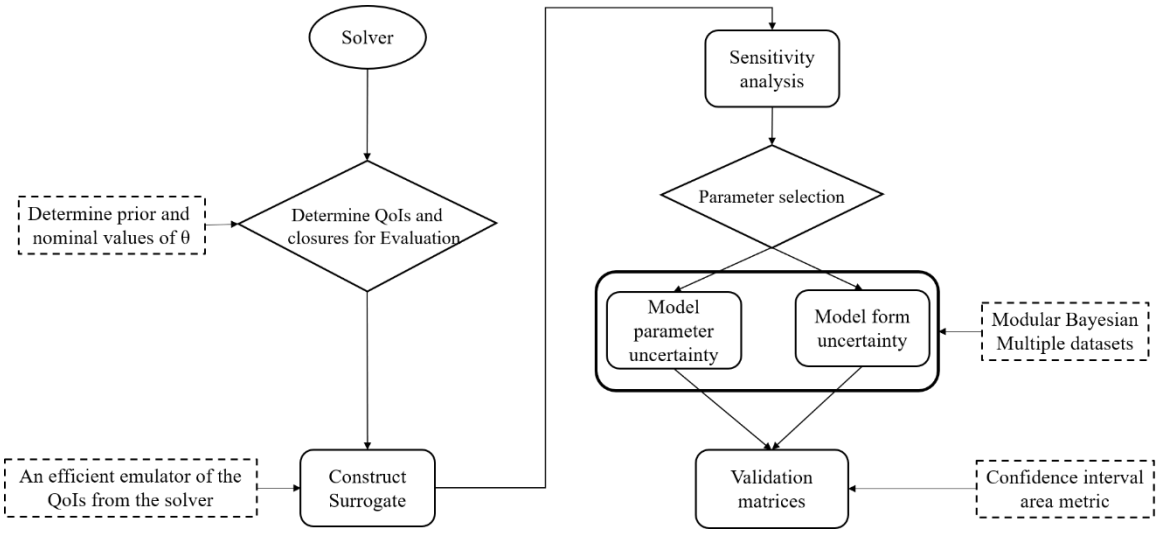

Figure 17. Workflow of the proposed VUQ procedure.

As discussed in the introduction, the ultimate goal of this VUQ procedure is to answer the question: How to evaluate if a MCFD adequately represents the underlying physics of a multiphase system of interest? In this section, this question is further decomposed into several smaller questions and is discussed in detail:

- How to choose relevant closure relations for a given scenario (section 4.2.1)
- How to build an accurate and efficient surrogate model for statistical inference? (section 4.2.2)
- How to identify influential and identifiable parameters (section 4.2.3 and section 4.2.4)
- How to quantify the model parameter uncertainty with the given available datasets? (section 4.2.5)
- How to evaluate the model form uncertainty which should be independent of parameters? (section 4.2.5)
- How to quantify the uncertainties of all the QoIs in a simultaneous manner? (section 4.2.5)
- How to build confidence in applying VUQ results on an untested condition? (section 4.2.5)?
- How to quantitatively validate the QoIs' full distribution against experimental measurements (section 4.2.6)?

4.2.1 First step: solver evaluation and data collection

The MCFD solver deals with many different scenarios related to multiphase flow, from adiabatic bubbly flow to subcooled boiling flow. For different problems, the closure relations used for simulation could be different and the Quantities of Interest (QoIs) would also vary. In this sense, the initial step in the procedure is evaluating the solver based on scenarios and collecting relevant experimental data to support the VUQ process.

Based on the studied scenario, several items should be addressed in this step:

- Evaluation of QoIs
- Collection of available experimental datasets
- Evaluation of closure relations
- Evaluation of input parameters

The QoIs of an MCFD simulation is scenario dependent and needs to be specified in the first place. For example, for boiling flow simulation, the wall superheat is considered as a QoI since it closely relates to safety, while the heat partitions are also QoIs since it relates to the heat transfer efficiency. For adiabatic bubbly flow, the interface distribution (characterized by void fraction) and the phasic flow field are QoIs. Once the QoIs for the given scenario is determined, the experimental measurement for the QoIs should be collected. For most cases, the resolution of the experimental measurement is coarse than the simulation results whose resolution can be easily controlled through mesh setup. The proposed VUQ procedure can deal with such limited data availability issue within the Bayesian framework. On the other hand, the VUQ results would be more accurate with the support of detailed measurement from validation experiments. Also, the procedure takes measurements from different conditions (e.g. different mass flow rate, heat flux, etc.) for the VUQ work simultaneously. This would generate robust VUQ results that can be extended to untested conditions.

As discuss in Section 4.1, there are many closure relations of different categories in a MCFD solver. Some of the closure relations have very limited influence on certain QoIs or are even not activated in certain scenarios. Thus, it is impractical and unnecessary to evaluate the parameters of all closure relations for a given scenario. The evaluation of

closure relations aims to identify closure relations that are relevant to the QoIs of a scenario. Once the relevant closure relation is evaluated, the uncertainties of the corresponding parameters are then inversely quantified through the Bayesian inference. To perform Bayesian inference, the empirical parameters are treated as random variables with given prior distribution that base on "expert judgment".

4.2.2 Second step: surrogate construction

The VUQ process requires many solver evaluations, considering the relative expensiveness of running a MCFD simulation, a surrogate model, also known as response surface model or emulator, is a necessary. In this step, a surrogate model is constructed based on the outputs of a limited number of runs of the original solver.

In some research area, the parameter number of a model can be very huge, for this type of model, the usual procedure is to do a simplified sensitivity analysis and parameter selection first to reduce the dimensionality of parameter space, then construct the surrogate on the reduced parameter space [82]. For MCFD solver, the parameter size is not that huge compared to these problems. Thus in this procedure, the surrogate is constructed with all parameters relevant to a certain scenario, then comprehensive sensitivity analysis and parameter selection are performed based on the surrogate.

There are multiple statistical and numerical methods that can be used for constructing a surrogate model. Each has various applications, such as polynomial response surface [83], stochastic collocation [84], and Gaussian Process [85], etc. In this work, we chose the Gaussian Process (GP) regression for surrogate construction, which has been widely used in the area of data-driven modeling and optimization. The only assumption for GP model is the QoIs are smooth over the whole input domain, which is generally valid for the considered problems.

The general form of a GP model can be expressed as

$$y^M(\mathbf{q}, \boldsymbol{\beta}) = \sum_{i=1}^{m} h_i(\mathbf{q})^T \beta_i + \mathbf{f}(\mathbf{q}) = \mathbf{h}(\mathbf{q})^T \boldsymbol{\beta} + Z(\mathbf{q}) , \tag{19}$$

where $\mathbf{q} = [\boldsymbol{q_1}, \boldsymbol{q_2}, \ldots, \boldsymbol{q_p}]$ is the p input variables, which can be empirical parameters or boundary conditions and $\boldsymbol{y^M}(\mathbf{q})$ is the model output with the given input. The first term on

the right side is a deterministic trend function, which is the product of regression coefficients $\boldsymbol{\beta}=[\boldsymbol{\beta_1}, \boldsymbol{\beta_2}, \ldots, \boldsymbol{\beta_m}]$and the basis function$\mathbf{h}(\mathbf{q}) = [\boldsymbol{h_1}(\mathbf{q}), \boldsymbol{h_2}(\mathbf{q}), \ldots, \boldsymbol{h_m}(\mathbf{q})]$, which has known form, usually set to be a constant or polynomial function. The second term Z(q) is a GP error model with zero mean, variance σ^2 and non-zero covariance $\mathbf{cov}[\boldsymbol{Z}(\mathbf{q^1}), \boldsymbol{Z}(\mathbf{q^2})]$, which can be modeled as

$$\mathrm{cov}\left[Z\left(\mathbf{q}^{(i)}\right), Z\left(\mathbf{q}^{(j)}\right)\right] = \sigma^2 K\left(\mathbf{q}^{(i)}, \mathbf{q}^{(j)}\right), \tag{20}$$

where $K(\mathbf{q}^{(i)}, \mathbf{q}^{(j)})$ is called kernel function, which is usually chosen to be a function of the distance between the two input vectors. This functional form ensures that two inputs with close distance will produce outputs that are also close together. Since not all input variables are equally important to the output, it is natural to introduce weighting factors for each input variable in the kernel. There are many forms of kernel functions, an example is the powered-exponential kernel

$$K\left(\mathbf{q}^{(i)}, \mathbf{q}^{(j)}\right) = \exp\left(-\sum_{k=1}^{p} \omega_k \left|q_k^{(i)} - q_k^{(j)}\right|^{\gamma_k}\right), \tag{21}$$

where ω_k is the weighting factors, and γ_k is termed "roughness factors", which influence the smoothness of the kernel function.

The two vectors ω and γ in equation (3), along with variance σ^2 are called hyperparameters of the GP model. The values of hyperparameters can be estimated by several methods such as Maximum Likelihood Estimation (MLE) or Bayesian inference.

As noted, a GP model needs a limited number of runs from original solver before it can be used to do prediction. Suppose *N* simulations of MCFD solvers are performed, the following matrices and vector are calculated:

Basis function:

$$\mathbf{H}=[\boldsymbol{h}(\mathbf{q}^1), \boldsymbol{h}(\mathbf{q}^2), \ldots, \boldsymbol{h}(\mathbf{q}^N)]=\begin{pmatrix} h_1(\mathbf{q}^1) & \ldots & h_1(\mathbf{q}^N) \\ \vdots & \ddots & \vdots \\ h_m(\mathbf{q}^1) & \cdots & h_m(\mathbf{q}^N) \end{pmatrix} \tag{22}$$

Kernel function:

$$\mathbf{K}=\begin{pmatrix} K(\mathbf{q}^{(1)},\mathbf{q}^{(1)}) & \dots & K(\mathbf{q}^{(1)},\mathbf{q}^{(N)}) \\ \vdots & \ddots & \vdots \\ K(\mathbf{q}^{(N)},\mathbf{q}^{(1)}) & \cdots & K(\mathbf{q}^{(N)},\mathbf{q}^{(N)}) \end{pmatrix} \tag{23}$$

Output QoI:

$$\mathbf{y}^{M} = [y^{M}(\mathbf{q}^{(1)}), y^{M}(\mathbf{q}^{(2)}),\dots, y^{M}(\mathbf{q}^{(N)})] \tag{24}$$

The regression coefficients $\boldsymbol{\beta}$ can be obtained through least-square estimate

$$\widehat{\boldsymbol{\beta}}=(\mathbf{H}^{\mathbf{T}}\mathbf{R}^{-1}\mathbf{H})^{-1}\mathbf{H}^{\mathbf{T}}\boldsymbol{K}^{-1}\mathbf{y}^{\mathrm{M}} \tag{25}$$

Thus, for a new unknown input $\boldsymbol{q}^{*}$, the GP model can give following prediction:

$$\hat{y}^{M}(\mathbf{q}^{*}) = \mathbf{h}(\mathbf{q}^{*})\widehat{\boldsymbol{\beta}} + K^{*T}\boldsymbol{K}^{-1}(\mathbf{y}^{M} - \mathbf{H}\widehat{\boldsymbol{\beta}}), \tag{26}$$

where

$$\mathrm{K}^{*}=\sigma^{2}[\mathrm{K}(\mathbf{q}^{*},\mathbf{q}^{(1)}),\mathrm{K}(\mathbf{q}^{*},\mathbf{q}^{(2)}),\dots,\mathrm{K}(\mathbf{q}^{*},\mathbf{q}^{(\mathrm{N})})]\,. \tag{27}$$

In addition, the GP model also give the variance of the predictor, which can be expressed as

$$\mathrm{Var}[(\hat{y}^{M})] = \sigma^{2}\left[1-(\mathbf{h}^{\boldsymbol{T}}(\mathbf{q}^{*}) \quad K^{*T})\begin{pmatrix} 0 & \mathbf{H}^{T} \\ \mathbf{H} & \mathbf{R} \end{pmatrix}^{-1}\begin{pmatrix} \mathbf{h}^{\boldsymbol{T}}(\mathbf{q}^{*}) \\ K^{*T} \end{pmatrix}\right]. \tag{28}$$

The accuracy of GP model in predicting the QoI with untried inputs can be evaluated through the cross-validation method. The details will be discussed in the specific applications detailed in Section 5.2.2.

For many cases, such as bubbly flow problem, the output QoI from MCFD solver is not only a function of boundary condition and empirical parameters but also a function of locations. Thus for even one single QoI, the output is a vector, with the size of the vector depending on the mesh setup. The straightforward approach is to train a GP model for every element of the vector respectively. This approach, however, is too cumbersome and

totally ignores the spatial correlation between those outputs. The dimension reduction method Principal Component Analysis (PCA) can be applied to resolve this issue.

The PCA uses an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables called principal components. The number of principal components is less than or equal to the smaller of the number of original variables or the number of observations. The Singular Value Decomposition (SVD) is a robust algorithm for PCA.

The implementation of PCA to the MCFD solver output can be summarized as several steps. Firstly, concatenate the output vectors of different QoIs to form a long vector; e.g.:

$$\mathbf{y}^M = \left[\alpha_1, \alpha_2, \ldots \alpha_d, U_{g1}, U_{g2}, \ldots U_{gd}\right]^T, \tag{29}$$

where α_{i} and U_{gi} are the void fraction and gas velocity in mesh index i, respectively. Thus the length of the output vector is $n \times d$ where n is the number of QoIs and d is the number of meshes.

Evaluate the output vector with N different MCFD simulations. Thus an output matrix $\mathbf{Y}$ with dimension $nd \times N$ can be constructed:

$$\mathbf{Y} = \left(\mathbf{y}^M(\mathbf{q}^{(1)}), \mathbf{y}^M(\mathbf{q}^{(2)}), \ldots, \mathbf{y}^M(\mathbf{q}^{(N)})\right). \tag{30}$$

Center the output matrix by subtracting the mean column vector $\bar{\mathbf{y}}^M = \frac{1}{N}\sum_{i=1}^{N} y^M(\mathbf{q}^{(i)})$in each column to obtain $\mathbf{Y_c}$. Then Perform SVD on $\mathbf{Y_c}$:

$$\mathbf{Y_c} = \mathbf{U\Sigma V}^{\boldsymbol{T}}. \tag{31}$$

This means the centered output matrix $\mathbf{Y_c}$ can be decomposed as the products of three matrices, where $\mathbf{\Sigma}$ is a $nd \times N$diagonal matrix whose diagonal entries are non-negative and ordered from largest to smallest, those diagonal entries are known as singular values of $\mathbf{Y_c}$ which are the root of the eigenvalues of $\mathbf{Y_c}\boldsymbol{Y_c^T}$, whereas the eigenvalues are the measure of the variances of $\mathbf{Y_c}$. $\mathbf{U}$ is a $nd \times nd$unitary matrix, whose columns are the left-singular vectors of $\mathbf{Y_c}$ and $\mathbf{V}$ is a $N \times N$ unitary matrix whose column vectors are the right-singular vectors of $\mathbf{Y_c}$. The column vectors of $\mathbf{U}$ are called the Principal Components (PCs).

Since **U** is unitary matrix, it has the property for such manipulation:

$$\mathbf{U}^T\mathbf{Y}_c = \mathbf{U}^T\mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T = \boldsymbol{\Sigma}\mathbf{V}^T = \mathbf{S}\,. \tag{32}$$

This means the PCs map each row vector of $\mathbf{Y_c}$ to a new vector (the row vector in **S**) which is termed PC scores. The PC scores are the transformed representations of the original input matrix $\mathbf{Y_c}$. For most cases, the singular values of $\mathbf{Y_c}$ decrease very quickly which means the first few PCs can quantify the structures of the $\mathbf{Y_c}$. So the manipulations in Eq.(32) can be modified as:

$$\mathbf{U}^{*T}\mathbf{Y}_c = \mathbf{S}^*\,, \tag{33}$$

where $\mathbf{U}^*$ is a $nd \times d^*$ matrix with first d^* PCs, $\mathbf{S}^*$ is the corresponding $d^* \times N$ matrix with first d^* PC scores. In practice, one usually retains the first d^* PCs to ensure the corresponding variances can account for 95% - 99.9% of the total variance. In this way, the dimension of outputs can be reduced from nd to d^*. This means one only need to train d^* uncorrelated GP models instead of nd highly correlated ones.

In training GP models for the PC scores, each column in $\mathbf{S}^*$serve as a sample. For a new input, the GP model give predictions for the corresponding PC scores $\mathbf{S}^*$ can be transformed back from the PC subspace to the original space by the following manipulation:

$$\mathbf{y}^M = \mathbf{U}^*\mathbf{s}^* + \bar{\mathbf{y}}^M \tag{34}$$

4.2.3 Third step: sensitivity analysis

Once the surrogate model for the MCFD solver is constructed, Sensitivity Analysis (SA) regarding the empirical parameters would be performed based on it. The general objective of SA is to quantify the individual parameter's contribution towards the QoIs and determine how variations in parameters affect the QoIs.

In this procedure, the Global SA (GSA) is conducted which considers the QoI uncertainties due to combinations of parameters throughout the whole admissible input space. It should also be noted that when conducting SA, the prior distributions and nominal values of all parameters are already determined in Step 1. In the following analysis, the

input variables $\boldsymbol{q}$ used in constructing GP model are further characterized by parameters $\boldsymbol{\theta}$ (suppose have p dimensions) and condition variable $\boldsymbol{\nu}$ (such as mass flow rate, wall heat flux, etc.) and spatial dependent variable $\boldsymbol{x}$, so that $\mathbf{y}^M(\mathbf{q})$ is expressed as $\mathbf{y}^M(x, \mathbf{\nu}, \theta)$. There are two different methods for GSA: the Morris screening method and the Sobol indices method.

Morris Screening method

Morris Screening method evaluates local sensitivity approximations, termed elementary effects, over the input space. Morris Screening method can rank parameters according to their importance, but cannot quantify how much one parameter is more important to another. The major advantage of Morris Screening is its low computational cost.

Morris Screening is based on the linearization of the model. To construct the elementary effect, one partitions [0,1] into l levels. Thus the elementary effect associated with the i^{th} input can be calculated as

$$d_i(\boldsymbol{\theta}) = \frac{\int y^M\left(x, \mathbf{v}, \left[\theta_1, \ldots, \theta_{i-1}, \theta_i + \Delta, \theta_{i+1}, \ldots \theta_p\right]\right)dx - \int y^M(x, \mathbf{v}, \boldsymbol{\theta})dx}{\Delta}, \tag{35}$$

which means the QoIs are integrated over the whole input condition space for evaluation. The step size Δ is chosen from the set

$$\Delta \in \left\{\frac{1}{l-1}, \ldots, 1 - \frac{1}{l-1}\right\}. \tag{36}$$

For r sample points, the sensitivity measurement for x_i can be represented by the sampling mean μ_i, standard deviation σ_i^2, and mean of absolute values μ_i^*, which can be calculated respectively.

$$\mu_i^* = \frac{1}{r}\sum_{j=1}^{r} |d_i^j(\mathbf{q})| \tag{37}$$

$$\mu_i = \frac{1}{r}\sum_{j=1}^{r} d_i^j(\mathbf{q}) \tag{38}$$

$$\sigma_i^2 = \frac{1}{r-1}\sum_{j=1}^{r}\left(d_i^j(\mathbf{q}) - \mu_i\right)^2 \tag{39}$$

The mean represents the effect of the specific parameter on the output, while the variance represents the combined effects of the input parameters due to nonlinearities or interactions with other inputs. The obtained μ^* and σ can help ranking the parameters by the relative order of importance. If the value of sigma is high compared to mu (same order of magnitude), a non-linear influence and/or interactions with other parameters are detected. This measure is however only qualitative.

Variance based method: Sobol indices

Variance-based methods decompose the output variance into contributions of the input variances, in this method, the importance of parameter can be quantitatively evaluated. The Sobol indices method is one of the most widely used variance based method on GSA. In here the basic idea of Sobol indices is discussed, assuming the parameters are independent and uniformly distributed on [0,1]. A more general situation is discussed in [82].

A general computational model $y^M(\boldsymbol{q})$ can be expressed with second-order Sobol expansion:

$$y^M(\boldsymbol{q}) = f_0 + \sum_{i=1}^{p} f_i(q_i) + \sum_{1\le i<j\le p} f_{ij}(q_i, q_j)\,, \tag{40}$$

where the zeroth-, first-, and second-order terms on the right-hand side can be expressed as

$$\begin{aligned} f_0 &= \int_{\Gamma^p} y^M(\boldsymbol{q})d\boldsymbol{q}\,, \\ f_i(q_i) &= \int_{\Gamma^{p-1}} y^M(\boldsymbol{q})d\boldsymbol{q}_{\sim i} - f_0\,, \\ f_{ij}(q_i, q_j) &= \int_{\Gamma^{p-2}} y^M(\boldsymbol{q})d\boldsymbol{q}_{\sim i} - f_i(q_i) - f_j(q_j) - f_0\,, \end{aligned} \tag{41}$$

where $\Gamma^p = [0,1]^p$. The notation $\boldsymbol{q}_{\sim i}$ denotes the vector having all the components of $\boldsymbol{q}$ except i^{th} element.

The total variance D of the model prediction y^M is

$$D = \text{var}(y^M) = \int_{\Gamma^p} {y^M}^2(\boldsymbol{q}) d\boldsymbol{q} - f_0^2 , \tag{42}$$

which can be further expressed as

$$D = \sum_i D_i + \sum_{1 \le i < j \le p} D_{ij} , \tag{43}$$

where D_i and D_{ij} are two partial variances can be expressed as

$$\begin{aligned} D_i &= \int_0^1 f_i^2(q_i) dq_i \\ D_{ij} &= \int_0^1 f_{ij}^2(q_i, q_j) dq_i \end{aligned} . \tag{44}$$

The Sobol indices are defined to be

$$S_i = \frac{D_i}{D}, \ S_{ij} = \frac{D_{ij}}{D}, \ i, j = 1, \ldots, p \tag{45}$$

By definition, those indices satisfy

$$\sum_{i=1}^{p} S_i + \sum_{1 \le i < j \le p} S_{ij} = 1 , \tag{46}$$

where S_i is termed the *first-order sensitivity indices*, large value of S_i indicate strong influence of the corresponding parameter on the model prediction. S_{ij} measures the influence of interaction between two corresponding parameters. Based on this two terms, the *total sensitivity indices* can be calculated to quantify the total effect of the corresponding parameter on model prediction

$$S_{T_i} = S_i + \sum_{j=1}^{p} S_{ij} \tag{47}$$

In practice, both the first-order indices and total sensitivity indices are considered using this method. The algorithm proposed by [86] can be used to calculate Sobol indices.

Firstly, create two $M \times p$ sample matrices through random sampling, where M is the sample size, p is the number of parameters investigated.

$$\mathbf{A} = \begin{pmatrix} \theta_1^1 & \dots & \theta_i^1 & \dots & \theta_p^1 \\ \vdots & & \vdots & & \vdots \\ \theta_1^M & \dots & \theta_i^M & \dots & \theta_p^M \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} \hat{\theta}_1^1 & \dots & \hat{\theta}_i^1 & \dots & \hat{\theta}_p^1 \\ \vdots & & \vdots & & \vdots \\ \hat{\theta}_1^M & \dots & \hat{\theta}_i^M & \dots & \hat{\theta}_p^M \end{pmatrix}. \tag{48}$$

Secondly, create p different $M \times p$ matrices based on A and B.

$$C_i = \begin{pmatrix} \hat{\theta}_1^1 & \dots & \theta_i^1 & \dots & \hat{\theta}_p^1 \\ \vdots & & \vdots & & \vdots \\ \hat{\theta}_1^M & \dots & \theta_i^M & \dots & \hat{\theta}_p^M \end{pmatrix}, \tag{49}$$

which is identical to B except the i^{th} column which is taken from A.

Then compute $M \times 1$ vectors of model prediction:

$$y_A = f(A), y_B = f(B), y_C = f(C) \tag{50}$$

The total number of model evaluations is $M(p + 2)$.

The first-order sensitivity indices can be estimated as

$$S_i = \frac{\frac{1}{M}\sum_{j=1}^{M} y_A^j y_{C_i}^j - f_0^2}{\frac{1}{M}\sum_{j=1}^{M} (y_A^j)^2 - f_0^2}, \tag{51}$$

where f_0 is the mean of model prediction which can be approximated as

$$f_0^2 = \left(\frac{1}{M}\sum_{j=1}^{M} y_A^j\right)\left(\frac{1}{M}\sum_{j=1}^{M} y_B^j\right). \tag{52}$$

The total sensitivity indices can be estimated as

$$S_{T_i} = 1 - \frac{\frac{1}{M}\sum_{j=1}^{M} y_B^j y_{C_i}^j - f_0^2}{\frac{1}{M}\sum_{j=1}^{M} (y_A^j)^2 - f_0^2}. \tag{53}$$

4.2.4 Fourth step: parameter selection

For a complex system such as the MCFD solver, many empirical parameters exist in the closure relations, and the parameter identifiability arises as a major issue for conducting Bayesian inference. This issue stems from the fact that with a limited number of datasets, there could exist different values of parameters that produce very similar results and fit the data equally well. The convergence of Bayesian inference would face difficulty with non-identifiable parameter exists, unless good prior distributions are provided. Moreover, for engineering applications, the VUQ results should provide guidance for setting up parameters for future cases, thus a subset of parameters quantified with small uncertainties is more helpful for engineering simulation compared to the complete set of parameters quantified with large uncertainties. In this sense, it is desired to find a subset of parameters that not only can be identified using measured data but also with high sensitivity so that the Bayesian inference would give smaller posterior uncertainties of parameters. In this sense, the parameter selection is closely related to SA. The objective of this step is to perform SA and select a subset of parameters based on SA for the Bayesian inference in next step.

It is natural to select parameters based on the GSA results. The general idea is to select parameters with high impact to the QoIs which are identifiable in the sense that they can be uniquely determined by the data. However, even two highly influential parameters cannot guarantee that they are mutually identifiable. A most straightforward example is $y = a + b$, where a and b are of equal importance, but they are not identifiable with each other. Thus based on the GSA results, the parameter selection is still an *ad hoc* solution which requires trial-and-error.

An algorithm for parameter selection has been developed by [87], which is based on the local sensitivity matrix, instead of the GSA results. The selection scores of different combinations of parameters can be obtained.

The local sensitivity around nominal parameter values $\mathbf{q}_0$ can be analyzed based on the sensitivity matrix which is defined as

$$\chi(\boldsymbol{\theta}_0) = \begin{pmatrix} \frac{\partial y^M}{\partial \theta_1}(x_1, v, \boldsymbol{\theta}_0) & \cdots & \frac{\partial y^M}{\partial \theta_p}(x_1, v, \boldsymbol{\theta}_0) \\ \vdots & \ddots & \vdots \\ \frac{\partial y^M}{\partial \theta_1}(x_n, v, \boldsymbol{\theta}_0) & \cdots & \frac{\partial y^M}{\partial \theta_p}(x_n, v, \boldsymbol{\theta}_0) \end{pmatrix} \tag{54}$$

Based on sensitivity matrix $\chi(\boldsymbol{\theta}_0)$, the Fisher information matrix is

$$\mathbf{F}(\boldsymbol{\theta}_0) = \frac{1}{s_0^2}\left(\chi^T(\boldsymbol{\theta}_0)\chi(\boldsymbol{\theta}_0)\right), \tag{55}$$

where s_0^2 is the estimated error variance:

$$s_0^2 = \frac{1}{n-p}\sum_{i=1}^{n}(y^E(v_i) - y^M(v_i, \boldsymbol{\theta}_0))^2 \tag{56}$$

The covariance matrix $\mathbf{V}$ can be estimated as

$$\mathbf{V}(\boldsymbol{\theta}_0) = s_0^2\left(\chi^T(\boldsymbol{\theta}_0)\chi(\boldsymbol{\theta}_0)\right)^{-1} = \mathbf{F}^{-1}(\boldsymbol{\theta}_0). \tag{57}$$

The coefficient of variation can be defined as:

$$\upsilon(\boldsymbol{\theta}_0)_i = \frac{\left(\mathbf{V}(\boldsymbol{\theta}_0)_{i,i}\right)^{1/2}}{(\boldsymbol{\theta}_0)_i}, \tag{58}$$

where $\mathbf{V}(\boldsymbol{\theta}_0)_{i,i}$ is the i^{th} diagonal entry in the covariance matrix. The coefficient of variation $\upsilon(\boldsymbol{\theta}_0)_i$ represents the ratio of standard error of θ_i to its nominal value. Based on it, the parameter selection score $\beta(\boldsymbol{\theta}_0)$ can be defined as the Euclidean norm of the vector $\upsilon(\boldsymbol{\theta}_0)$:

$$\beta(\boldsymbol{\theta}_0) = \|\upsilon(\boldsymbol{\theta}_0)\|_{l-2} = \sqrt{\sum_{i=1}^{p}(\upsilon(\boldsymbol{\theta}_0)_i)^2} \tag{59}$$

If one wants to select a subset of parameters of size p^* from the whole parameters (of size p), the parameter selection based on $\beta(\boldsymbol{\theta}_0)$ can be accomplished by the following algorithm:

1). For $p^* < p$, consider all possible choices of indices i_1 , i_2,…, i_{p^*} with lexicographical order, which is the enumeration of the combination $\binom{p}{p^*}$.

2). Initialize the minimum selection score $\beta^{sel}=\infty$ and the selected index vector ind^{sel} to be (1,2,…, p^*).

3). Start with the first choice $\left(i_1^{(k)}, i_2^{(k)}, \dots, i_{p*}^{(k)}\right)$ and completes the following steps:

Step k: for index $\left(i_1^{(k)}, i_2^{(k)}, \dots, i_{p*}^{(k)}\right)$, compute $r = rank(\mathbf{F}(\boldsymbol{\theta}_{0,i_1^{(k)}}, \boldsymbol{\theta}_{0,i_2^{(k)}}, \dots, \boldsymbol{\theta}_{0,i_{p*}^{(k)}})$

if $r < p^*$ (which means selected parameters in this step are not mutually identifiable), go to step k+1

if $r = p^*$, compute the corresponding selection score $\beta_k = \beta(\boldsymbol{\theta}_{0,i_1^{(k)}}, \boldsymbol{\theta}_{0,i_2^{(k)}}, \dots, \boldsymbol{\theta}_{0,i_{p*}^{(k)}})$:

if $\beta_k > \beta^{sel}$, go to step k+1

if $\beta_k < \beta^{sel}$, set $\beta^{sel}=\beta_k$ and ind^{sel} to be $\left(i_1^{(k)}, i_2^{(k)}, \dots, i_{p*}^{(k)}\right)$, then go to step k+1.

Through this algorithm, the subset of parameters with size p^* with minimum selection score can be selected, the low score means low uncertainty probabilities in the estimation. The choice of p^* depends on the rank analysis of the full Fisher information matrix $\mathbf{F}(\boldsymbol{\theta}_0)$.

Compared to the trial-and-error approach based on GSA, this algorithm also has its drawbacks. Since the algorithm is based on the nominal parameter value $\boldsymbol{\theta}_0$, it does not fully explore the whole parameter space. In this sense, there is a possibility that such selection may be misleading if the parameter posterior distribution deviates significantly to the nominal value. Moreover, since there are multiple QoIs taken into consideration, the "optimal" parameter subset for different QoI could be different. In this sense, subjectivity cannot be avoided in the parameter selection. To sum up, the parameter selection process introduced in this work can be regarded as an "expert judgment informed by sensitivity analysis" procedure. In the following Bayesian inference step, only the selected parameters are treated as random variables while other parameters are fixed at their nominal values.

4.2.5 Fifth step: Uncertainty quantification

For a general computational model, the relationship between the outputs and experimental measurements can be expressed as

$$\mathbf{y}^E(\mathbf{x},\mathbf{v}) = \mathbf{y}^M(\mathbf{x},\mathbf{v},\boldsymbol{\theta}) + \delta(\mathbf{x},\mathbf{v}) + \varepsilon(\mathbf{x},\mathbf{v}) , \tag{60}$$

where $\mathbf{y}^E(\mathbf{x},\mathbf{v})$ is the experimental measurement, $\mathbf{y}^M(\mathbf{x},\mathbf{v},\boldsymbol{\theta})$ is the model prediction, $\delta(\mathbf{x},\mathbf{v})$ is the model form uncertainty, which is usually caused by missing physics, simplified assumptions or numerical approximations in the model, $\varepsilon(\mathbf{x},\mathbf{v})$ is the measurement uncertainty which is assumed to be i.i.d (independent and identically distributed) normal distributions with zero means and know variance σ^2 in this work:

$$\varepsilon(\mathbf{x},\mathbf{v}) \sim N(0,\sigma^2\mathbf{I}) \tag{61}$$

The goal in inverse UQ process is to evaluate the uncertainty of the parameter based on the data. In the framework of Bayesian inference, which treats the parameter as random variables, the prior knowledge for the parameter is also considered. The prior knowledge usually comes from previous simulations, other experimental observations or purely expert opinion. The Bayes formula is the foundation of Bayesian inference:

$$p(\boldsymbol{\theta}|\mathbf{y}^E) = \frac{p(\mathbf{y}^E|\boldsymbol{\theta})p_0(\boldsymbol{\theta})}{p(\mathbf{y}^E)} \propto p(\mathbf{y}^E|\boldsymbol{\theta})p_0(\boldsymbol{\theta}) , \tag{62}$$

where $p(\mathbf{y}^E|\boldsymbol{\theta})$is the likelihood function, and $p_0(\boldsymbol{\theta})$is the prior distribution of $\boldsymbol{\theta}$. Under the assumption that $\varepsilon(\mathbf{x},\mathbf{v}) \sim N(0,\sigma^2\mathbf{I})$, the likelihood function has the form

$$L(\boldsymbol{\theta}|\mathbf{y}^E) = \frac{1}{(2\pi\sigma^2)^{n/2}} \exp\left(- \sum_{i=1}^{n} \frac{\left(y^E(x_i,\mathbf{v}) - y^M(x_i,\mathbf{v},\boldsymbol{\theta}) - \delta(x_i,\mathbf{v})\right)^2}{2\sigma^2} \right), \tag{63}$$

If we only care about the point estimate of the parameter, the Maximum a posteriori (MAP) method can be applied. If the prior is set to be uniform, the MAP can be obtained by maximizing Eq.(64). Due to the monotonicity of the logarithm function, it is numerically advantageous to maximize the log-likelihood function, which can be expressed as

$$l(\boldsymbol{\theta}|\mathbf{y}^E) = -\frac{n}{2}\ln(2\pi) - \frac{n}{2}\ln(\sigma^2) - \sum_{i=1}^{n} \frac{(y^E(x_i,\mathbf{v}) - y^M(x_i,\mathbf{v},\boldsymbol{\theta}) - \delta(x_i,\mathbf{v}))^2}{2\sigma^2}. \quad (64)$$

On the other hand, the full distribution of the posterior is more difficult to obtain. For most cases, the posteriori cannot be direct calculated. Based on the likelihood function and prior distribution, the posterior distribution of the parameter can be drawn through the Markov Chain Monte Carlo (MCMC) sampling. However, a still unresolved issue is the unknown term model form uncertainty $\delta(\mathbf{x},\mathbf{v})$. The investigation of the model form uncertainty is an active topic in the statistics community. The intrinsic difficulty for this problem is the confounding between model form uncertainty $\delta(\mathbf{x},\mathbf{v})$ and the parameter $\boldsymbol{\theta}$. In other words, if discrepancy is observed between the model prediction and experimental data, there is no way to distinguish if this is caused by mode form uncertainty or a "poorly" chosen model parameter without given any prior knowledge.

The data driven approach, which use a Gaussian process to evaluate the model form uncertainty term $\delta(\mathbf{x},\mathbf{v})$, has been widely used since proposed by [44]. This data driven approach can be further divided into two types. One approach is often called "full Bayesian" [88], which treats the hyperparameters of the GP for $\delta(\mathbf{x},\mathbf{v})$ as random variables with specified prioris and infers their posterior distributions in the same MCMC process for the inference of empirical parameters. For problems with rich data sources, such as weather forecasting, those hyperparameters can have noninformative prior in the Bayesian inference. For other cases where the data sample is not that rich, tight priors need to be assigned to those hyperparameters. Such prior setup reflect the assumption that one has good understanding on the model form uncertainty terms. The other approach is termed "modular Bayesian" approach [89], which firstly evaluate the hyperparameters in $\delta(\mathbf{x},\mathbf{v})$ with the point estimate method such as MLE to get a fixed form of $\delta(\mathbf{x},\mathbf{v})$. Then performs MCMC for only empirical parameters with the $\delta(\mathbf{x},\mathbf{v})$ term included in the process. The "modular Bayesian" approach doesn't require a rich data support of strong prior for the hyperparameters and is thus considered to be a more realistic and feasible approach

compared to "full Bayesian". In this work, the "modular Bayesian" approach is adopted for the treatment of $\delta(\mathbf{x}, \mathbf{v})$ which can be summarized in following steps:

1. Split the available datasets into three groups: one for parameter inverse UQ, one for model form uncertainty evaluation, one for testing.
2. The model form uncertainty term $\delta(\mathbf{x}, \mathbf{v})$ is modeled by GP (noted that this GP model is independent with the surrogate model) based on the training datasets: $y^E(\mathbf{x}, \mathbf{v}) - y^M(\mathbf{x}, \mathbf{v}, \boldsymbol{\theta}_0)$ with all the parameter fixed at their nominal values. The hyperparameters are obtained using MLE method and keep fixed.
3. obtain the posterior uncertainty distributions of parameters using Bayesian inference. When performing Bayesian inference on the parameters, $\delta(\mathbf{x}, \mathbf{v})$ is introduced in as shown in Eq.(60).
4. Propagate the uncertainty of the parameters through the solver (which is represented by the surrogate model) to obtain the uncertainty of the QoIs.

The procedure is summarized in Figure 18.

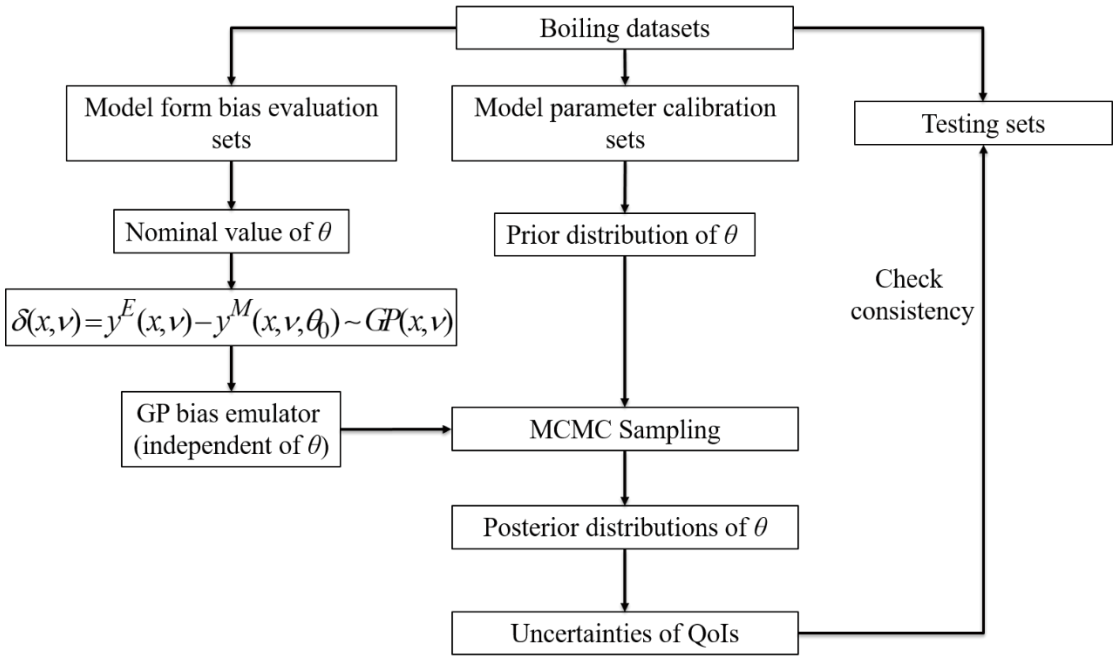

Figure 18. Evaluation of model form uncertainty and model parameter uncertainty.

The general idea of MCMC is to construct Markov chains that converge to the posterior parameter distributions. For a given parameter, it is proved that the stationary distribution of the Markov chains is the posterior density. There are multiple algorithms for MCMC sampling; in this work, the Delayed Rejection Adaptive Metropolis (DRAM) algorithm [90] is used. Once the posterior distributions of parameters are obtained, the forward UQ can be applied to obtain the distributions of QoIs based on the surrogate model.

4.2.6 Sixth step: validation metrics

Once the uncertainties of QoIs are obtained, the last step of the procedure is to quantitatively evaluate the agreement between QoIs predicted by solver and the experimental measurements. This is done by calculating the validation metrics.

Generally speaking, validation metrics currently applied to scientific computation problems can be characterized into three categories. The first type is hypothesis testing. In this type of validation metric, two hypotheses are constructed. The first one is called null hypothesis which is initially assumed to be true and usually it is set to be "the model is in agreement with the observed data". The second one is called alternative hypothesis, which contradicts the null hypothesis, such as "the model is not in agreement with the observed data". Hypothesis testing based on the observed data to construct a test statistics *S* based on which to decide whether to accept or reject the null hypothesis, thus the outcome of this type of validation metric is only a "Yes or No" statement. The second type is Confidence Interval (CI) proposed by [41] which measures the discrepancy between the mean of predicted QoIs and the experimental data, plus the uncertainty of measurement. The confidence interval can be constructed as

$$\left(\tilde{\mathbf{E}} - t_{\alpha/2,\nu} \cdot \frac{\mathbf{s}}{\sqrt{n}}, \tilde{\mathbf{E}} + t_{\alpha/2,\nu} \cdot \frac{\mathbf{s}}{\sqrt{n}}\right), \tag{65}$$

where $\tilde{\mathbf{E}}$ is the estimated error between model and data,

$$\tilde{\mathbf{E}} = \mathbf{y}^M\left(\mathbf{x}, \mathbf{v}, \bar{\boldsymbol{\theta}}_{post}\right) - \bar{\mathbf{y}}^E(\mathbf{x}, \mathbf{v}), \tag{66}$$

s is the standard deviation of the experimental data, $t_{\alpha/2,\nu}$ is the 1-α/2 quantile of the t-distribution with freedom of ν used to quantify the uncertainty of experimental data. The

obtained CI can be interpreted as "we have $(1-\alpha)\times 100\%$ confidence that the true discrepancy between model and observed data is within the interval".

The third type is termed area metric, which is proposed by [42]. In this type of validation metric, both the experimental data and model predicted QoIs are treated as random variables, whose probability distribution is their uncertainty distribution. The area metric measures the area between the two Cumulative Distribution Functions (CDFs), which can be expressed as

$$d\left(F_{x_i}^E, F_{x_i}^M\right) = \int_{-\infty}^{+\infty} \left|F_{x_i}^E(x) - F_{x_i}^M(x)\right| dx \,. \tag{67}$$

One of a major merit of the area metric is that it takes the full uncertainty distribution of both data and model prediction into consideration. It also needs to mention that there are also other forms of validation metrics such as u-pooling and p-box [43] which are extensions of the area metrics.

In this practice, two types of validation metric are calculated. The first type is Confidence interval. For experimental data obtained from literature, the detailed measurement information is usually not available. Thus the uncertainty of experimental data is assumed to follow normal distribution instead of t-distribution. The Eq.(65) based on t-distribution is modified to following

$$\left(\tilde{\mathbf{E}} - z_{\alpha/2}\cdot\sigma, \tilde{\mathbf{E}} + z_{\alpha/2}\cdot\sigma\right), \tag{68}$$

where σ is the standard deviation of the experimental data, and $z_{\alpha/2}$ is the 1-α/2 quantile of the statistic distribution of the experimental data. The obtained CI can be interpreted as "we have $(1-\alpha)\times 100\%$ confidence that the true discrepancy between model and observed data is within the interval".

The second type applied in the procedure is the area metric. In this type of validation metric, both the experimental data and model predicted QoIs are treated as random variables, whose probability distribution is their uncertainty distribution. The area metric measures the area between the two Cumulative Distribution Functions (CDFs), which can be calculated with Eq.(67).

The validation metrics are calculated for both pure model predictions and model prediction plus model form uncertainty. The discrepancy between these two different calculations reflects the closure relations' ability to capture the corresponding physical process in the given application domain. An example of the two different validation metrics is provided in Figure 19.

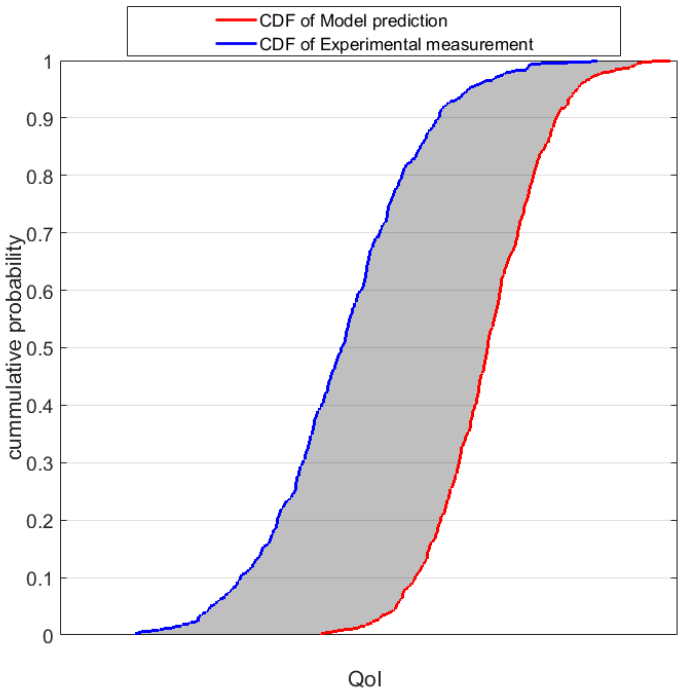

Figure 19. Example of area metric.

4.3 Summary remarks

In this chapter, a VUQ procedure designed for Eulerian-Eulerian two-fluid-model based MCFD solver is proposed. The gold of the procedure is to evaluate whether the solver represents the underlying physics of a multiphase flow and boiling system of interest with acceptable accuracy. This is accomplished through a six step procedure with two major results obtained: i). quantify the uncertainties of the closure parameters and predictions of the MCFD solver; ii). evaluate the agreement between the solver predictions and the experimental measurements.

There are several advantages of the proposed procedure. Firstly, it is modular and non-intrusive. This ensures the procedure's flexibility for different scenarios as well as its extendibility to different MCFD solvers. Secondly, the procedure also has strong flexibility

with data. It is capable of simultaneously taking into account multiple QoIs measurement under different conditions. Moreover, it can provide a reasonable VUQ result with limited data availability, while providing a more accurate result with better data support. Thirdly, the TDMI treatment ensures the VUQ considers the possible closure interactions in the MCFD solver. Last, the model form uncertainty is considered which can serve as an additional correction term to the model prediction. The validation metrics differences between pure model prediction and prediction corrected by model form uncertainty can also serve as a measure of closure relations' ability to capture the corresponding physical process in the given application domain.

There are also limitations of the procedure. Firstly, the numerical error introduced by discretizing the PDEs is not considered in this procedure. Rather, this part of the error is implicitly integrated into the model form uncertainty term. Secondly, additional uncertainty would be introduced by the statistical methods applied in the surrogate construction step, such as Gaussian process and dimension reduction. Such uncertainty varied through different statistical methods and is difficult to get an estimation for the whole validation domain. A more comprehensive work in future should include the verification work to take into account the numerical error and uncertainty, also with a more rigorous method to evaluate the uncertainty introduced by statistical methods.

CHAPTER.5 CASE STUDIES OF THE PROPOSED VALIDATION AND UNCERTAINTY QUANTIFICATION PROCEDURE

In this chapter, two case studies, the wall boiling heat transfer in subcooled boiling flow and the adiabatic bubbly flow, are performed as demonstrations of the VUQ procedure proposed in Chapter 4. The influential closure relation parameters for multiple quantities of interest (QoIs) are identified through two different global sensitivity analysis (GSA) methods: Morris screening and Sobol indices. The model form uncertainty and model parameter uncertainty of relevant closure relations are evaluated using the "modular Bayesian" approach. The uncertainties of QoIs are quantified by propagating the obtained uncertainties through the solver. The agreement between solver predicted QoIs and the experimental measurement are evaluated using two different validation metrics: confidence interval and area metric.

Strictly speaking, a complete VUQ for a MCFD solver requires taking all the QoIs from a scenario into consideration, quantifying the uncertainties of the solver predictions of all those QoIs and corresponding closure parameters, then evaluating the agreement between the solver predictions and the experimental measurement. This is compatible with the total data-model integration (TDMI) approach whose fundamental idea is to take all the available experimental data, the relevant closure models, and the solver into simultaneous consideration for the VUQ process. The TDMI approach serves as the basic idea of this proposed VUQ procedure and is achieved with the Bayesian method. Thus such VUQ requirement is compatible with the procedure. On the other hand, the complete VUQ requires the support of the measurements for all the QoIs in a given scenario. This includes measurements of the phasic velocities, the void fraction, the pressure drop, the temperature, and the wall heat transfer, etc. It is impractical to measure all the phenomena in a single experimental facility. Moreover, the lack of a satisfactory scaling method for two phase flow system prohibits the use of experimental data from different facilities into one complete VUQ process of the MCFD solver.

In this work, the limitation of available experimental data is considered, and a more practical approach for the VUQ of the MCFD solver is applied. The VUQ of MCFD is decomposed into two separate case studies. One focuses on the wall heat transfer in the

scenario of subcooled flow boiling in a vertical channel, the other focuses on the flow dynamics in the scenario of adiabatic bubbly flow. The decomposition is depicted in Figure 20.

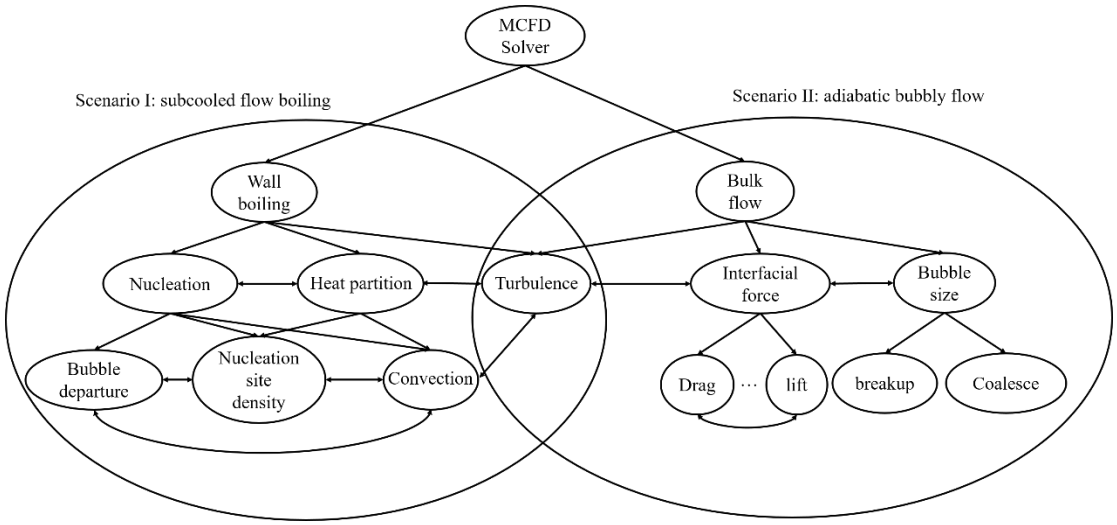

Figure 20. Decomposition of VUQ case studies.

An MCFD solver developed based on open source CFD toolbox OpenFOAM [91] is evaluated in the following work, the UQLab package [92] is used for sensitivity analysis.

5.1 Case Study I: VUQ on wall boiling heat transfer

5.1.1 Solver evaluation and data collection

The scenario investigated in this case study is the subcooled flow boiling, with a focus on the wall heat transfer behavior. In this case study, the wall superheat and the three heat transfer components, i.e. evaporation heat transfer, quenching heat transfer, and forced convective heat transfer, are chosen to be the QoIs. Although the boiling process is crucial for a two phase flow system, the traditional experiments can only measure the wall superheat through thermocouples, as in [93]. Hence the previous validation efforts for the boiling related scenario can only be performed for the wall superheat [3].

Recently, the experiments using IR camera makes it possible to measure the detailed wall temperature development in boiling process, thus making the derivation of wall heat transfer components possible [94-97].

In this work, the experimental data is collected from [27], which measures the wall superheat and three heat transfer components in an upward subcooled boiling flow. The QoIs over 12 different input heat fluxes are extracted from the report. Since the data provided in the report are averaged on the whole 10mm-by-10mm heat surface, the QoIs obtained from MCFD simulations in this work are also averaged in the same manner to match the data.

The closure relation that is pertinent to the boiling process is the wall boiling model which consists of the heat partition and the nucleation related empirical correlations. Thus the wall boiling closure relations are the focus of this scenario. It is also assumed that other closure relations including the interfacial forces have a minor impact on the wall heat transfer behavior and thus do not need to be considered in the UQ process in this case study. This assumption is based on the observation of the chosen boiling closure relations studied in this case, which relies only on one flow feature: the $\boldsymbol{y}^+$ in near wall cell. The $\boldsymbol{y}^+$ value is obtained along with the mesh study and is assumed to be independent from the wall boiling closure relations. The evaluation process can be summarized in Table 6.

Table 6. Summary of evaluations for wall boiling heat transfer scenario

Scenario	QoIs	Varying input Condition	Closure relations studied
Subcooled flow boiling	$T_{sup}, q_{Ev}, q_{Qu}, q_{Fc}$	Applied wall heat flux: 500 ~ 2500 kW/m^2	Nucleation site density, Bubble departure diameter, Bubble departure frequency, Effective bubble area, Bubble growth waiting time, Convective heat transfer

The wall boiling closure relations studied in this case are based on the heat partitioning model proposed by [65]. The heat is partitioned into these three components, and each component is support by one to several empirical closure relations.

$$q_{Wall} = q_{Ev} + q_{Qu} + q_{Fc} \,. \tag{69}$$

The evaporation heat transfer is dependent on three nucleation correlations: the active nucleation site, the bubble departure diameter and bubble departure frequency:

$$q_{Ev} = \frac{\pi}{6} D_d^3 \rho_v f_a N_a h_{fg}. \tag{70}$$

The nucleation site density model studied in this work is proposed by [71]

$$N_a = N_{avg}\left[1 - exp\left(-\frac{\theta^2}{8\mu_{con}^2}\right)\right]\left[exp\left(\frac{\lambda' g(\rho^+)}{R_c}\right) - 1\right], \tag{71}$$

where

$$R_c = \frac{2\sigma\left\{1 + \left(\rho_g/\rho_f\right)\right\}/P_f}{\exp\left\{h_{fg}(T_g - T_{sat})/R\,T_g T_{sat}\right\} - 1} \tag{72}$$

$$f(\rho^+) = -0.01064 + 0.48246\rho^+ - 0.22712\rho^{+^2} + 0.05468\rho^{+^3} \tag{73}$$

$$\rho^+ = \log\left(\frac{\rho_l - \rho_g}{\rho_g}\right). \tag{74}$$

Here N_{avg}, μ_{con} and λ' are empirical parameters that represent the average cavity density, angle scaler, and cavity radius scaler respectively.

The bubble departure diameter model studied in this work is proposed by [74]:

$$D_d = d_1\theta\left(\frac{\sigma}{g\Delta\rho}\right)^{0.5}\left(\frac{\Delta\rho}{\rho_g}\right)^{0.9}. \tag{75}$$

The bubble frequency model studied in this work is proposed by [76]

$$f_d = \sqrt{\frac{4g(\rho_l - \rho_g)}{3D_d\rho_l}}. \tag{76}$$

The Quenching heat transfer is based on analytical analysis [66], yet several terms in the expression are still depend on empirical parameters:

$$q_{qu} = A_b \frac{2}{\sqrt{\pi}} f_d \sqrt{t_{wait} \lambda_l \rho_l c_{p,l}} \left(T_{sup} - T_l\right). \tag{77}$$

A_b in the expression is the effective bubble area fraction,

$$A_b = \max\left(\pi \left(a \frac{D_d}{2}\right)^2 N_a, 1.0\right), \tag{78}$$

and a is the bubble influence factor which is an empirical parameter. Moreover, t_{wait} is the waiting time between the bubble departure and the appearance of a new bubble at a given nucleation site. In this work, the model proposed by [77] is selected:

$$t_{wait} = \frac{e}{f_d}, \tag{79}$$

where e is the waiting time coefficient, the suggested value is 0.8.

The forced convective heat transfer can be expressed as

$$q_{Fc} = (1 - A_b) h_l (T_{sup} - T_l), \tag{80}$$

where h_l is the heat transfer coefficient, which can be evaluated through the analysis of heat transfer in turbulent boundary layer flow. The wall function of turbulence model is applied which relates h_l to the turbulent Prandtl numbers amd the dimensionless near-wall flow velocity. In this work, a simplified version of the correlation proposed by [98] is adopted as

$$h_l = u_\tau \rho c_p \left[\mathrm{Pr}_t \frac{1}{\kappa} \ln(E y^+) + P\right]^{-1}, \tag{81}$$

where E and P are the empirical parameters from the wall function.

As noted from the empirical correlations discussed previously, the number of the empirical parameters is quite large but not all of them have clear physical meanings. Based on this, a preliminary parameter selection is processed to choose those with clear physical meanings for further UQ process. The prior uncertainties of those selected empirical parameters are determined based on expert judgment. The results are summarized in Table 7.

Table 7. Prior uncertainties of studied empirical parameters in wall boiling closure relations

Empirical parameter	Physical meaning	Nominal value	Lower bound	Upper bound
N_{avg}	Averaged Nucleation site density coefficient	4.72×10^5	4.72×10^4	4.72×10^6
μ_{con}	Contact angle scaler	0.722	0.4	3.14
d_1	Departure diameter constant	0.0015	0.0005	0.003
a	Effective bubble area factor	1	0.5	2
e	Bubble growth waiting time factor	0.8	0.5	0.95
E	Wall function Log law offset	9.79	1	15
P	Wall function coefficient	0	-9	9

5.1.2 Surrogate construction

As noted for the aforementioned closure relations, the wall boiling closure relations can be regarded as source terms that only loosely coupled with the conservative equations. Thus, a simplified physical based surrogate model is constructed for this case study. The structure of the simplified model is illustrated in Figure 21.

It can be found that the wall boiling closure relation can be regarded as a network of different correlations, which combine to form a non-linear equation with the constraint of a fixed and known value: the total wall heat flux. Following this structure, the wall superheat can be calculated by solving a non-linear equation using the Newton-Raphson method. Once the wall superheat is obtained, the heat transfer components can be calculated respectively.

For this simplified model, only one feature is required from the simulation of MCFD solver: the y^+ of the near wall cell. For this case, the y^+ is obtained and prescribed through

the mesh study, which is performed before any cases are simulated in the solver. Since the wall function of the turbulence model is included for all the simulation in this work, the near wall y^+ should be greater than 30. A uniform mesh is chosen to meet this criterion through the mesh study. Based on the mesh setup, the y^+ is obtained and assumed to be independent of parameters of wall boiling closure relations.

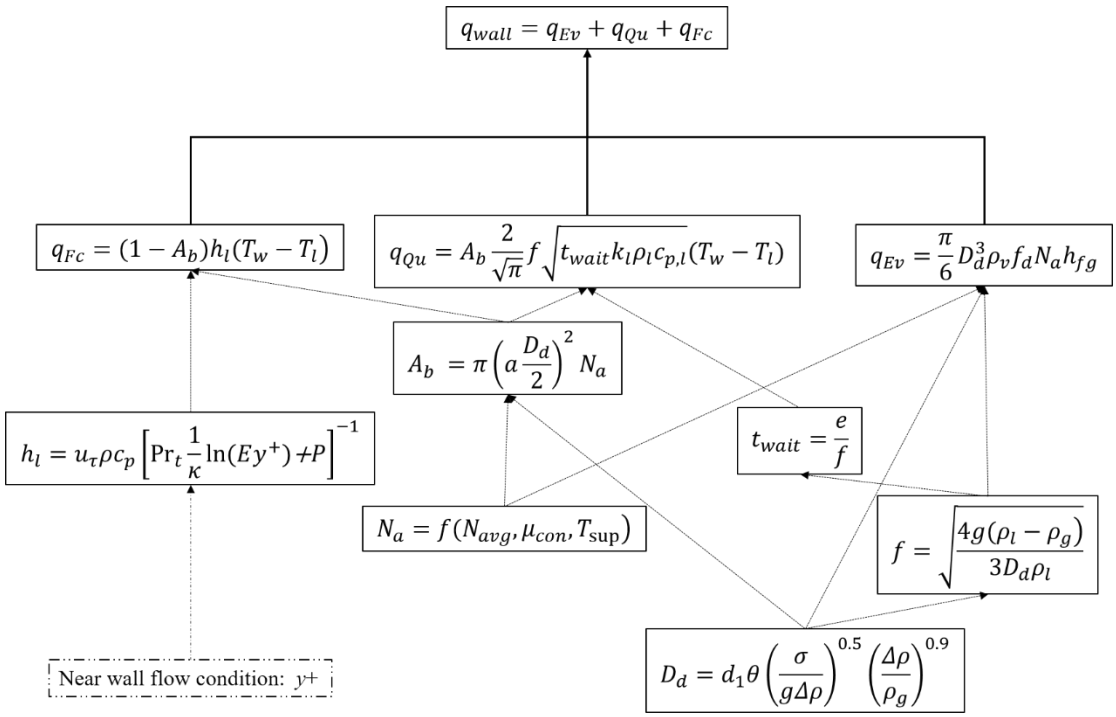

Figure 21. Structure of wall boiling closures in this case study.

5.1.3 Sensitivity analysis

In this step, two methods of global sensitivity analysis (GSA) are performed. As previously discussed, the QoIs are the averaged quantities over the heating surface. In this step, those obtained QoIs are further integrated over the whole simulation domain to generate global responses quantity for GSA. The results of Morris Screening method are plotted in Figure 22.

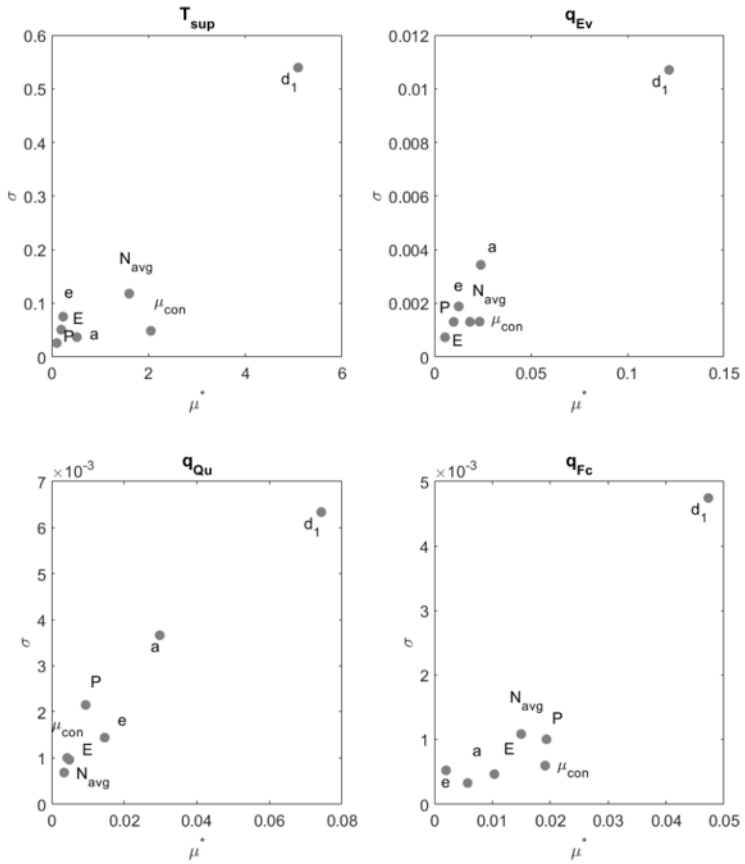

Figure 22. Morris screening measures for wall boiling empirical parameters.

It can be observed that for all the QoIs, the bubble departure diameter constant $d_1$is the most influential parameter. This is reasonable considering the fact that the evaporation heat transfer component is dependent on the third power of d_1, while for other parameters the relationship is linear. For other parameters, the importance varies by different QoIs. For example, the nucleation site density constant N_{avg} has relatively strong influence on wall superheat, whereas the contact angle scaler μ_{con} has influence on wall superheat and convective heat transfer component. Moreover, the bubble effect area factor a plays a relatively important role on evaporation and quenching heat transfer components.

Morris screening is an efficient measurement method for GSA, but it cannot generate a quantitative measurement of the parameters sensitivity on the QoIs. The Sobol indices method, on the other hand, can generate indices that quantitatively measure the sensitivity of parameters on the QoIs, while is much more computationally expensive. The results of Sobol indices for the four QoIs are plotted in Figure 23.

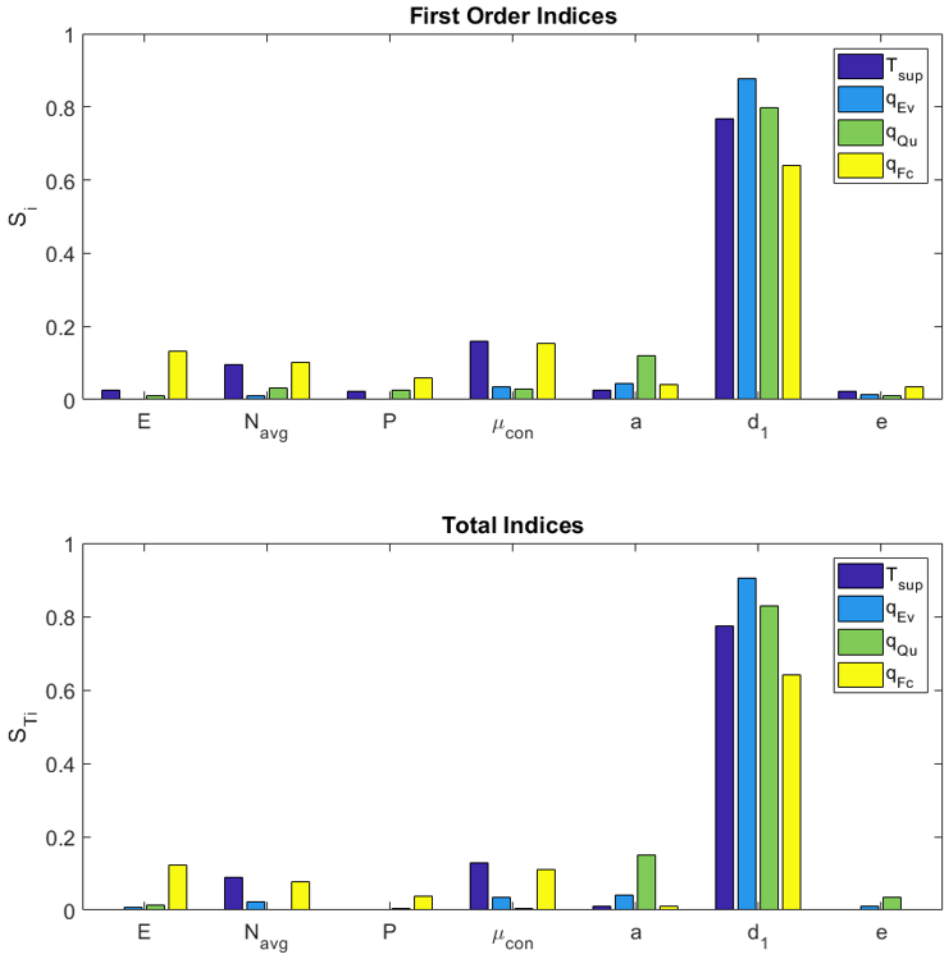

Figure 23. Sobol indices for wall boiling empirical parameters.

It is noted in Figure 23 that the bubble departure diameter constant d_1 is the dominant parameter for all QoIs, the results of other parameters are also consistent with the Morris Screening method. Thus we can conclude that the GSA results from those two methods are consistent.

5.1.4 Parameter selection

As noted from GSA results, the most dominant parameter for all QoIs is d_1. However, all other parameters are still influential to at least one QoI. There is no clear clue to rule out anyone of them. Moreover, preliminary MCMC sampling suggest that the parameter identifiability issue exists when inferring all the parameters together. Thus, the parameter selection algorithm proposed by [87] is applied. Firstly, the rank of the sensitivity matrix is computed using singular value decomposition (SVD). The rank of the

sensitivity matrix is found to be 4, which equals to the maximum number of identifiable parameters for the given closure relations with nominal parameter values. Then the selection score for all the possible combinations of parameters is calculated based on the proposed algorithm. This procedure is performed for all QoIs respectively. Some results are summarized in Table 8.

Table 8. Optimal selection scores for different QoIs

Parameter combination	Selection score for T_{sup}	Selection score for q_{Ev}	Selection score for q_{Qu}	Selection score for q_{Fc}
(a,d_1,e,E)	103.36	1.446E+06	**3.047E+06**	1.398E+07
(a,μ_{con},e,P)	133.76	1.823E+06	8.679E+06	**4.772E+06**
(a,μ_{con},d_1,e)	**90.55**	**6.518E+05**	4.630E+06	6.548E+06

It can be found from the results that for different QoI, the parameter selection with minimum selection score is different, in this sense, trade-off needs to be made between different QoIs. In a preliminary MCMC, single-value dependence between μ_{con} and d_1 is observed, which is undesired for the Bayesian inference. Thus in this work, the parameter combination (a,d_1,e,E) is selected for the Bayesian inference. A more robust selection criterion for multiple QoIs is desired for future work.

5.1.5 Uncertainty quantification

In the step of uncertainty quantification, the modular Bayesian is applied, which firstly evaluate the model form uncertainty using Gaussian process while fix the parameters at their nominal values. In this case, the full datasets are divided into three parts: 6 datasets are used to evaluate the model form uncertainty, 5 datasets are used for model parameter uncertainty inference, while 1 dataset is left for testing. The decomposition is summarized in Table 9.

Table 9. Boiling datasets decomposition for different purposes

Datasets decomposition	Input heat flux (kW/m^2)
Model form uncertainty evaluation datasets	500, 1000, 1500, 2200, 2400, 2450
Parameter uncertainty evaluation datasets	750, 1250, 1750, 2100, 2300
Testing dataset	2000

In this work, the QoIs measured over the surface are averaged, thus the model form uncertainty term is spatially independent and can be expressed by a single variant GP: $\delta(q_{wall}) \sim GP(q_{wall})$. Six datasets are used for evaluating the model form uncertainty. The exponential kernel is selected as the kernel function of GP. The hyperparameters of the kernel are evaluated using the MAP method. The obtained results are plotted in Figure 24.

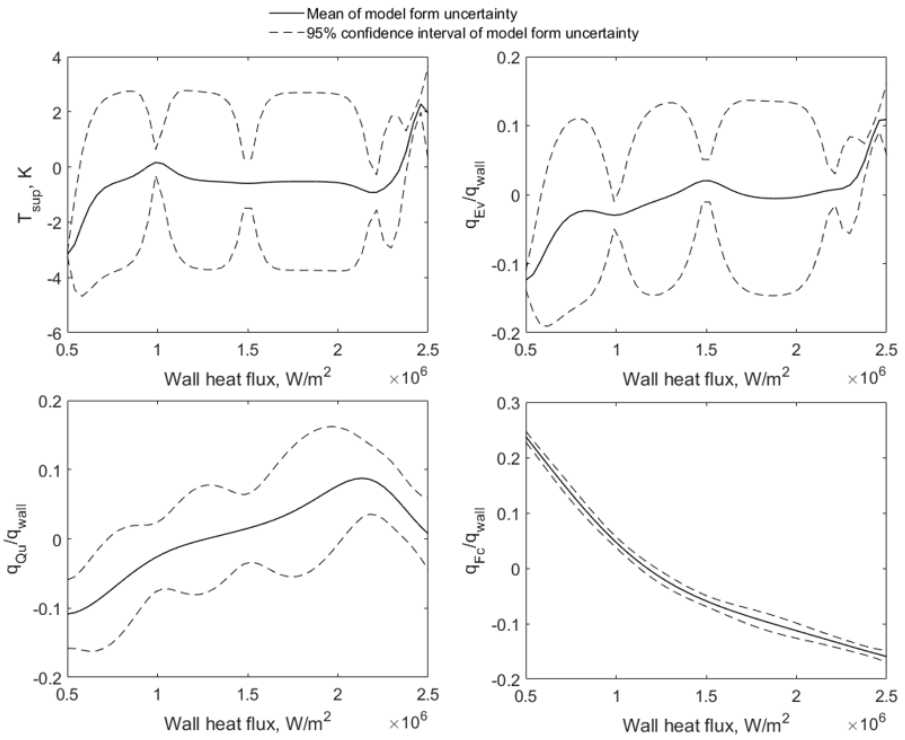

Figure 24. Model form uncertainty $\delta(q_{wall})$ for different QoIs.

It can be found from Figure 24 that the T_{sup} has small model form uncertainties, with exceptions under high and low heat flux conditions. The other QoIs have significant discrepancy between model predictions and experimental measurements. To some extent, this is expected considering the fact that when the wall boiling closure relations were proposed, the only measurable data was the wall superheat. Thus, the wall boiling closure relations have already been compared against many different wall superheat measurements, and can give wall superheat predictions with reasonable accuracy. On the other hand, the heat partitioning prediction lacks comparison due to the lack of experimental measurements, thus the existent significant model form uncertainty cannot be identified if only takes the wall superheat into consideration. In this practice, the model form uncertainties can be identified under the TMDI approach by "learning" from the multiphysics measurement.

Once the parameters for inverse UQ are selected, the MCMC method is applied for Bayesian inference. The MCMC has been applied to similar applications such as the inverse UQ on turbulence model [99] and fluidized-bed gasifier simulations [100]. The Delayed Rejection Adaptive Metropolis (DRAM) algorithm proposed by [90] is applied. There are two features of DRAM, one is delayed rejection, which means if a candidate is rejected in the sampling process, an alternate candidate is constructed to induce greater mixing. The other is adaption, which means the covariance matrix of the parameters is continuously updated using the accepted candidates.

The purpose of the MCMC is to construct stationary distribution of a Markov chain that equals to the posterior distribution of the parameter. In practice, the first 5000 samples from the chain are disregarded to ensure the convergence (or so-called "burn-in") of the following chains. Moreover, for the "burned-in" chain, only every 10^{th} elements are kept in order to reduce the auto-correlation of the chain as a requirement of the stationary distribution.

The processed sample chain for all selected parameters, and their autocorrelations are plotted in Figure 25. Good mixing and the fast decay of auto-correlations for all parameters can be observed which indicate the process chain can be regarded as the stationary distributions of the Markov chains.

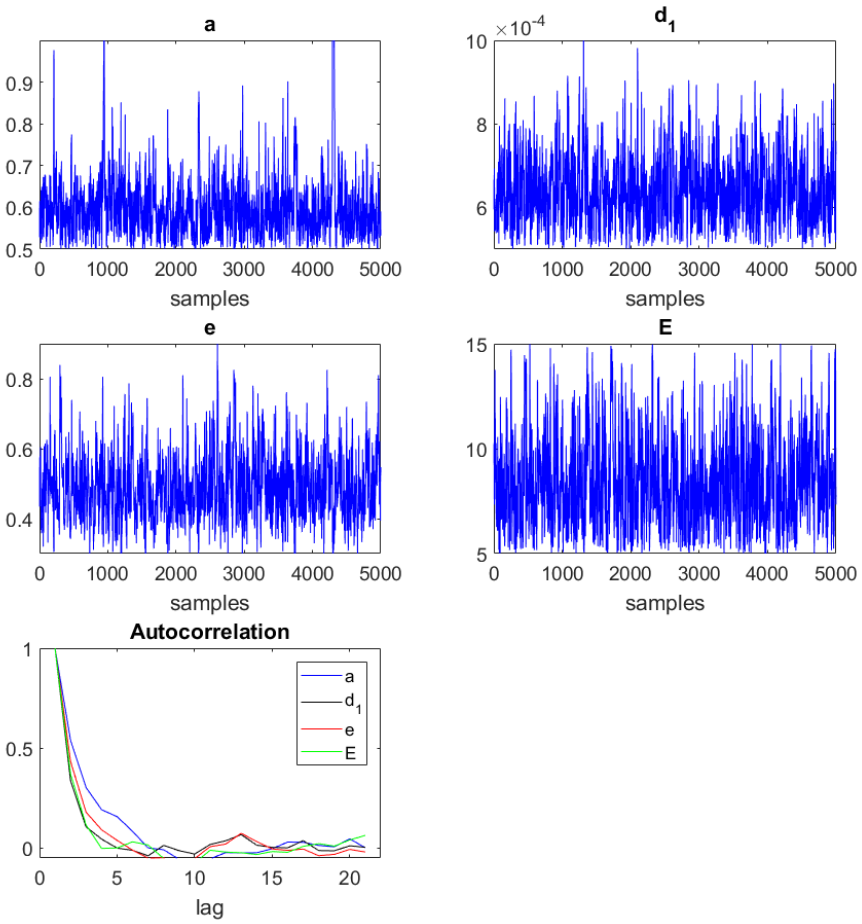

Figure 25. MCMC sample traces and auto-correlations of wall boiling closure relation parameters.

The marginal and pair-wise joint distributions of the four parameters are plotted in Figure 26. Light correlation between d_1 and e is observed, while they come from different closure relations. This suggests that with the TDMI approach, which takes all closure relations and all QoIs of the solver into consideration simultaneously, the potential interaction between different phenomena can be identified. The statistics of the parameter distribution are summarized in Table 10.

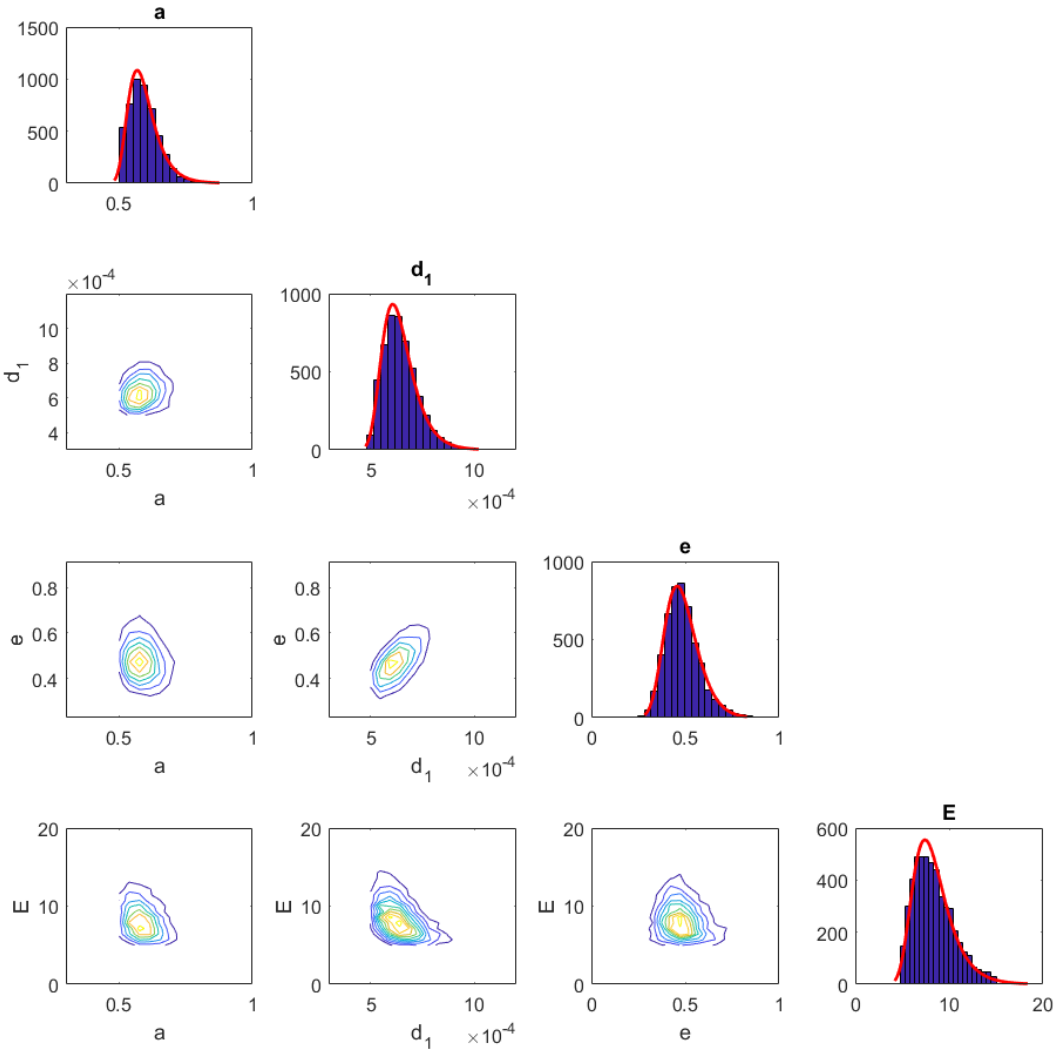

Figure 26. Marginal and pair-wise joint distributions of selected empirical parameters.

Table 10.posterior statistics of selected empirical parameters

Parameter	Mean	Standard deviation	95% Confidence interval
a	0.5943	0.0743	[0.4523, 0.7445]
d_1	6.42×10^{-4}	7.98×10^{-5}	$[4.85 \times 10^{-4}, 7.98 \times 10^{-4}]$
e	0.5135	0.0916	[0.3520, 0.7110]
E	8.3839	2.1374	[4.2580, 12.6366]

Once the posterior distributions of the parameters are obtained, the uncertainties of QoIs can be estimated by propagating the posterior distributions through the solver, the

previously obtained model form uncertainty can be added to the obtained QoIs as a correction term. The obtained results are displayed in Figure 27.

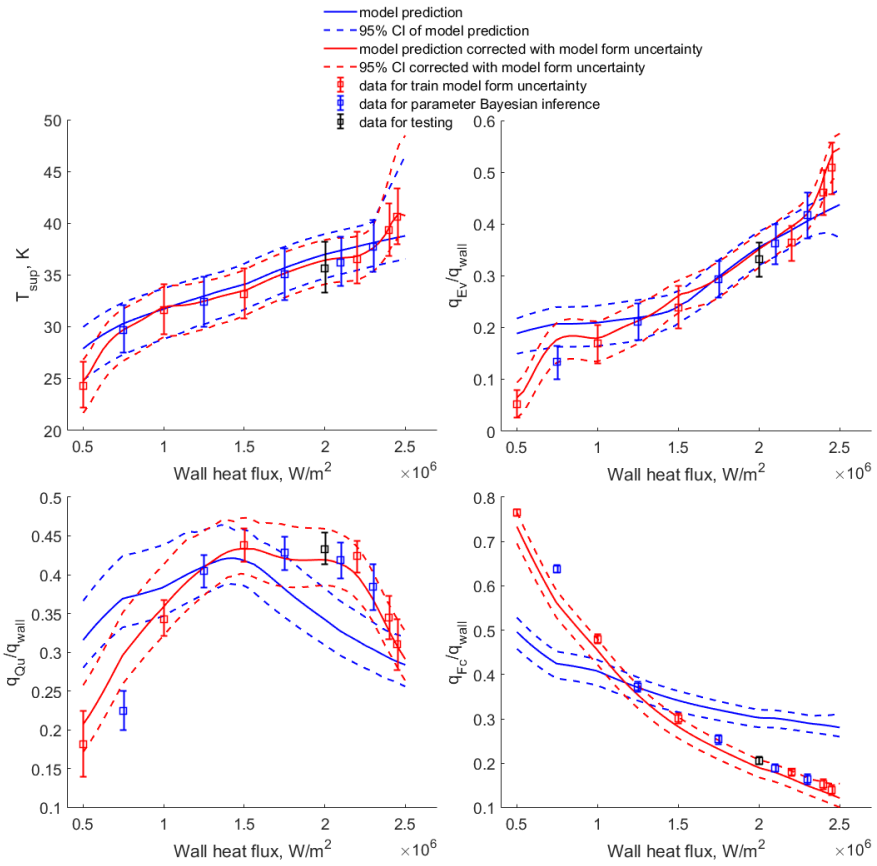

Figure 27. 95% confidence intervals (CIs) for different QoIs from wall boiling closure relations.

It can be observed from Figure 27 that the predictions of T_{sub} and q_{Ev} are generally in good agreement with the whole dataset. However, q_{Qu} is overestimated in low heat flux region while is underestimated in high heat flux region, and q_{Fc} demonstrates the opposite trend. Moreover, for all QoIs, if takes the model form uncertainty into consideration, the model prediction will be in good agreement with the experimental measurement for all the input conditions. Moreover, since the model form uncertainty is constructed using Gaussian Process (GP) which treat the input condition as variable, it can be interpolated to any unmeasured wall heat flux. In this sense, the applicable range of the UQ results is 500 kW/m^2 ~ 2500 kW/m^2, which basically covers normal nucleate boiling regime.

5.1.6 Validation metrics

In this work, two different types of validation metrics are calculated: the confidence interval [41] and the area metric [43]. The confidence intervals for the QoIs from wall boiling closure relations are plotted in Figure 28.

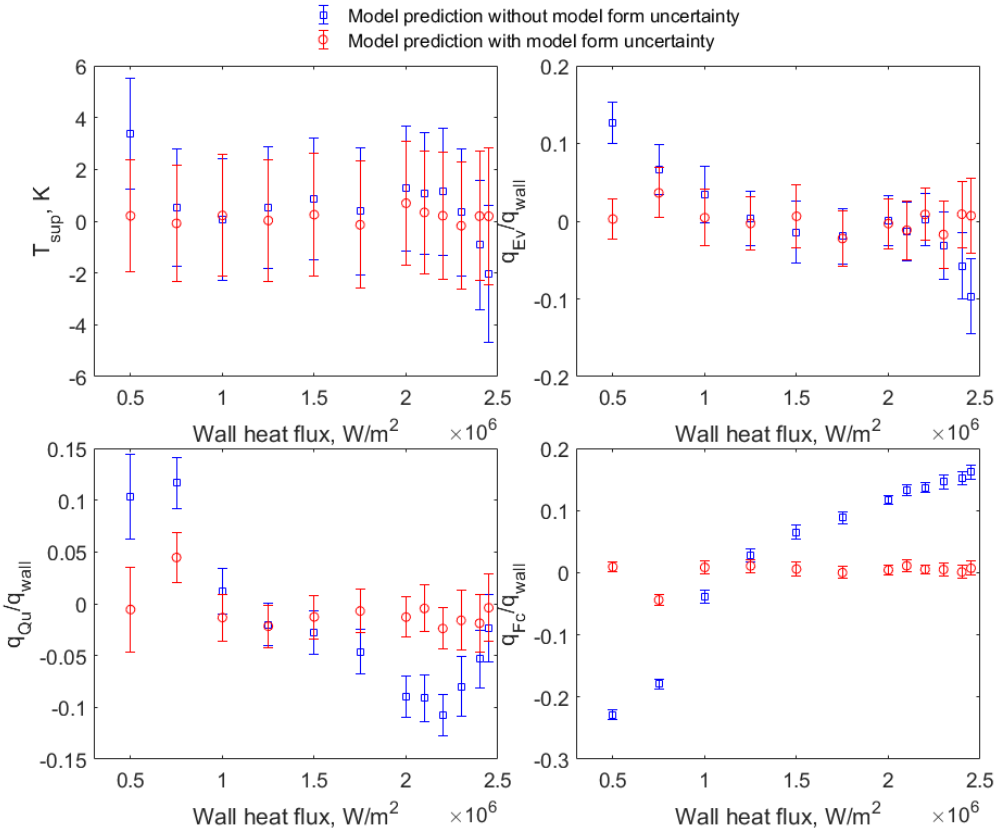

Figure 28. Confidence intervals for different QoIs from wall boiling closure relations.

If only consider the pure model prediction, it can be found the confidence intervals of T_{sup} are close to or covers zero, with exceptions in 500kW/m^2 and 2500kW/m^2. This suggests that the model prediction of T_{sup} is in good agreement with the experimental measurement. For other three QoIs, most of the confidence intervals deviate from zero. This indicates that the true error between model predictions and data are significant. Taking the q_{Fc} as an example, it can be observed that the wall boiling closure relation significantly underestimates the convective heat transfer for low heat flux cases while overestimating it for high heat fluxes. On the other hand, it can be observed that with the consideration of model form uncertainty, the errors are significantly reduced for almost all the cases. Such

improvement indicates that the wall boiling closure relations has intrinsic model form uncertainty, and such model form uncertainty can be successfully evaluated through the proposed “Modular Bayesian” approach.

The advantage of confidence interval is it has very clear physical interpretation while also easy to implement. For example, the 95% confidence interval can be interpreted as we are 95% confident that the true error of QoI predicted by the solver is within the given interval. Such interpretation can be directly applied in the design and safety analysis for a certain engineering problem. One disadvantage of confidence interval, however, is only the mean of the QoIs predicted by solver is considered and the interval contains only statistics from the uncertainty of experimental data. In other words, the confidence interval fails to consider the full uncertainty of the solver predictions and the experimental data.

The area metric, on the other hand, compensates for the disadvantage of confidence interval by comparing the discrepancy between the cumulative distribution functions (CDFs) of the experimental data and the solver prediction. The results of area metric are plotted in Figure 29. It should be noted that every single point displayed in the figure represent the integrated area between the two CDFs mentioned above, a large value of area metric suggests large difference between the CDFs of model prediction and experimental data.

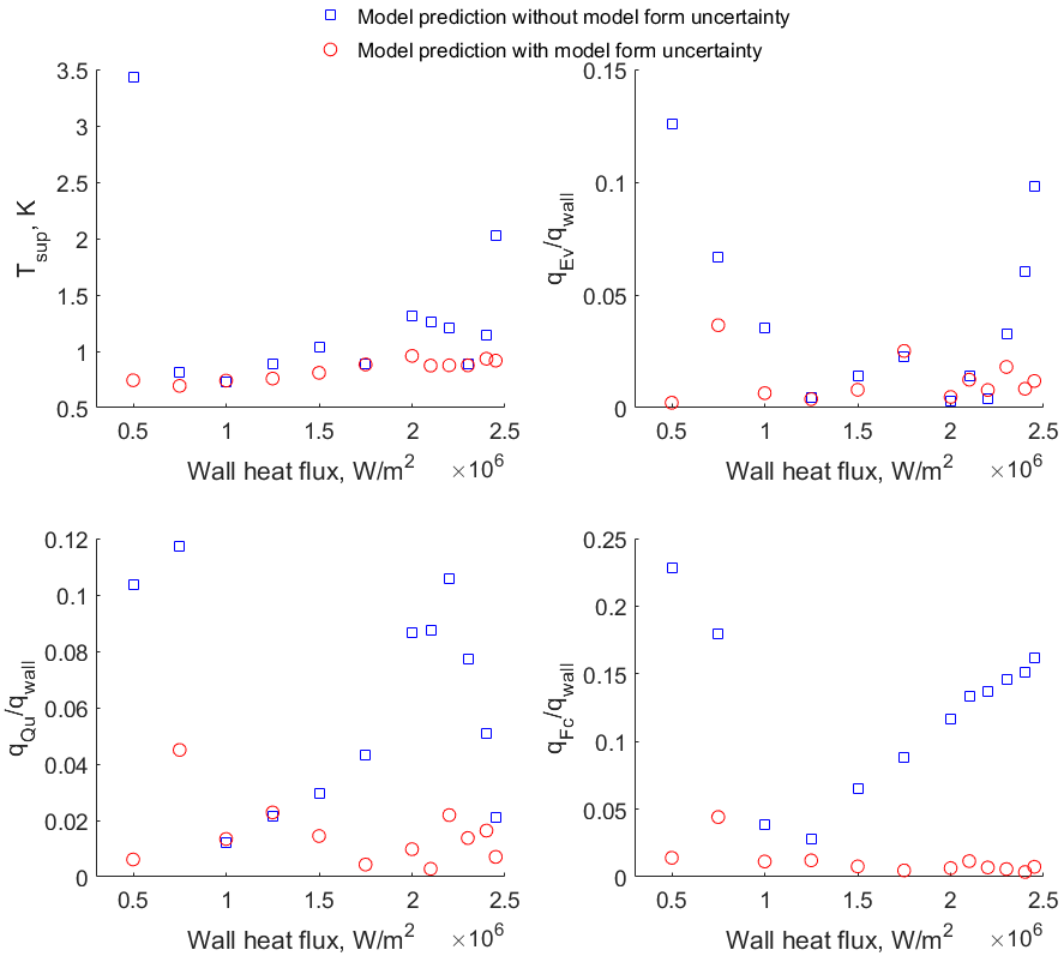

Figure 29. Area metrics for different QoIs from wall boiling closure relations.

It can be found that for the pure model prediction, T_{sup} has small area metrics, with exceptions in 500kW/m^2 and 2500kW/m^2. For the other three QoIs, the pure model predictions have significant difference with experimental data. With model form uncertainty is accounted for, the results are significantly improved. In this sense, the two validation metrics are consistent.

Another advantage of area metric is that it has several properties as suggested by [42] that make it mathematically well behaved and well understood. Such properties can be expressed as following for CFDs of two random variables X and Y:

- Non-negativity: $d(X,Y) \geq 0$
- Symmetry: $d(X,Y) = d(Y,X)$
- Triangle inequality: $d(X,Y) + d(Y,Z) \geq d(X,Z)$

- Identity of indiscernible: $d(X,Y) = 0$ if and only if $X = Y$

On the other hand, there are also two disadvantages of the area metric. One is that the area metric is much more complicated to calculate compared with confidence interval. It also puts higher requirement for the data quality, requires not just the statistics of the data such as its mean and variance, but the full distribution. This usually requires measuring the same quantities multiple times. The other disadvantage is that the area metric measures the absolute difference between the solver predictions and the data. Thus it cannot discern whether the solver is overestimate or underestimate the QoIs. Thus, it is suggested to calculate both validation metrics for a comprehensive evaluation of the agreement between the solver prediction and experimental data.

5.2 Case study II: VUQ on flow dynamics

5.2.1 Solver evaluation and data collection

The MCFD's performance on flow dynamics in the context of upward adiabatic bubbly flow is studied in this case. The reason to choose this scenario is the relatively rich experimental data. Several experimental investigations can be found from literature, which includes a series measurement of the phasic velocity, the void fraction, and the bubble dynamics. Representative works include [101, 102], and many more summarized in [103].

A simplified case is considered which focuses on only two QoIs: the void fraction and gas velocity. Based on the idea of TDMI, the experimental data from different conditions are taken into consideration simultaneously. Thus, the experiments conducted by [104] are chosen as the datasets, from which eight datasets of different inlet conditions are extracted. The summary of evaluations is summarized in Table 11.

. For the VUQ process, the uncertainty of data is important. However, the detail uncertainty analysis is not included in the original literature. In this work, the conservative estimation based on an incomplete error analysis from the original literature is adopted, the uncertainty for all QoIs is assumed to be 10%.

Table 11. Summary of evaluations for interfacial momentum closure relations

Scenario	QoIs	Varying input Condition	Closure relation studied
Adiabatic bubbly flow	spatial distributions of α, U_1	j_g from 0.09 – 0.48 m/s, j_l from 0.64 – 2.0 m/s	drag force, lift force, turbulent dispersion force, virtual mass force, wall lubrication force

The interfacial force coefficients are investigated. For drag and lift force, there are many semi-mechanistic correlations for the coefficients. In this work, a simplified version is used which assumes those coefficients are independent with the flow condition and can be expressed by constant values. Besides the interfacial forces, the bubble size also has a significant influence on the bubbly flow simulation. The state of the art method for bubble size prediction is to solve the interfacial area concertation equation along with the conservation equations. This approach, however, includes more parameters and requires many more extra sampling runs in the MCFD solver. Considering the limitation of computational resources, in this first demonstration case, a simplified assumption for bubble size was made. The bubble size is set to be a constant whose value equals to the inlet bubble size. Thus, only the interfacial momentum closure relations are studied and the model selected is summarized in Table 12. The prior uncertainties of the interfacial force coefficients are summarized in Table 13. The uniform uncertainties are assumed for those parameters, as the same to the Case Study I.

Table 12. Summary of the studied interfacial momentum closure relations

Force type	Expression	Parameter studied
Drag force	$\mathbf{M}_g^D = -\frac{3}{4}\frac{C_d}{D_s}\rho_l\alpha\|\mathbf{U}_g - \mathbf{U}_l\|(\mathbf{U}_g - \mathbf{U}_l)$	C_d
Lift force	$\mathbf{M}_g^L = C_l\rho_l\alpha(\mathbf{U}_g - \mathbf{U}_l) \times (\nabla \times \mathbf{U}_g)$	C_l
Wall lubrication force [55]	$\mathbf{M}_g^{WL} = -f_{WL}(C_{wl}, y_w)\alpha\rho_l\frac{\|\mathbf{U}_r - (\mathbf{U}_r \cdot \mathbf{n}_w)\mathbf{n}_w\|^2}{D_s}\mathbf{n}_w$, $f_{WL}(C_{wl}, y_w) = \max\left(-0.2C_{wl} + (\frac{C_{wl}}{y_w})D_s, 0\right)$	C_{wl}
Turbulent dispersion force[56]	$\mathbf{M}_g^{TD} = -\frac{3}{4}\frac{C_D}{D_s}\frac{v_l^t}{\sigma^t Pr_l^t}\rho_l\|\mathbf{U}_g - \mathbf{U}_l\|\nabla\alpha$	σ^t
Virtual mass force[57]	$\mathbf{M}_g^{VM} = -C_{vm}\rho_l\alpha\left(\frac{D\mathbf{U}_g}{Dt} - \frac{D\mathbf{U}_l}{Dt}\right)$	C_{vm}

Table 13. Prior uncertainties of parameters in interfacial momentum closure relations

empirical parameter	Nominal value	Lower bound	Upper bound
C_d	0.6	0.44	1.0
C_l	0.03	-0.05	0.1
C_{wl}	0.05	0.03	0.08
C_{vm}	0.5	0.3	1.0
σ^t	0.9	0.6	1.2

To match the experimental measurement, the studied QoIs are set to be the distributions of void fraction and the gas velocity along the radial direction of the tube at (z/D = 62).

5.2.2 Surrogate construction

The surrogate model is constructed using Gaussian Process. There are eight datasets with different liquid and gas superficial velocities. Each case contains detailed 13 radial distributions of void fraction and gas velocity. Thus, the output QoIs can form a high dimensional vector. The length of the output QoIs is $8 \times 13 \times 2 = 208$. It is very inefficient to construct 208 separate GPs for the QoIs, thus the dimension reduction based on PCA is performed for this case.

Firstly, the MCFD simulations are performed with perturbed interfacial force coefficients sampling from Latin hypercube sampling. A total 64 samples are generated and run in the MCFD solver, among which 56 samples are used for surrogate construction and 8 samples are used for cross-validation.

Considering the high dimensionality of the outputs, the principal component analysis (PCA) applied for dimensionality reduction. The void fraction and gas velocity simulation results are centered and scaled according to the inlet conditions, then stacked to create a matrix to which PCA is applied.

The results of the PCA applied to the combined void fraction and fluid temperature predictions are shown in Figure 30. It can be observed that up to 99.5% of the total variances can be explained by the first 8 principal components (PCs).

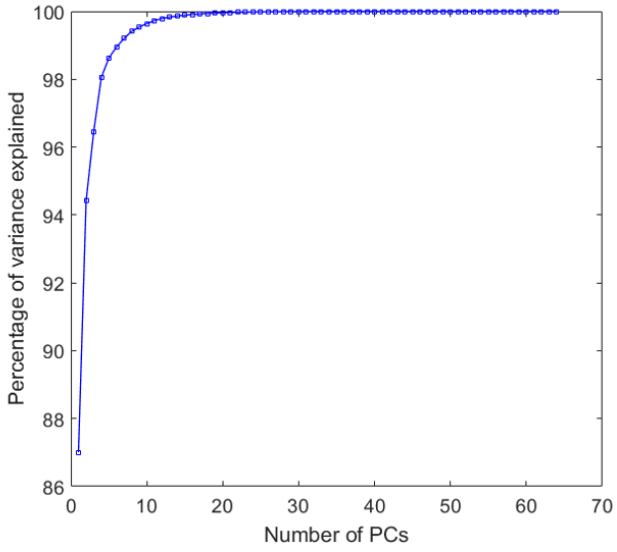

Figure 30. Accumulative percentage of variances explained by PCs.

Based on the PCA results, the first 8 PCs are selected for constructing surrogate, this reducing the dimension of outputs from 208 to 8. The variations of the 8 PCs for one condition are summarized in Figure 31.

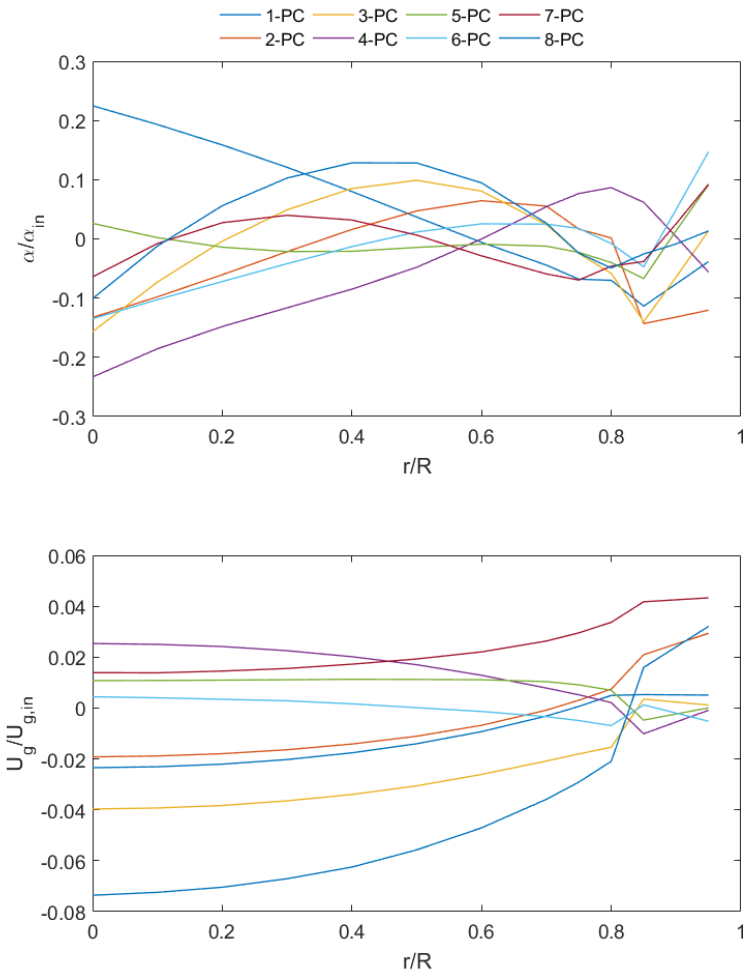

Figure 31. Variations of two QoIs captured by 8 PCs (j_g = 0.29 m/s, j_l = 1.1 m/s).

The GP surrogate is constructed for the 8 PCs, respectively. The constructed GP surrogate is an approximation of the original MCFD solver, whose accuracy should be evaluated. In this work, the cross validation is performed for the accuracy assessment. The procedure of cross-validation is done through the following steps:

- Randomly divide the sampled test case results into *k* groups of the same size.
- Construct the surrogate with *k*-1 groups of results and left 1 group as validation set.
- Repeat the previous step *k* times, in each time a different group is treated as a validation set.

- Average the obtained results from the *k* evaluations.

The comparisons of PC scores between MCFD simulations and GP predictions are plotted in Figure 32 , where an accurate GP prediction should fall into or very close to the diagonal line in the plot. It can be found from the figure that the GP surrogate predictions are in good agreement with original MCFD simulations for most of the PC scores. Except for 3rd and 7th score which have some moderate discrepancies.

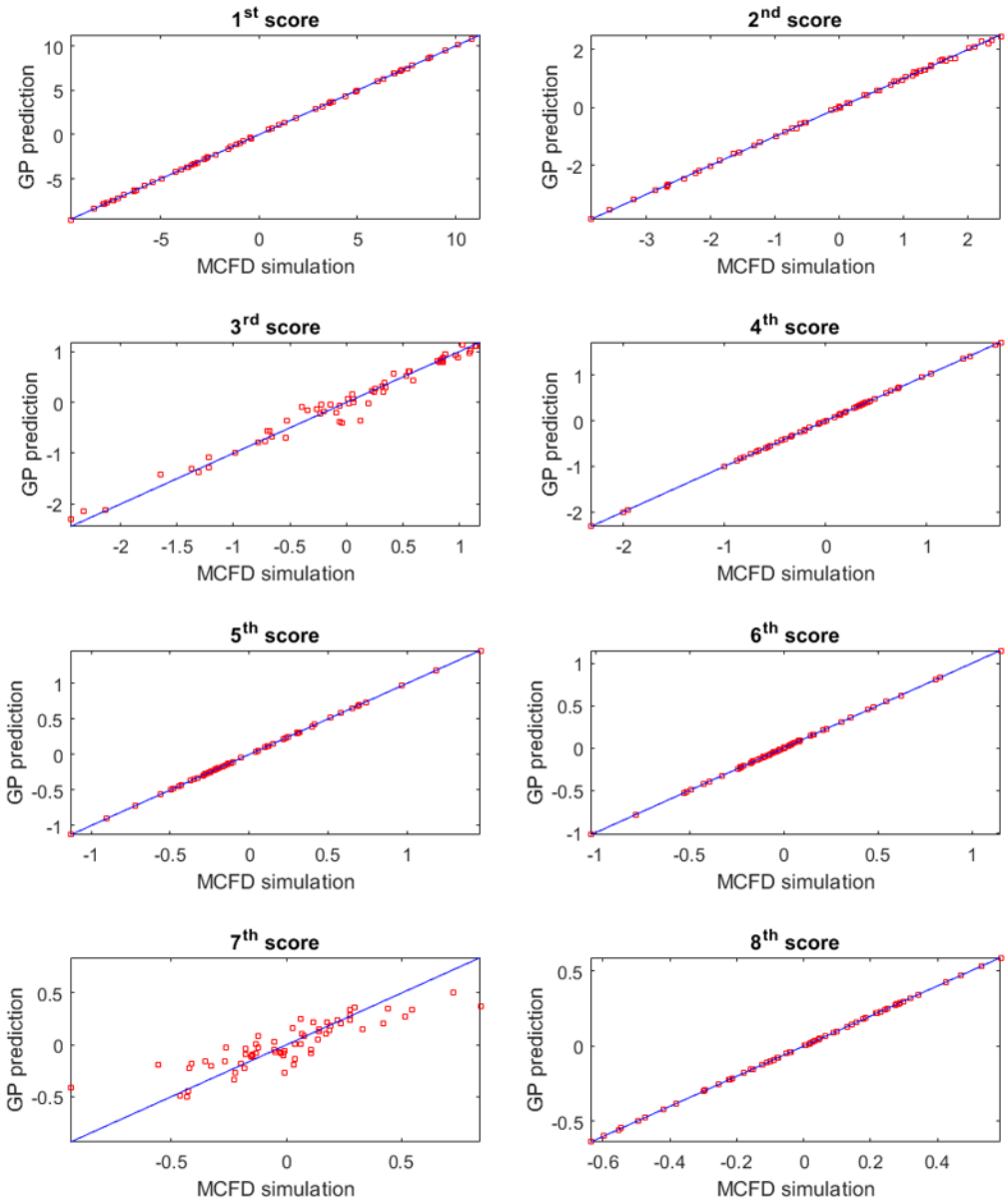

Figure 32. Comparison of PC scores between MCFD simulations and GP predictions.

The averaged accuracy of surrogate can be estimated with the mean square error, the results for all the PCs are summarized in Table 14.

$$\mathrm{MSE} = \frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i^M)^2 \tag{82}$$

Table 14. Cross validation results

PC scores	MSE
1st score	6.8542e-4
2nd score	0.0012
3rd score	0.0169
4th score	6.4328e-9
5th score	3.7789e-09
6th score	1.2514e-9
7th score	0.0240
8th score	3.0624e-09

The results confirm the qualitative observation from Figure 32, that the 3rd and 7th PC score has relative large MSE, but still in the acceptable range. It should be noted that both the GP surrogate and the PCA would inevitably introduce additional uncertainties. A rough estimate of these uncertainties can be made by comparing the differences between original solver predictions and the predictions given by surrogate. For most predictions, the difference between surrogate predictions and the original MCFD solver is less than 1%. However, extreme cases exist for predictions with near zero values where differences can be around 10%. It is also worth noting that it is still unclear how to estimate the uncertainty introduced by the statistical methods over the whole validation domain. In this work, we assume the uncertainties introduced by statistical methods is trivial compared to the model parameter uncertainty and model form uncertainty thus can be neglected.

5.2.3 Sensitivity analysis

Similar to the first case study, the solver predictions are averaged to generate a global response for the GSA. This averaging process indicates that the void fraction and velocity in different locations under different flow conditions are treated with equal importance. Thus two globally averaged quantities are analyzed using two GSA methods: the void fraction and gas velocity.

The GSA results using Morris Screening method are displayed in Fig. It can be found that for both QoIs, only three parameters are influential and their importance can be ranked as: $C_l > C_{wl} > C_d$. While the virtual mass force coefficient C_{vm} and turbulence dispersion coefficient σ^t has less influence on both of the QoIs.

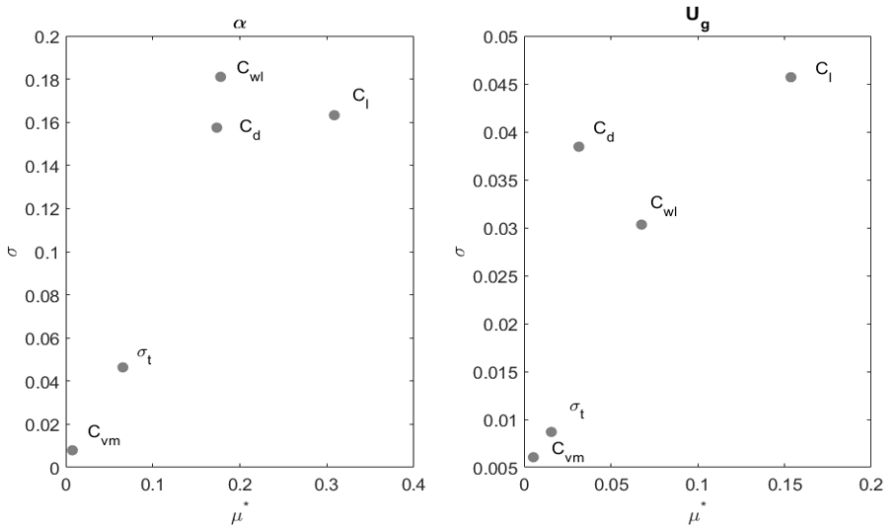

Figure 33. Morris screening measures for interfacial force coefficients.

The Sobol indices plotted in Figure 34 is consistent with the Morris screening results, both the first order indices and total indices confirmed the same importance order.

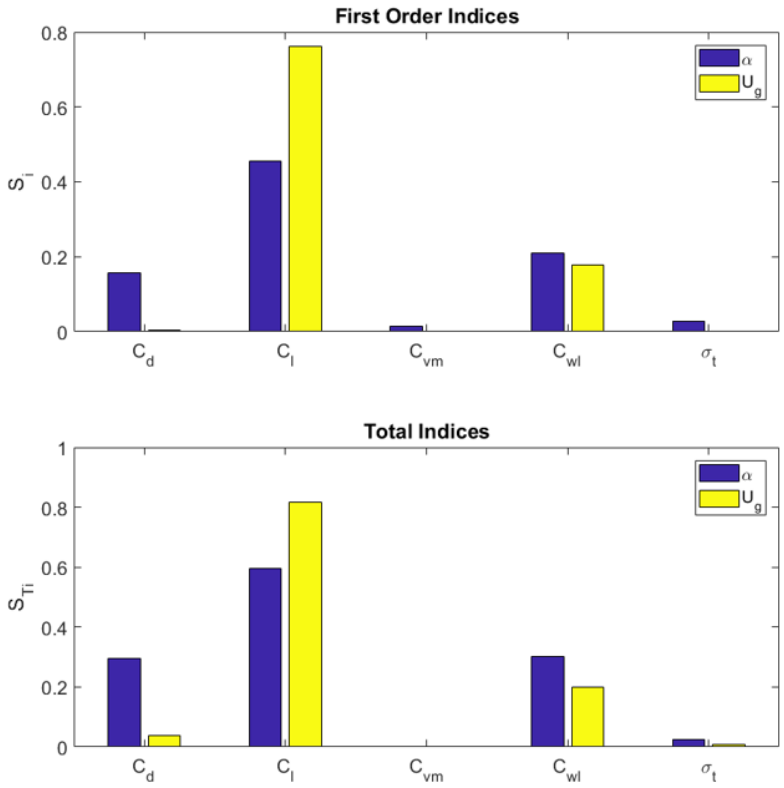

Figure 34. Sobol indices for interfacial force coefficients.

There are only three influential parameters and the preliminary MCMC result shows no parameter identifiability issue. Thus there is no need to perform the parameter selection algorithm. The three parameters are directly applied for the Bayesian inference for uncertainty quantification.

5.2.4 Uncertainty quantification

The model form uncertainty is evaluated using Gaussian process. The hyperparameters of the kernel function are evaluated using the MAP method. Unlike the first case study, the QoIs in this case is a spatial distribution, and there are two inlet conditions. Thus the model form uncertainty term is modeled by a multi-variant GP: $\delta(r/R, j_g, j_l) \sim GP(r/R, j_g, j_l)$. Due to the limitation of available datasets, 4 datasets are used for model form uncertainty evaluation, while 3 are used for model parameter uncertainty evaluation, and one dataset is left for testing. The decomposition is summarized in Table 15. The model form uncertainty distribution as a function of inlet superficial velocities at one location are plotted in Figure 35. It can be observed that the model form

uncertainty of U_g at this location is negative over the whole input space, this suggest U_g is overestimated for all cases at this location.

Table 15. Flow dynamics datasets decomposition for different purposes

Datasets decomposition	Inlet phasic superficial velocities (j_g, j_l) (m/s)
Model form uncertainty evaluation datasets	(0.16,0.64), (0.09,2.0), (0.16,2.0), (0.48,2.0)
Parameter uncertainty evaluation datasets	(0.09,0.64), (0.29,1.1), (0.29,2.0)
Testing dataset	(0.09,1.1)

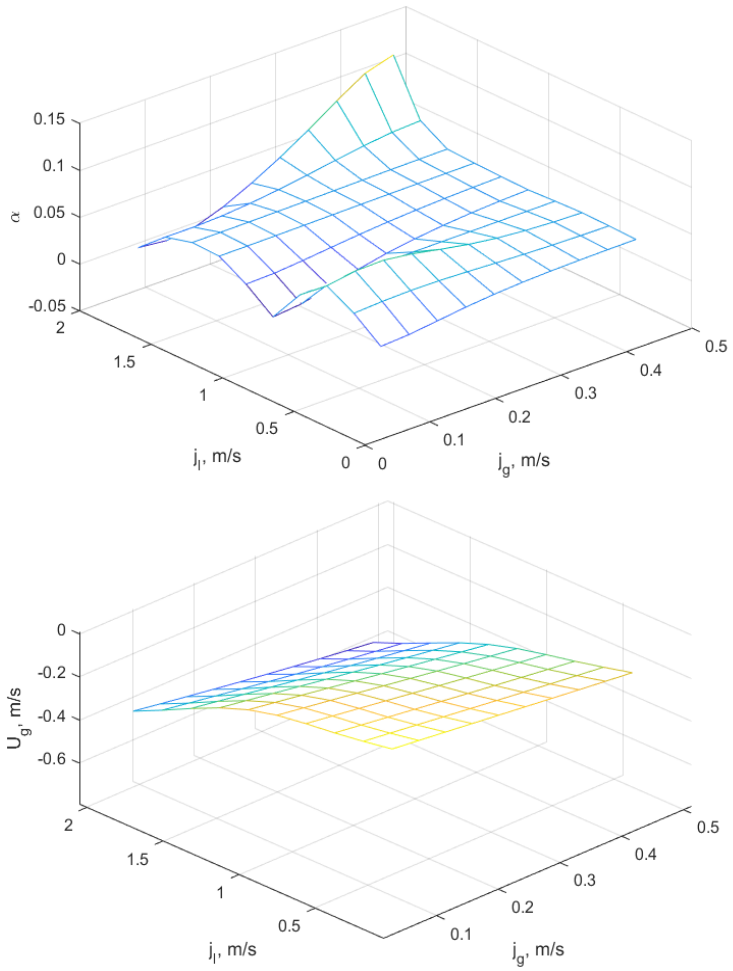

Figure 35. Model form uncertainty distribution at r/R = 0.55.

For the inverse UQ on model parameters, the same process discussed in Section 5.1.5 is applied. The processed sample chain for all selected parameters and the autocorrelations of them are plotted in Figure 36. Good mixing and the fast decay of auto-correlations for all parameters can be observed.

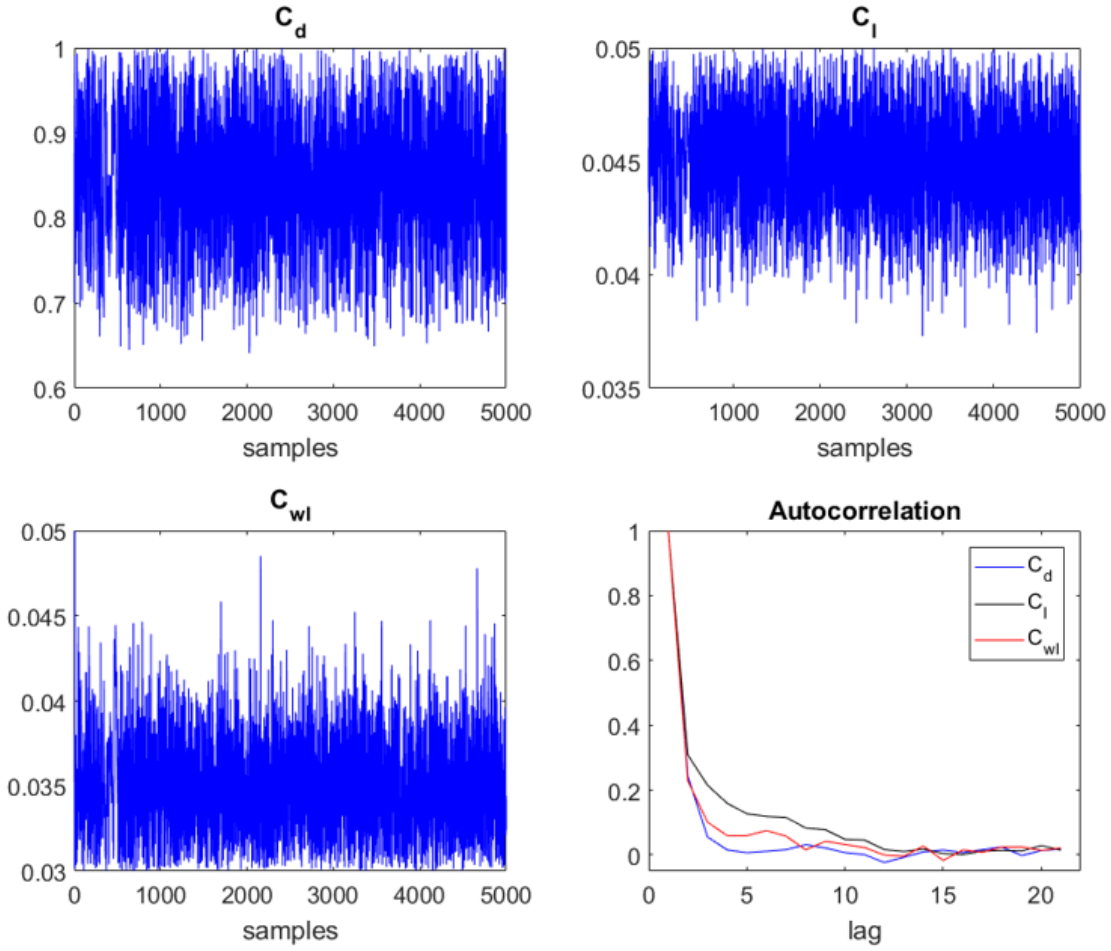

Figure 36. MCMC sample traces and auto-correlations of selected interfacial force coefficients.

The obtained samples can be used to construct the stationary distribution of the Markov chain which can be regarded as the posterior distributions of the parameters. The obtained marginal and point-wise distributions of the three parameters are plotted in Figure 37. The statistics of the parameter distribution are summarized in Table 16. It can be observed that there is a correlation between C_l and C_{wl}. Generally speaking, large value of C_l will result a large value of C_{wl}. This again indicates TDMI can detect the possible interaction between different closure relations.

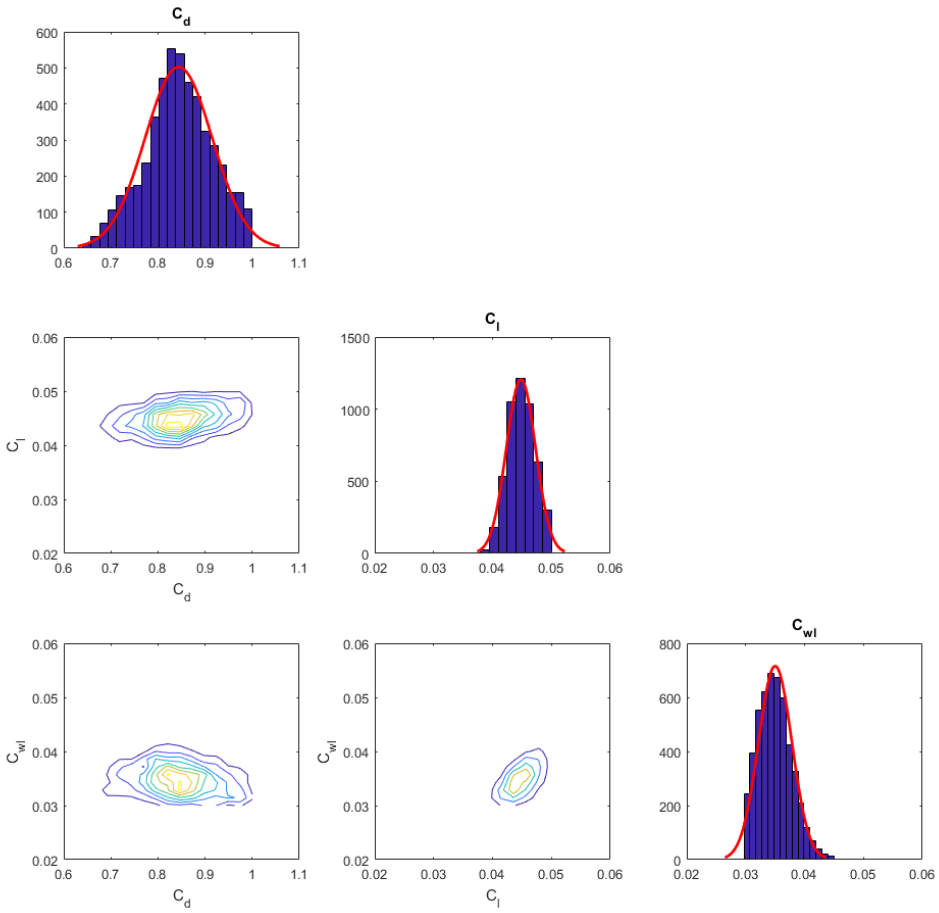

Figure 37. Marginal and pair-wise joint distributions of selected interfacial force coefficients.

Table 16. Summary of posterior distributions of the influential interfacial force coefficients

Parameter	Mean	Standard deviation	95% Confidence interval
C_d	0.8440	0.0711	[0.6977, 0.9788]
C_l	0.0448	0.0025	[0.0403, 0.0492]
C_{wl}	0.0350	0.0029	[0.0304, 0.0412]

Once the parameter uncertainties and model form uncertainties are obtained, the uncertainties of QoIs, with and without model form uncertainty, can be evaluated. The 95% predictive intervals of QoIs are plotted in Figure 38, Figure 39, Figure 40, which represent

the model form uncertainty evaluation case, parameter uncertainty evaluation case and the test case respectively.

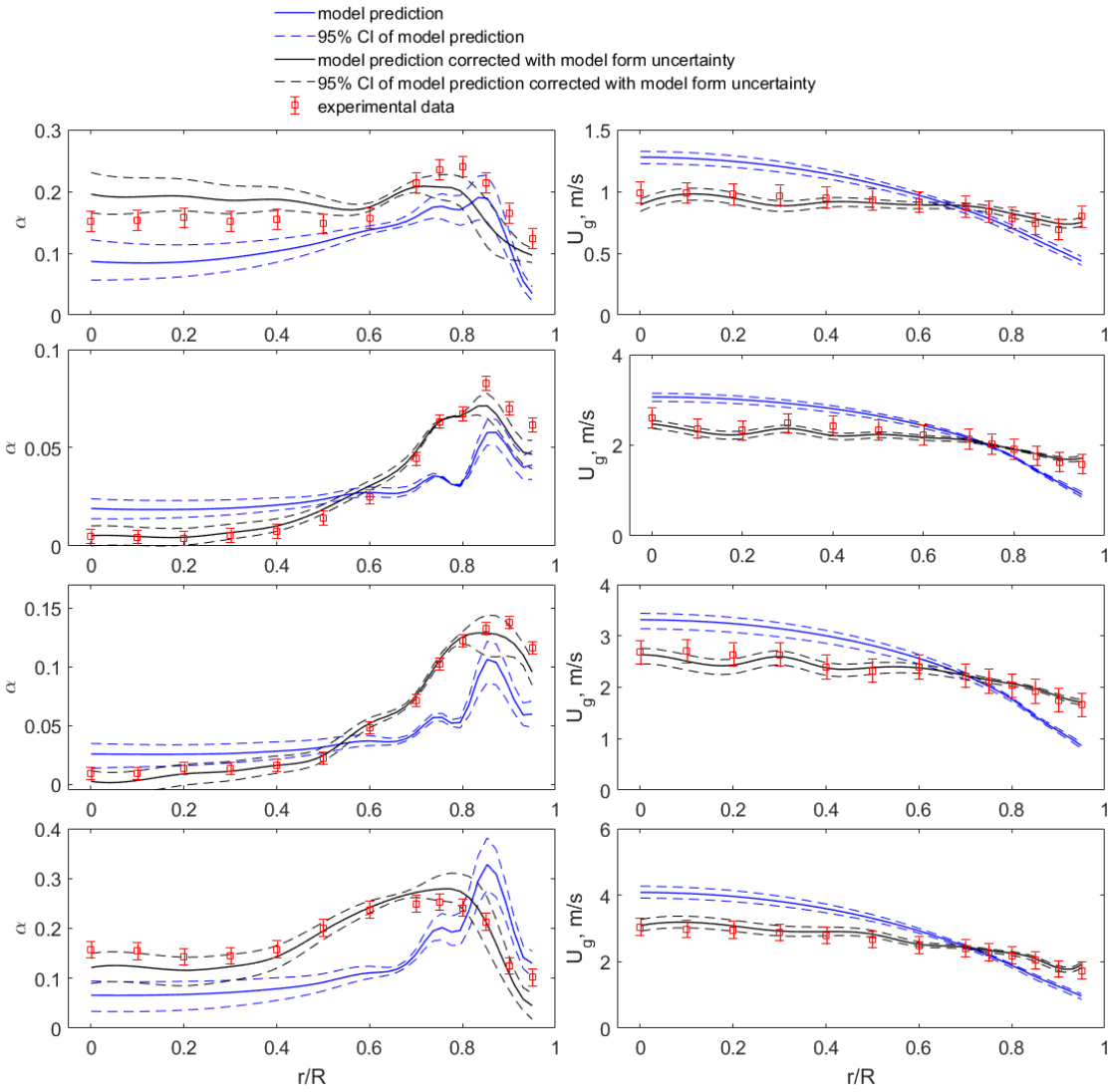

Figure 38. 95% confidence intervals of QoIs for model form uncertainty evaluation cases (first row: j_g = 0.16 m/s, j_l = 0.64 m/s; second row: j_g = 0.09 m/s, j_l = 2.0 m/s; third row: j_g = 0.16 m/s, j_l = 2.0 m/s; fourth row: j_g = 0.48 m/s, j_l = 2.0 m/s).

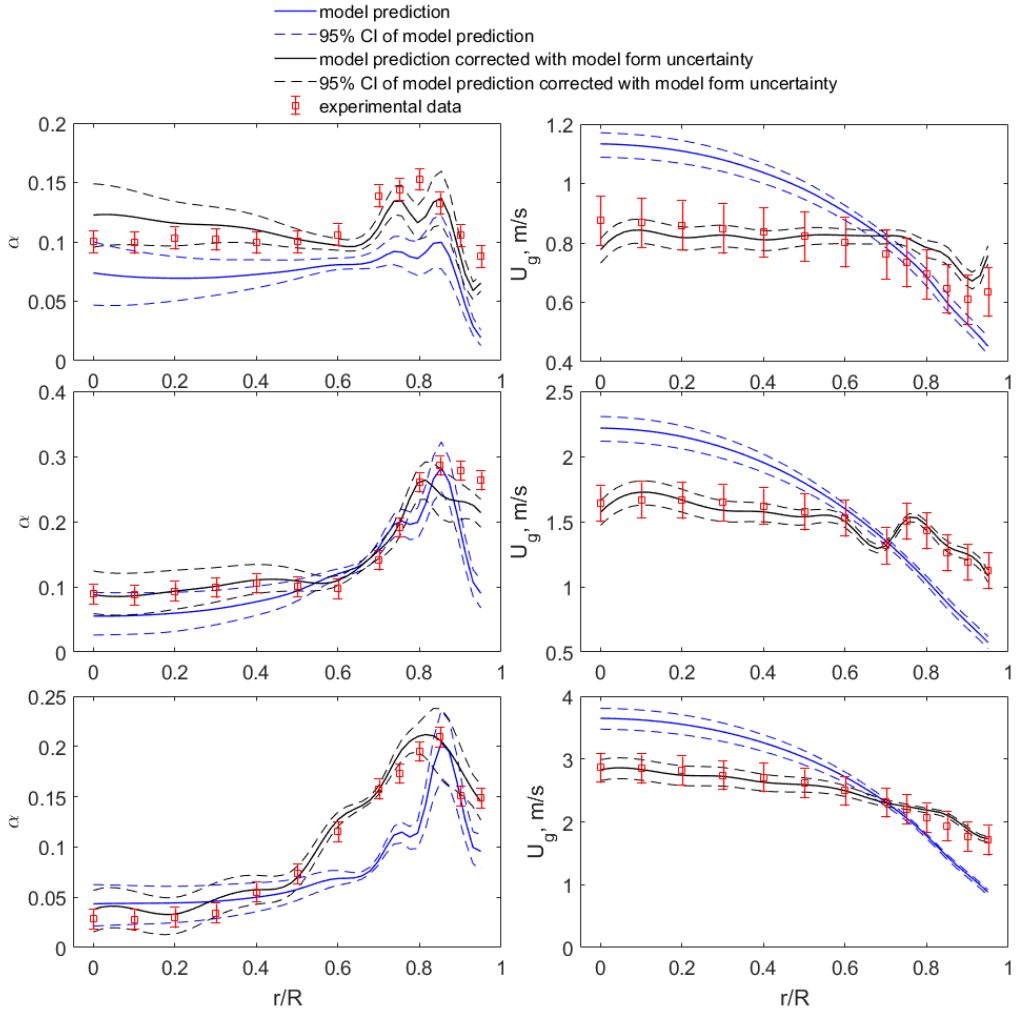

Figure 39. 95% confidence intervals of QoIs for model parameter uncertainty evaluation cases (first row: j_g = 0.09 m/s, j_l = 0.64 m/s; second row: j_g = 0.29 m/s, j_l= 1.1 m/s; third row: j_g = 0.29 m/s, j_l = 2.0 m/s).

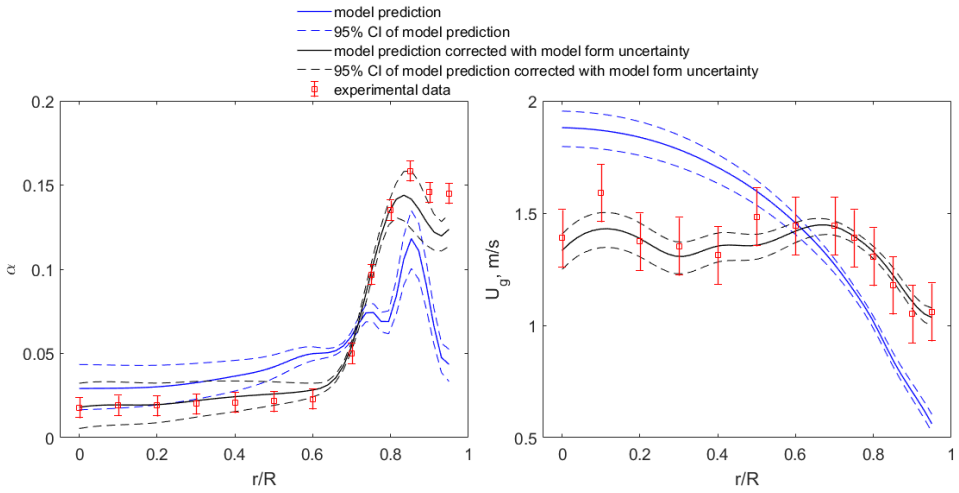

Figure 40. 95% confidence intervals of QoIs for test case (j_g = 0.09 m/s, j_l = 1.1 m/s).

It can be found from these three figures, the calibrated interfacial force closure relations can capture the near wall void fraction peak for all the conditions. The peak location can also be identified with acceptable accuracy. On the other hand, there are still relative large discrepancies between experimental data and solver prediction for the absolute values of void fraction. For the gas velocity, the solver predictions indicate a consistent pattern for all cases: underestimating the velocity in the near wall region while overestimating it in the central region. Such discrepancies, including the void fraction and gas velocity, can be corrected by adding the model form uncertainty term to the solver predictions. This indicate that there are significant model form uncertainties if the constant interfacial coefficients are chosen since significant information regarding the bubble dynamics flow are neglected. Moreover, it also demonstrates that the modular Bayesian approach can incorporate such model form uncertainty.

5.2.5 Validation metrics

The confidence interval and the area metric are calculated in this step. Three representative results, model form uncertainty evaluation case, model parameter uncertainty evaluation case and test case, are plotted in Figure 41 and Figure 42.

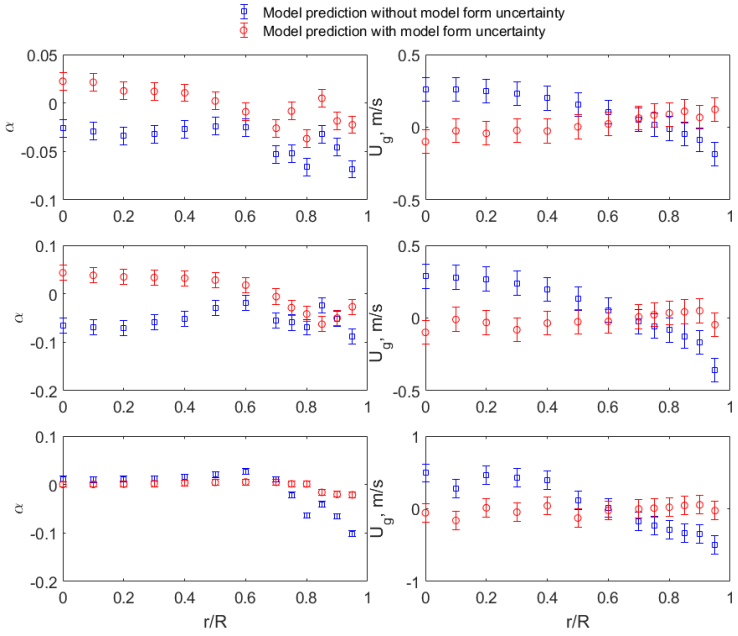

Figure 41. Confidence intervals for three representative cases (first row: j_g = 0.09 m/s, j_l = 0.64 m/s; second row: j_g = 0.16 m/s, j_l = 0.64 m/s; third row: j_g = 0.09 m/s, j_l = 1.1 m/s).

From the confidence interval plotted in Figure 41, several observations can be made:

- Generally, for void fraction, the model prediction is in better agreement with data in the central region than the near wall region. Such a pattern is most significant in the third case, j_g = 0.09 m/s, j_l = 1.1 m/s.
- For velocity, the confidence interval of error is positive in central region and negative in near wall region; this is consistent with the observation from uncertainty quantification results.
- With the consideration of model form uncertainty, the confidence interval of errors become very close to or covers zero. This reaffirmed that the model form uncertainty can be correctly accommodated with the modular Bayesian approach.

The area metrics of the same representative cases plotted in Figure 42 is consistent with the observations from confidence interval.

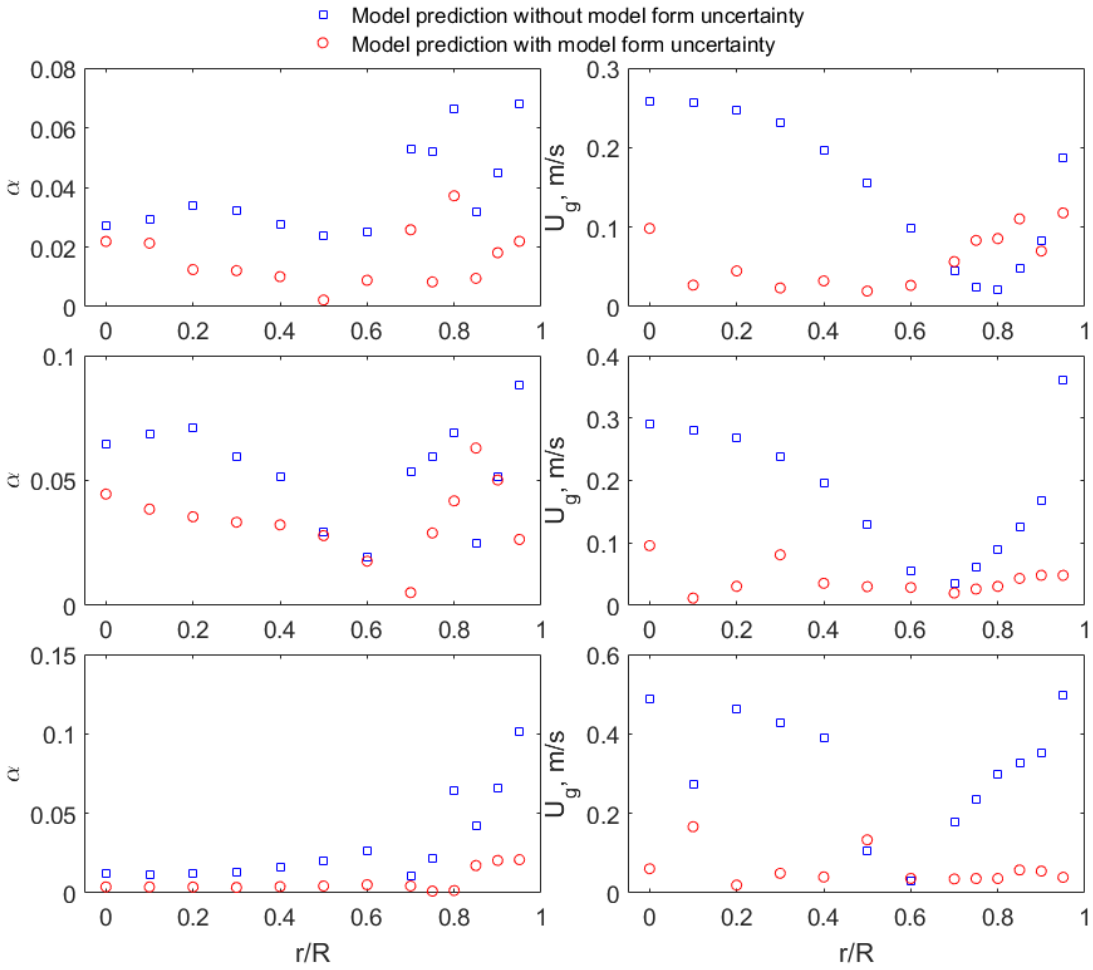

Figure 42. Area metrics for three representative cases (first row: j_g= 0.09 m/s, j_l = 0.64 m/s; second row: j_g = 0.16 m/s, j_l = 0.64 m/s; third row: j_g = 0.09 m/s, j_l = 1.1 m/s).

5.3 Summary remarks

In this chapter, two case studies are performed to demonstrate the VUQ procedure designed for MCFD solver proposed in Chapter 4. One case study focuses on the wall heat transfer in the scenario of subcooled flow boiling in a vertical channel, whereas the other focuses on the flow dynamics in the scenario of adiabatic bubbly flow. Based on the VUQ results, several summaries can be drawn on the relevant closure relations of the MCFD solver:

- For both cases, only a subset of empirical parameters' uncertainties are quantified, whereas nominal values are employed to other parameters. There are two reasons for not quantifying the uncertainties of all parameters. Firstly, some parameters are not influential on the QoIs and thus do not need to be considered in the UQ process. Secondly, possible parameter identifiability issue could exist among certain parameters. The UQ process will hence not update the prior distribution of these unidentifiable parameters. Given non-informative priors, this UQ process will lead to flat posteriors of these unidentifiable parameters, from which one can hardly get useful information for future simulation setup. In this sense, the parameter selection is an essential step for avoiding the parameter identifiability issue.
- For subcooled flow boiling case, the UQ results suggest that with the studied wall boiling closure relation, the MCFD solver can have predictions that are in good agreement with the wall superheat with experimental data, while having a relatively large model-data discrepancy on the wall heat transfer component. Such discrepancy can be quantified by the model form uncertainty term.
- For adiabatic bubbly flow, despite some discrepancy between solver prediction and experimental measurement, the void fraction predicted by MCFD solver, with calibrated interfacial force coefficients, can quantify the near wall peak observed in experimental measurement. But the predictions for bubble velocity tend to overestimate it in bulk flow region while underestimate it in near wall region. For both QoIs, the model-data discrepancy can be reduced by introducing the model form uncertainty term.

The obtained VUQ results confirm feasibility and basic promises of the proposed procedure in several aspects:

- The surrogate modeling is found to be an effective method to alleviate the computation burden for performing the parametric study on MCFD solver.
- The uncertainties of closure relation parameters can be quantified through the inverse UQ with Bayesian inference. Since such UQ process is performed with datasets of different conditions instead of a single case study, the obtained results can be applied over the whole domain covered by data.
- There is significant model form uncertainty of the studied closure relations. These model form uncertainties can be evaluated through the modular Bayesian approach applied in the procedure. On the other hand, this also emphasizes the importance to develop state-of-art closure relations that generalize more experimental evidence to reduce the model form uncertainty.
- The uncertainty of QoIs can be obtained by propagating the parameter uncertainties through the MCFD solver. Validation metrics are calculated to give a quantitative and objective measurement of the agreement between model predicted QoIs and the experimental measurement.

There are still several limitations of the current procedure which requires improvement for further study:

- The numerical uncertainty is not explicitly evaluated in the procedure. As a topic of verification, the numerical uncertainty is not discussed in this work. A more comprehensive evaluation requires the VUQ procedure to be extended to the VVUQ.
- whereas using the statistical model to construct a surrogate for the high dimensional output of the MCFD solver is helpful for significantly reducing the computation burden, additional uncertainties are introduced through this process. Such uncertainty is not evaluated in the current procedure. The cross-validation can give an estimation of such uncertainties by calculating the differences between original solver predictions and the predictions given by surrogate. However, whether such estimation can be extended to the whole validation domain is still unclear.

An additional topic that needs investigation is the data availability for the VUQ of MCFD solver. In current work, the VUQ on MCFD solver is decomposed into two separate case studies due to the limitation of experimental measurement. On the other hand, it should be noted that the high-fidelity simulation demonstrates its potential to be an alternative data source in addition to experiment. Moreover, the feasibility of using high-fidelity computational model to quantify the uncertainty of low-fidelity model has been demonstrated within statistical framework [32]. Based on this, the direct numerical simulations with interface tracking method for two phase flow and boiling [29, 31] could serve for the VUQ of MCFD solver. Moreover, those high-fidelity simulations can provide detailed local information of different physical phenomena, which cannot be obtained with current experimental measurements. Those detailed features can serve as the database for a new model development paradigm: the data-driven modeling, which will be discussed in next chapter.

CHAPTER 6. DATA DRIVEN MODELING OF BOILING HEAT TRANSFER

The nucleate boiling is a complex multiphysics process that involves interactions between heating surface, liquid, and vapor. In the context of two-fluid model, the classical approach for predicting the heat transfer in nucleate boiling is to develop a wall boiling closure with a combination of mechanistic models as well as empirical correlations. However, as has been identified in last chapter, there exist significant model parameter uncertainty and model form uncertainty correlated to the wall boiling closure.

In this chapter, the data driven approach based on deep neural networks (DNNs) is studied which directly learns from high fidelity simulation data to predict the boiling heat transfer. The proposed network takes the near wall features of the boiling system as inputs, including momentum and energy convections, pressure gradients, and surface information. Then trained by the data processed from first principle simulation of pool boiling to predict the boiling heat transfer behavior, including heat transfer components, wall superheat, and near wall void fraction. The trained networks are tested in interpolation case and extrapolation cases which both demonstrate good agreement with the original simulation results. With the good expressiveness and generalization property of the DNN, the obtained closure relation can have minimized model form uncertainty and can be extrapolated to untested conditions.

6.1 Fundamentals of deep learning

The DNNs have some good mathematical properties for dealing with complex problem. It has been proven by [105] that even a two layer neural network of sigmoid activation function can approximate any continuous functions, given large enough hidden units and properly chosen weights. It is further proven by [106] that a DNN can achieve better **expressiveness** with more flexibility. Such property of DNN indicates a properly trained DNN can avoid the model form uncertainty. Moreover, DNN has good **optimization** property. It is also observed from practices that a DNN can be optimized and converges towards global minimum with a straightforward method, i.e. the stochastic gradient descent (SGD), although the theoretical explanation of its success is still weak [107]. The last good property is the **generalization** capability of DNN. It is observed through practice that a DNN usually has good performance in predicting the case it has not

been trained on [108]. On the other hand, although has so many good properties, DNN usually requires significantly large amount of data to train before it can be used for prediction. Moreover, the performance of DNN is highly dependent on the hyperparameters chosen in the training process.

This work is based on the deep feedforward NEURAL networks (DFNNs), which is the most fundamental type of DNN in practical applications, as is illustrated in Figure 43. In deep feedforward network, the first layer is the input and the last layer is the output, in between are the hidden layers, the units in hidden layer are term "neurons" and are represented by a nonlinear function of certain form. The term "deep" indicates the feedforward network has multiple hidden layers, while the term "feedforward" means there are no feedback connections in the hidden layers.

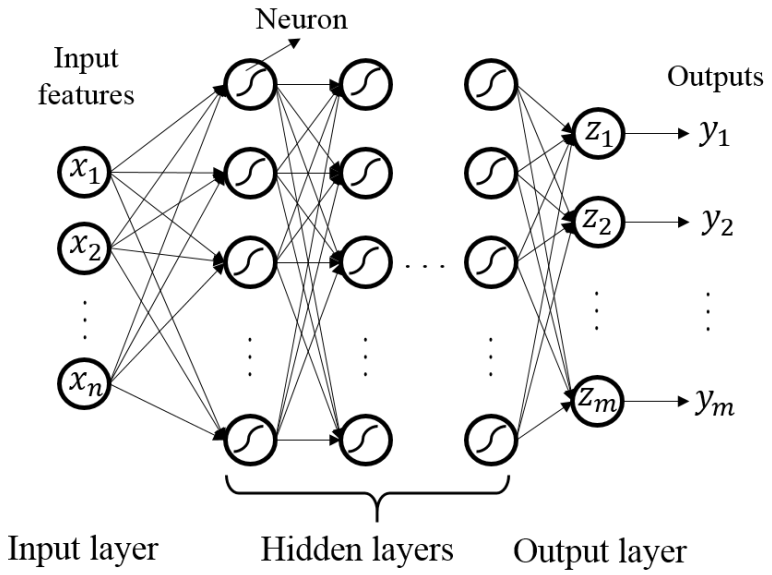

Figure 43. Architecture of a fully connected deep feedforward network.

Essentially, the DFNN can be regarded as a process where the input features go through a series of nonlinear transformations to predict the outputs as QoIs. Most DFNNs do so using an affine transformation controlled by learnable parameters weights $\boldsymbol{W}$ and biases $\boldsymbol{b}$, followed by a nonlinear function named activation function $g(x)$. In this sense, the 1neura$_l$ network in Figure 43 can be interpreted as the following transformations:

$$\begin{aligned} \boldsymbol{h}_1 &= g\left(\boldsymbol{W}_1^T \boldsymbol{x} + \boldsymbol{b}_1\right) \\ \boldsymbol{h}_2 &= g\left(\boldsymbol{W}_2^T \boldsymbol{h}_1 + \boldsymbol{b}_2\right) \\ &\vdots \\ \boldsymbol{y} &= g(\boldsymbol{W}_{l+1}^T \boldsymbol{h}_l + \boldsymbol{b}_{l+1}) \end{aligned} \tag{83}$$

There are many choices of activation function in practice, including sigmoid function, tanh function, rectified linear units (ReLU) [109], etc. In this work, a modified version of ReLU, the Exponential linear unit (ELU) [110] is chosen which can be expressed as follows:

$$g(x) = \begin{cases} \alpha(e^x - 1), & x < 0 \\ x, & x \geq 0 \end{cases} \tag{84}$$

With the input features, number of hidden layer, number of hidden units, weights $\boldsymbol{W}$ and biases $\boldsymbol{b}$, activation function, and outputs setup, the architecture of the DFNN is determined. It will accept an input feature vector $\boldsymbol{x}$ and propagate through every hidden layer to produce an output vector $\hat{\boldsymbol{y}}$, the weight matrix $\boldsymbol{W}$ and bias vector $\boldsymbol{b}$ determines the prediction. Without proper training, it cannot be guarantee that $\hat{\boldsymbol{y}}$ can approximate the real data $\boldsymbol{y}$, thus the prediction would be meaningless. The next step is to train the DFNN with enough data, generally speaking, the deeper the network, the more data required for training. For training a DFNN, a loss function $L(\hat{\boldsymbol{y}}, \boldsymbol{y})$ need to be defined to measure the error between the DFNN prediction $\hat{\boldsymbol{y}}$ and the real data $\boldsymbol{y}$. For physical problems, the L_1 and L_2 norm loss functions are most widely used:

$$\begin{aligned} L_1\ norm\text{: } L(\hat{\boldsymbol{y}}, \boldsymbol{y}) &= \|\hat{\boldsymbol{y}} - \boldsymbol{y}\|_1 = \frac{1}{m}\sum_{i=1}^{m} |\hat{y}_i - y_i| \\ L_2\ norm\text{: } L(\hat{\boldsymbol{y}}, \boldsymbol{y}) &= \|\hat{\boldsymbol{y}} - \boldsymbol{y}\|_2 = \frac{1}{m}(\hat{y}_i - y_i)^2 \end{aligned} \tag{85}$$

Once the loss function is defined the error gradients w.r.t. the weights and biases of each layer can be computed through the backpropagation based on the chain rule of calculus. Thus the training process is a repeat of two steps: first input the data, do a forward propagation through the network, then perform the back propagation to obtain the error derivatives w.r.t. the weights and biases of each layer. Figure 44 demonstrates such

procedure with a simple two hidden layer units and L_2 norm loss function, the bias is assumed to be zero for simplicity.

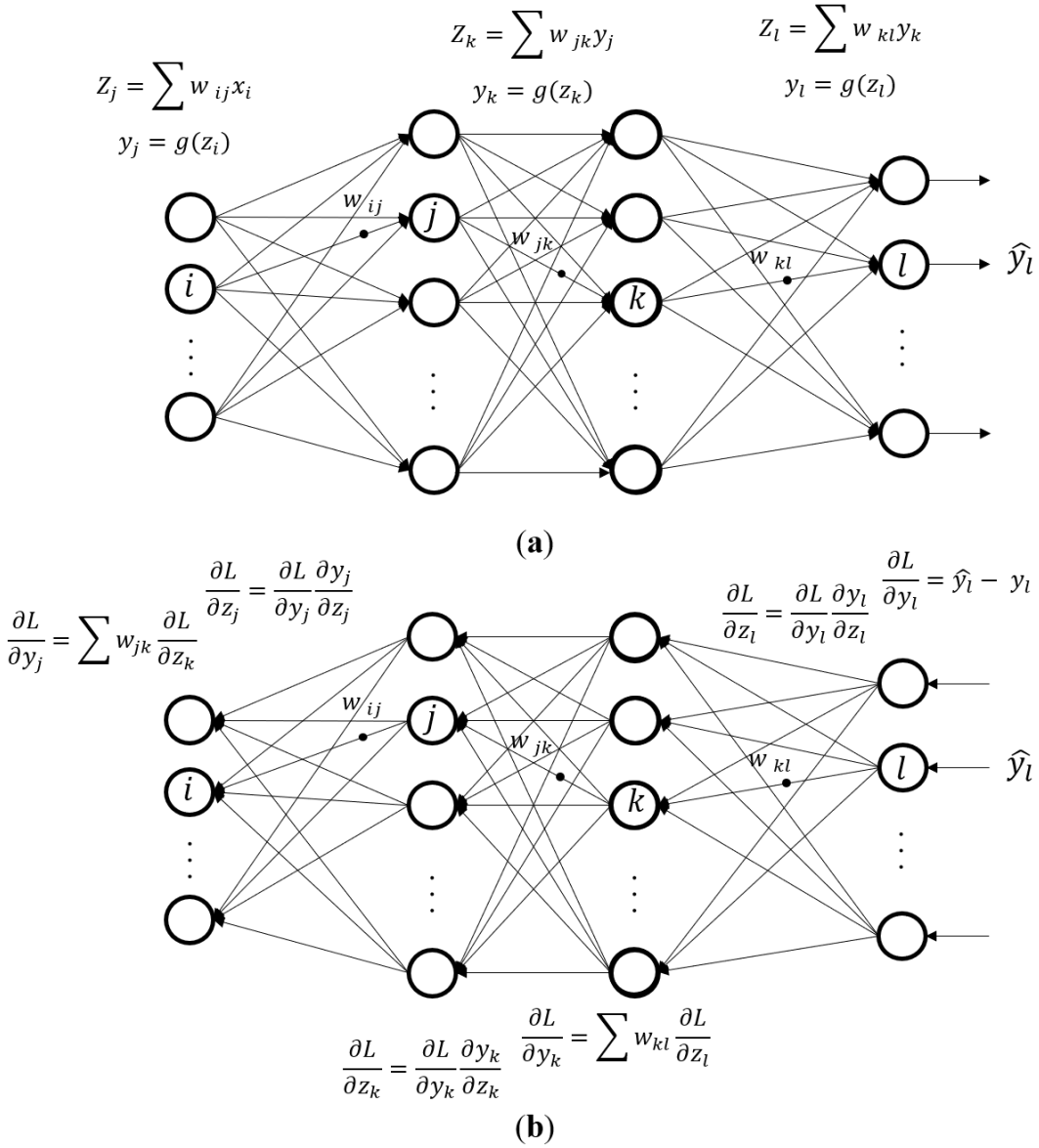

Figure 44. Demonstration of (a) forward-propagation of input features; (b) back-propagation of loss function gradients.

Based on the computed gradients, the weights and biases can be updated with the stochastic gradient descendent algorithm, the goal is to find a set of weights and biases (denoted by $\boldsymbol{\theta}$ here) for every layer to minimize the loss of all data:

$$J(\boldsymbol{\theta};\hat{\boldsymbol{y}},\boldsymbol{y}) = \frac{1}{N}\sum_{i=1}^{N} L(\hat{\boldsymbol{y}}^{(i)},\boldsymbol{y}^{(i)},\boldsymbol{\theta}) \tag{86}$$

$$\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta};\hat{\boldsymbol{y}},\boldsymbol{y}) = \frac{1}{N}\sum_{i=1}^{N} \nabla_{\boldsymbol{\theta}} L(\hat{\boldsymbol{y}}^{(i)},\boldsymbol{y}^{(i)},\boldsymbol{\theta}) \tag{87}$$

$$\boldsymbol{\theta}_{update} = \boldsymbol{\theta} - \varepsilon \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta};\hat{\boldsymbol{y}},\boldsymbol{y}) \tag{88}$$

In this process, the ε is termed learning rate which is the most important parameter during the training whose value directly affect the performance of the network. In practice, a more effective training process is to randomly sample many mini-batches from the full dataset and update $\boldsymbol{\theta}$ with every mini-batch. The size of the mini-batch is usually around hundred level no matter how large the full dataset is. Moreover, more advanced algorithm improved from the standard SGD is usually applied in practice. In this work, the adaptive moment estimation (Adam) algorithm [112] is applied which computes individual adaptive learning rates for different parameters from estimates of first and second moments of the gradients. The training process would go over the datasets many iterations (termed epoch in the deep learning community).

The backpropagation algorithm proposed by [111], for a DFNN with l layers, x as input features, y the output QoIs, the backward propagation of loss derivatives is applied in following way:

(1). Initialize the weights $\boldsymbol{W}^{(i)}$ and biases $\boldsymbol{b}^{(i)}$ for each layer, $i \in \{1, \dots, l\}$
(2). Do a forward pass:

$\boldsymbol{h}^{(0)} = \boldsymbol{x}$
FOR **k** = 1, … , l do
$\boldsymbol{z}^{(k)} = \boldsymbol{W}^{(k)}\boldsymbol{h}^{(k-1)} + \boldsymbol{b}^{(k)}$
$\boldsymbol{h}^{(k)} = g\left(\boldsymbol{z}^{(k)}\right)$
END FOR
$\hat{\boldsymbol{y}} = \boldsymbol{h}^{(l)}$

(3). Calculate the loss function, taking the regularization into account:

$J = L(\hat{\boldsymbol{y}},\boldsymbol{y}) + \lambda\Omega(\boldsymbol{\theta})$

After the forward pass, do the backward propagation of loss derivatives:
(4). Compute the gradient on the output layer:

$\boldsymbol{grad} \leftarrow \nabla_{\hat{\boldsymbol{y}}} J = \nabla_{\hat{\boldsymbol{y}}} L(\hat{\boldsymbol{y}},\boldsymbol{y})$

(5). Convert the gradient on layer's output into gradient on the pre-nonlinearity activation $\boldsymbol{a}^{(k)}$

FOR **k** = $l, l-1 \ldots, 1$ do

$\boldsymbol{grad} \leftarrow \nabla_{\boldsymbol{z}^{(k)}} J = \boldsymbol{grad} \odot g'(\boldsymbol{z}^{(k)})$, $\odot$ stands for element-wise multiplication

(6). Compute the gradients on weights and biases:

$\nabla_{\boldsymbol{b}^{(k)}} J = \boldsymbol{grad} + \lambda \nabla_{\boldsymbol{b}^{(k)}} \Omega(\boldsymbol{\theta})$

$\nabla_{\boldsymbol{W}^{(k)}} J = \boldsymbol{grad} \cdot \boldsymbol{h}^{(k-1)T} + \lambda \nabla_{\boldsymbol{W}^{(k)}} \Omega(\boldsymbol{\theta})$

(7). Propagate the gradients to the next lower level hidden layer:

$\boldsymbol{grad} \leftarrow \nabla_{\boldsymbol{h}^{(k-1)}} J = \boldsymbol{W}^{(k)T} \cdot \boldsymbol{grad}$

END FOR

In the practices of training a very deep DFNN, there are still two common issues: the overfitting and the potential vanishing or exploding gradients. If a DFNN is overfitted with the training data, the total loss function would be small, but it will fail to predict dataset that is not included in the training. A way to minimize the overfitting is to include a regularization term $\boldsymbol{\Omega}(\boldsymbol{\theta})$in the loss function L:

$$\tilde{J}(\boldsymbol{\theta}; \hat{\boldsymbol{y}}, \boldsymbol{y}) = J(\boldsymbol{\theta}; \hat{\boldsymbol{y}}, \boldsymbol{y}) + \lambda \boldsymbol{\Omega}(\boldsymbol{\theta}) \tag{89}$$

where λ is a positive hyperparameter that weights the relative contribution of the regularization term, large λ indicates strong regularization. In most practices, the regularization term is only applied to weights, thus $\boldsymbol{\Omega}(\boldsymbol{\theta})$ is equal to $\boldsymbol{\Omega}(\boldsymbol{w})$. Most widely used $\boldsymbol{\Omega}(\boldsymbol{w})$ are L_1 and L_2 regularization:

$$\begin{aligned} L_1\ regularization\text{:} \quad & \boldsymbol{\Omega}(\boldsymbol{w}) = \|\boldsymbol{w}\|_1 = \sum w_i \\ L_2\ regularization\text{:} \quad & \boldsymbol{\Omega}(\boldsymbol{w}) = \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} \end{aligned} \tag{90}$$

In this work, the L_2 regularization is applied. Moreover, the total dataset is divided into the training data and test data. to exam if a DFNN is overfitted or not. The test data is not included in the training process but is used to test the accuracy of the trained DFNN for inputs that it did not know in the training.

When training a very deep DFNN, the gradients can sometimes get either very big or very small through the backpropagation process. Such unstable behavior makes training difficult or even fail to converge. Currently, there is no universal solution for such issue. A

partial solution is to carefully chose the initialization of weights. In this work, the weights are initialized with state-of-the-art method, and are also closely monitored during the training process.

To summarize: although the DNN has demonstrate its power in many fields, there are still several unclear aspects about its property. In current practices, training a DNN depends on experience and is usually a trial-and-error process. The predictive capability of DNN is highly dependent on not only data but also its hyperparameters: the learning rate ε, the number of hidden layers, the number of hidden units in each layer, the regularization coefficient λ, the mini-batch size, etc. In this work, the effects of those hyperparameters are investigated.

6.2 Problem setup

In this work, the pool boiling simulation results obtained from [46] is used. The simulation uses color function to resolve the liquid-vapor interface, uses large eddy simulation (LES) for turbulence modeling. The computational domain is 40mm×40mm×38mm, the upper 32mm representing the fluid domain, and the lower 6mm representing the solid domain. In the solid domain, a cylindrical-shaped copper block of diameter 20mm locates at the center, and the surrounding solid material is the thermal insulation, for which the thermal conductivity is defined to be zero. The number of nucleation sites is derived based on the experimental measurement [113]. The location of those nucleation site was prescribed as a *priori*, and randomly distributed on the heating surface, together with a nucleation activation temperature T_{act}. The number of cells for the whole domain is 224×224×360. In this work, only the central part of the near wall region, a 10mm×10mm area, is chosen for data extraction in order to minimize the influence of boundary. The computational domain and the data extracted area is depicted in Figure 45.

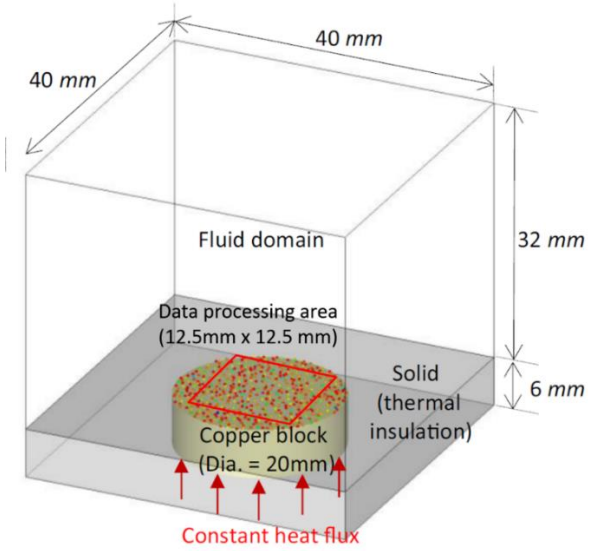

Figure 45. Computational domain and data sampling area of the ITM simulations.

The obtained results were validated against experiments from[113] and [114], with good agreement observed on both cases. Thus the simulation can be regarded as a high-fidelity.

The purpose of the desired DFNN is to use local flow features, which can be obtained in the MCFD solver without boiling closure relations, to predict the boiling heat transfer. The input features for DFNN is important and directly influences its performance. In these boiling cases, there is no well-developed bulk flow, thus the widely used dimensionless flow features such as y^+ and Reynolds number are not applicable. Considering this, comprehensive flow features are chosen to include all the relevant terms in the averaged conservation equations, including pressure gradient term, momentum convection term, energy convection term, and turbulent viscosity. For simplification purpose, these flow features are further averaged between the vapor phase and liquid phase, thus the number of the input features is significantly reduced. Besides the flow features, the features related to the heating surface, including the total heat flux applied to the heating surface q_{Total}, the nucleation site density N_{site}, and the nucleation activation temperature T_{act}, are also included in the inputs. The selected input features are consistent with the fact that the nucleate boiling is a multiphysics process that involves interactions between heating surface, liquid, and vapor. A total 19 features are selected as the inputs of DFNN, which are summarized in Table 17. However, it should be admitted that there should exist more

concise input features that represent the physical characteristics of the boiling process. As the first effort to predict the local boiling process with DNN, such investigation to find better input features is not conducted.

The wall boiling closure in MCFD provide predictions on heat partitioning and wall superheat. For the prediction of departure from nucleate boiling in MCFD solver, the near wall void fraction is also a key parameter. Thus, in this work, these four QoIs are set to be the outputs of the DFNN which are summarized in Table 18.

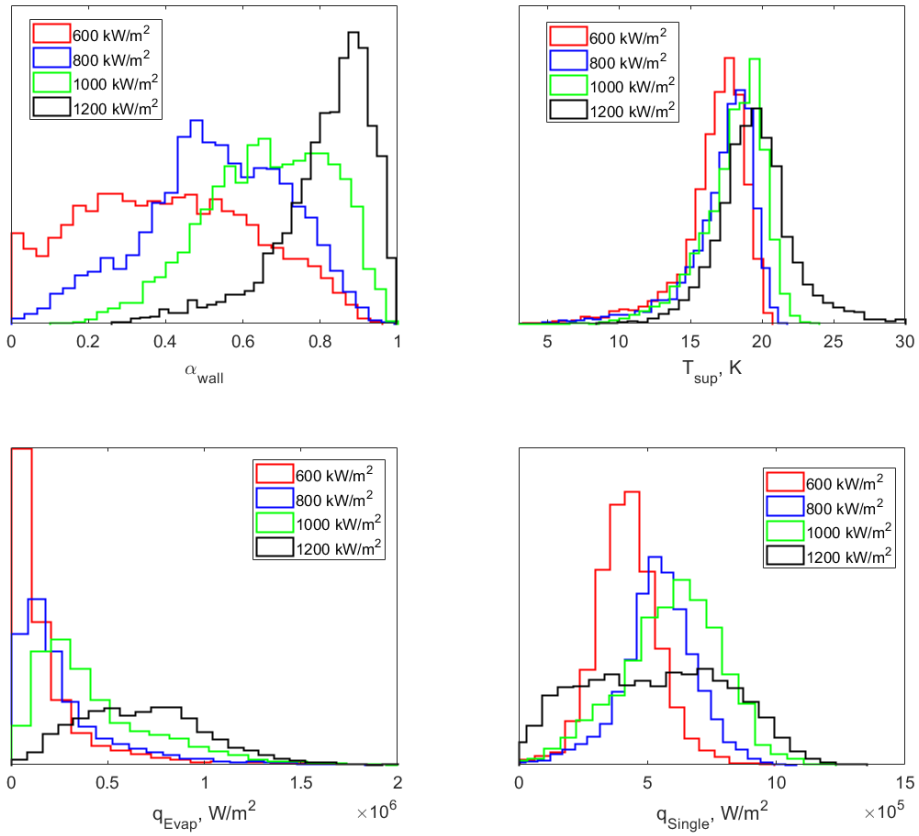

Figure 46. Histogram of 4 QoIs on different input heat fluxes.

It is further assumed that the boiling is only influenced by the near wall flow based on the scale-separation assumption [25], thus only the near wall flow features are processed from the ITM results. In the processing, the average length scale 0.25mm, the average time step is 0.1s. Following the procedure, the near wall region of the whole 224×224×360 computational domain in 100 time frames are averaged to 50×50 near wall flow features, thus 2500 data samples are collected for each simulation case.

For each simulation, constant heat flux is applied at the bottom of the cylindrical copper block. Due to the effect of conjugate heat transfer, randomly distributed nucleation sites, and the random activation temperature assigned to them, the heat flux imposes on the liquid contact surface varies significantly. Thus, the data collect from it covers a broad range of inputs and outputs for the DFNN. Thus, the trained DFNN would have good generalization capability.

In this work, the simulation cases of four different heat flux are used: 600 kW/m^2, 800 kW/m^2, 1000 kW/m^2, and 1200 kW/m^2. The extracted QoIs from 4 different simulation cases are illustrated in Figure 46 in the form of histogram. As can be observed, each case has a significant different distribution, which means each dataset reflects a different pattern.

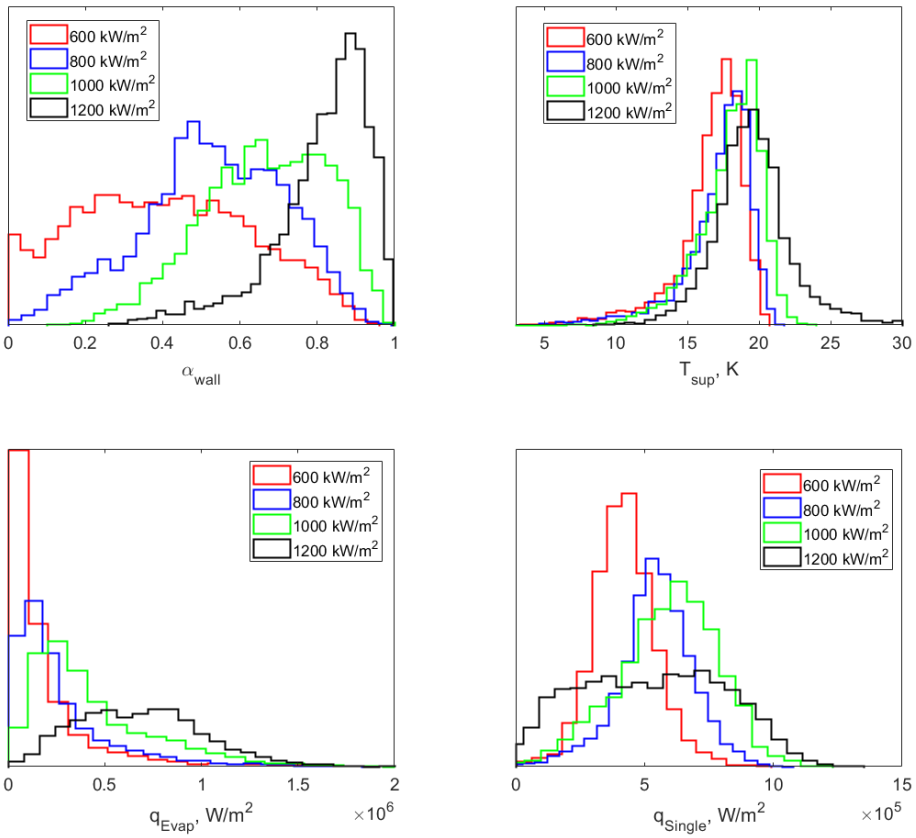

Figure 46. Histogram of 4 QoIs on different input heat fluxes.

Table 17. Summary of input features of DNN

Feature type	Feature expression
Pressure gradient	$\frac{\partial\langle p\rangle}{\partial x}$ $\frac{\partial\langle p\rangle}{\partial y}$ $\frac{\partial\langle p\rangle}{\partial z}$
Momentum convection	$\frac{\partial\langle\rho\rangle\langle u\rangle\langle u\rangle}{\partial x}$ $\frac{\partial\langle\rho\rangle\langle u\rangle\langle v\rangle}{\partial x}$ $\frac{\partial\langle\rho\rangle\langle u\rangle\langle w\rangle}{\partial x}$ $\frac{\partial\langle\rho\rangle\langle u\rangle\langle v\rangle}{\partial y}$ $\frac{\partial\langle\rho\rangle\langle v\rangle\langle v\rangle}{\partial y}$ $\frac{\partial\langle\rho\rangle\langle v\rangle\langle w\rangle}{\partial y}$ $\frac{\partial\langle\rho\rangle\langle u\rangle\langle w\rangle}{\partial z}$ $\frac{\partial\langle\rho\rangle\langle v\rangle\langle w\rangle}{\partial z}$ $\frac{\partial\langle\rho\rangle\langle w\rangle\langle w\rangle}{\partial z}$
Energy convection	$\frac{\partial\langle\rho\rangle\langle h\rangle\langle u\rangle}{\partial x}$ $\frac{\partial\langle\rho\rangle\langle h\rangle\langle v\rangle}{\partial y}$ $\frac{\partial\langle\rho\rangle\langle h\rangle\langle w\rangle}{\partial z}$
Turbulence viscosity	μ_t
Heat flux applied to heating surface	q_{Total}
Potential nucleation site density	N_{site}
Nucleation activation temperature	T_{act}

Table 18. Outputs of DFNN

QoIs	notation
Evaporation heat transfer	q_{Evap}
Convective heat transfer towards liquid	q_{Single}
Near wall void fraction	α_{wall}
Wall superheat	T_{sup}

To summarize, the DFNN is used to predict the near wall boiling behavior, with the averaged near wall flow features that compatible to the two-fluid model as inputs. The high-fidelity simulation results from four pool boiling simulation cases are used for training the DFNN. The architecture of this DFNN is illustrated in Figure 47.

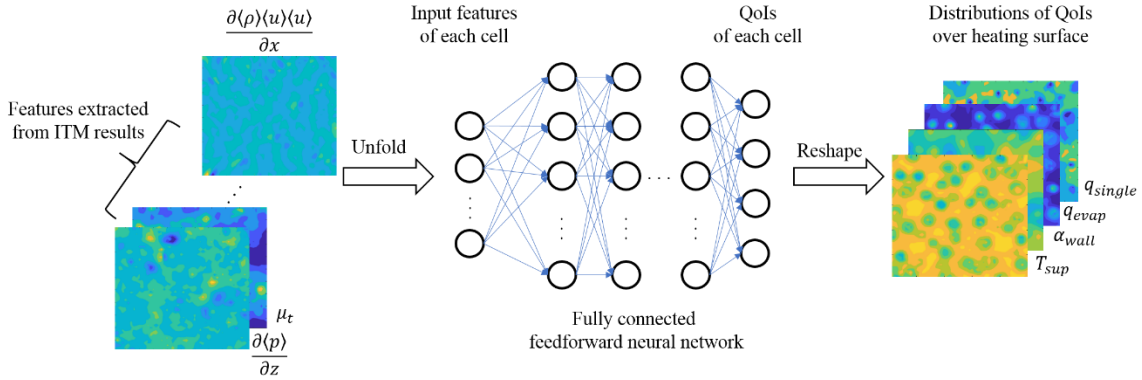

Figure 47. Architecture of DFNN used for predicting boiling heat transfer.

6.3 Results and discussions

In this work, the open source deep learning library Pytorch [115] is used for constructing, training and applying the DFNN. All cases are trained on GPU, which is significantly faster compared to the training on CPU. To test the performance of the trained network, the full dataset is divided into training dataset and testing dataset. As a rule of thumb, for data of medium size (10000 samples of this work), the training data should be 70%-80% of the full dataset, while the rest for testing. Based on this, four different cases are studied as summarized in Table 19. Each case chose a different simulation results as testing dataset, and three other simulation results as training datasets. Thus, four different DFNNs are trained and tested.

Table 19. Case studies based on different training/testing data decomposition

Cases	Training datasets	Testing dataset
Case 1	800 kW/m^2, 1000 kW/m^2, 1200 kW/m^2	600 kW/m^2
Case 2	600 kW/m^2, 1000 kW/m^2 ,1200 kW/m^2	800 kW/m^2
Case 3	600 kW/m^2,800 kW/m^2,1200 kW/m^2	1000 kW/m^2
Case 4	600 kW/m^2,800 kW/m^2,1000 kW/m^2	1200 kW/m^2

Comparing to choosing 25% of the results from each simulation as testing data, this decomposition increased the difficulty of training the DFNN. On the other hand, the DFNN trained in this way should have better generalization capability for unknown inputs, thus can be regarded as better predictive capability. From the perspective of traditional regression, Case 2 and 3 are interpolation cases, while Case 1 and 4 are extrapolation cases. With traditional regression method such as Gaussian process, the interpolation cases should have better accuracy compared to the extrapolation cases.

Before putting into the DFNN, all input features and output QoIs, are zero centered and normalized by the mean and standard deviation of the training dataset. As an approach to avoid the vanishing/exploding gradient during the training process, the weights are initialized with the method (known as "Xavier initialization") proposed by [116]. The ELU activation function defined in Section.2 is used. The L_2 loss function is used, with L_2 regularization term considered. The regularization coefficient λ is tuned to minimize the difference of loss between training data and testing data during the training process. Through a series of testing, λ is set and fixed at 0.0001 for all cases.

It should also be noted that, the influence of hyperparameter is still an active research topic in the deep learning community. The most suitable combination hyperparameters depends on the data and selected input features. Currently, the choose of hyperparameter is still through *ad hoc* analysis. Considering this, the effects of other hyperparameters, including learning rate ε, number of hidden units, and batch size, are also tested based on case 1. The three parameters are sampled over a series of discrete value using the Latin hypercube sampling (LHS) method. A few selected results regarding to the loss on test dataset are demonstrated on Figure 48, in which the average of the root mean square errors (RMSE) of the four QoIs are calculated:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N} L_2(\hat{\boldsymbol{y}}^{(i)}, \boldsymbol{y}^{(i)})} \tag{91}$$

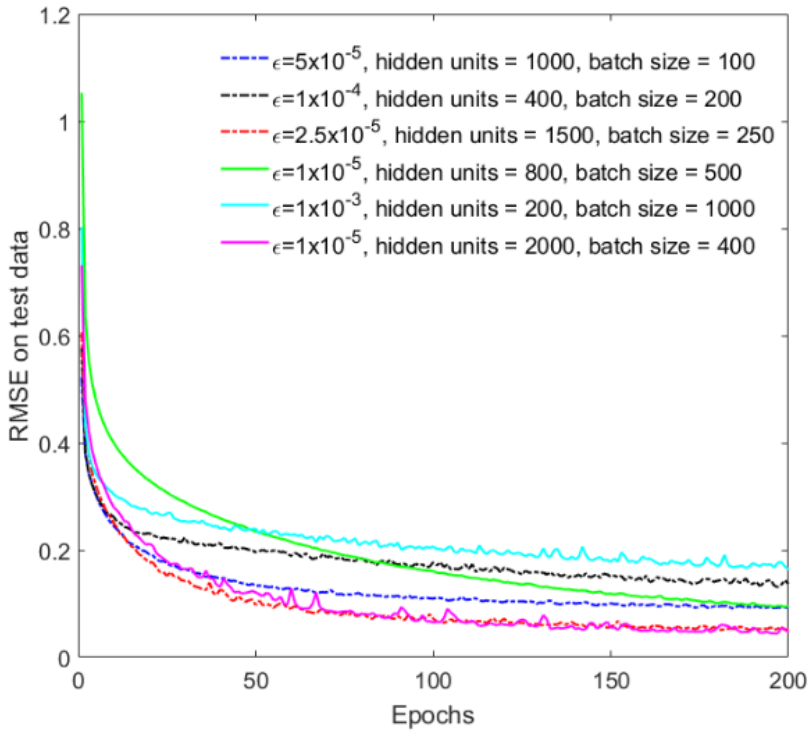

Figure 48. Demonstrating of hyperparameter influence on DFNN performance.

As can be observed in Figure 48, all the three hyperparameters have significant influence on the DFNN performance. Several observations can be drawn from this test:

- Too large learning rate ϵ is a negative factor for the DFNN's performance, while a too small value will not help increase the performance, but will slow the training process instead.
- Generally speaking, a small size of hidden units cannot have good generalization performance on the testing data.
- Large batch size is more computationally efficient on GPU, but cannot produce better performance. In this case, the batch size around 200~500 produce quite similar results.

In following training, the hyperparameters of the DNNs are tuned and adjust through trial-and-error for each case respectively to achieve minimum error in the testing dataset.

The performance of the trained DFNN can be examined through the scatter plots of DFNN predictions on the testing dataset against the original simulation results. Figure 49 and Figure 50 demonstrate the results of the two interpolation cases. In the figures, the solid 45° angle line stands for the ideal situation where DFNN predictions perfectly match the simulation results. The dash line stands for the 2σ bound, where σ is the standard deviation calculated from the differences between DFNN predictions and the original simulation results. The error bound 2σ is less than 20% of the averaged QoIs in both cases. Considering the complicity of the problem, and the large uncertainty of the classical wall boiling closures, the prediction of DFNN has lower uncertainty and thus can be regarded as an improvement for this problem. It can be further found from both figures that although some outliers are observed, most of the DFNN predictions are within the 2σ bound. Considering there are 2500 samples in the plot, those few outliers are not statistically significant.

Moreover, it can be observed in Figure 49 that the DFNN predictions on α are evenly distributed in the 2σ region, demonstrating a Gaussian distribution pattern, but the other three QoIs show certain skewness. Especially for q_{Evap} and q_{Single}, where in general q_{Evap} is underestimated and q_{Single} is overestimated by the DFNN prediction. Similar, but less significant, trends can also be observed in Figure 50. In deep learning community, such discrepancy is termed as high variance (noted that it is a different definition of variance compared to the widely used definition in statistics community), and can usually be resolved by increasing the training data.

The two extrapolation cases are depicted in Figure 51 and Figure 52. Again, the high variance problem is observed in Figure 51, where T_{sup} and q_{Single} are overestimated, while q_{Evap} is underestimated. On the other hand, it is also observed that both the 2σ bound and the scatter plots show quite similar pattern compared to the interpolation case. This indicates the DFNN generalized both the interpolation and the extrapolation cases with similar performance.

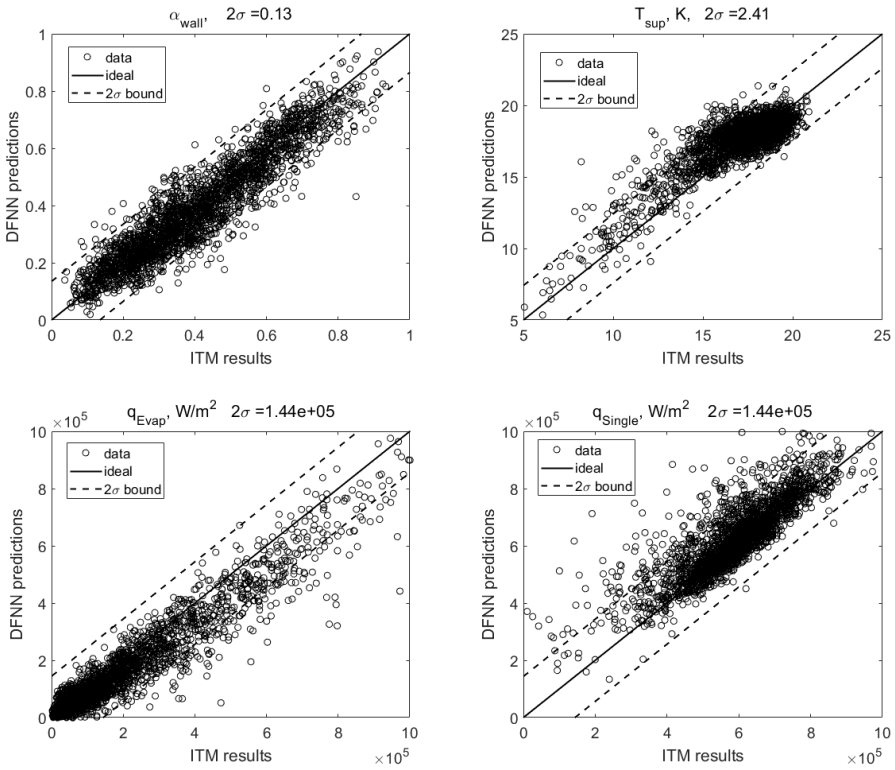

Figure 49. Comparison of DFNN predictions and real ITM simulations (Case 2).

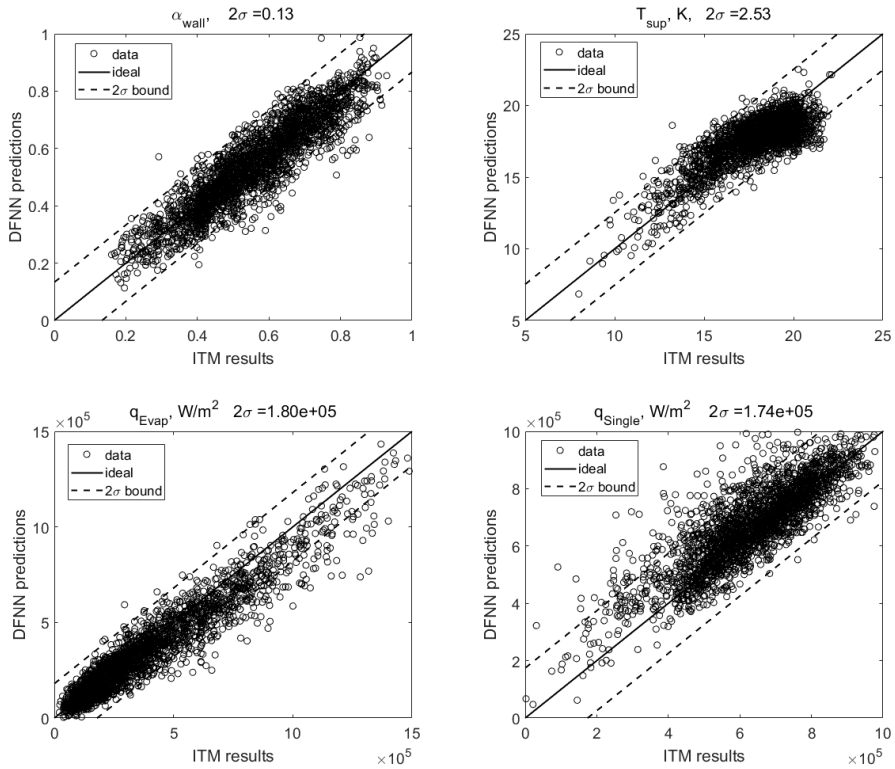

Figure 50. Comparison of DFNN predictions and real ITM simulations (Case 3).

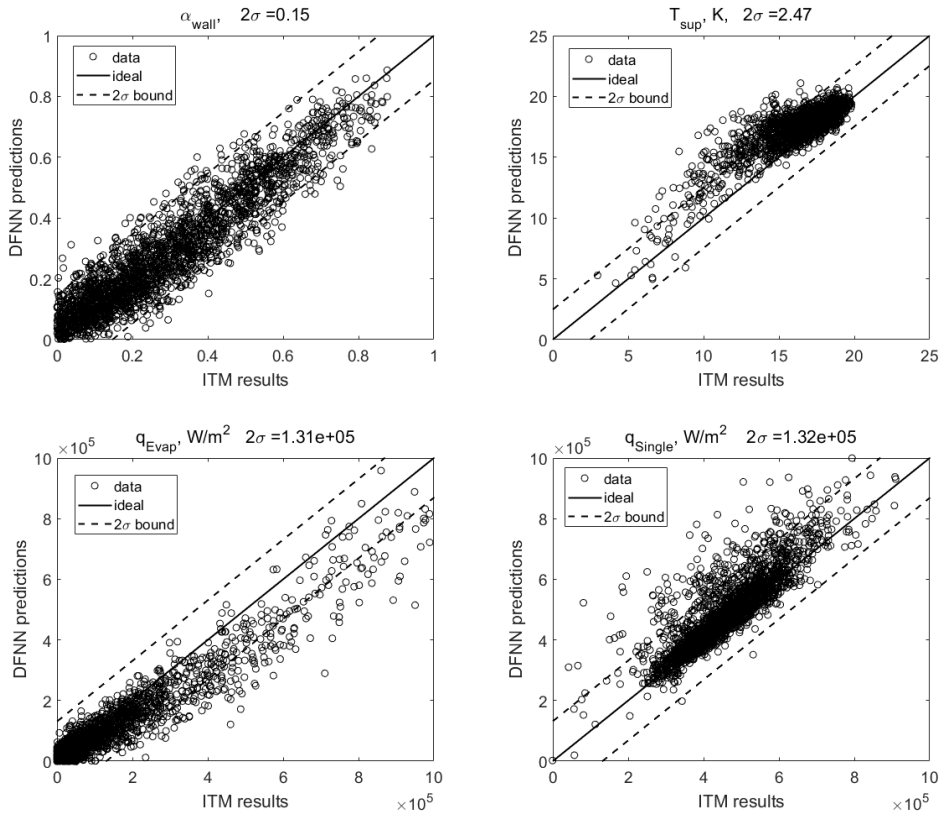

Figure 51. Comparison of DFNN predictions and real ITM simulations (Case 1).

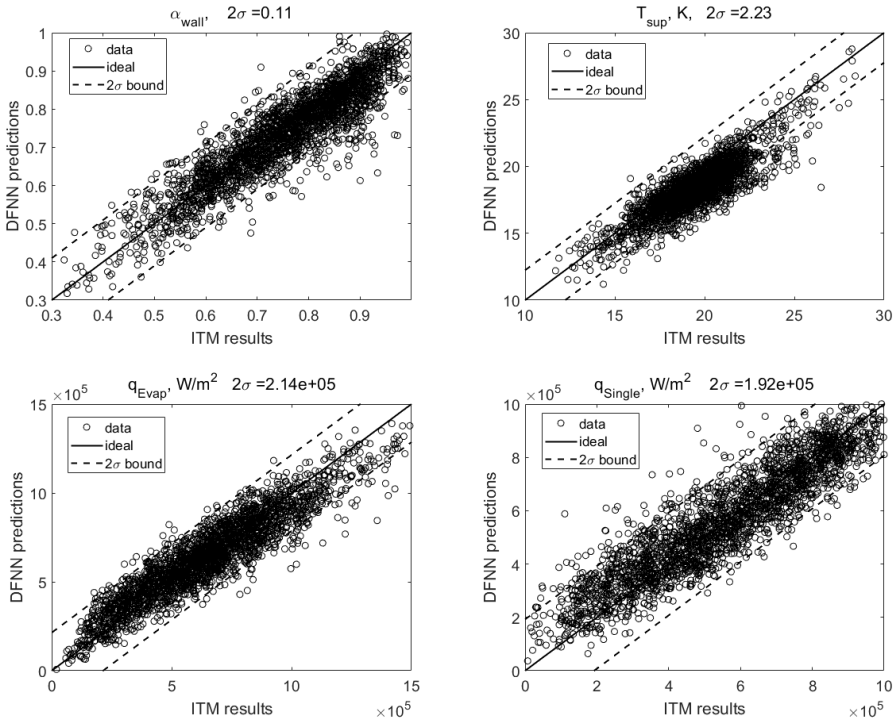

Figure 52. Comparison of DFNN predictions and real ITM simulations (Case 4).

A more quantitative comparison, i.e. the RMSE of the DFNNs on the prediction of test datasets are summarized in Table 20. It can be found the RMSE for all QoIs are below 20% of the averaged values. This confirms the DFNNs predict the original simulation results with reasonable good agreement. On the other hand, it is also noticed that the absolute value of the error is evenly distributed in the whole testing dataset. This means if judging by the error in percentage, the errors for small value predictions would be significantly higher than the large value predictions. This issue stems from the optimization process applied in current DFNN training. In the training process, the whole datasets are divided into several batches, each batch contains hundreds of data. The weights and biases of the DFNN are updated for each batch. This is a much more stable and efficient method compared to update the weights and biases for every single data. However, when using the L_2 norm loss function will average the data in each batch, thus small value data would be less important compared to large value data. In this sense, the trained DFNN would favor the large value data. To overcome this issue, new forms of loss function that specially designed for physical problems should be developed.

Table 20. RMSE on test data of each case

	α_{wall}	T_{sup}, K	q_{Evap}, W/m^2	q_{Single}, W/m^2
Case 1	0.074	1.57	7.17×10^4	7.29×10^4
Case 2	0.070	1.27	8.15×10^4	8.01×10^4
Case 3	0.068	1.29	9.73×10^4	9.65×10^4
Case 4	0.057	1.59	1.07×10^5	9.67×10^4

The predictive capability of the DFNN can be further demonstrated through the global boiling pattern prediction. The visual comparison between DFNN predictions and the original ITM results on the heating surface is depicted in Figure 53 (Case 2) and Figure 54 (Case 4). The original ITM simulation results suggest two different boiling patterns in these two cases. In Case 2 ($q_{total} = 800$ kW/m^2), the individual nucleation sites can be clearly identified which suggested the frequently activated nucleation location in the simulation. The boiling in this case is the well-developed nucleate boiling. Whereas in Case 4 ($q_{total} = 1200$ kW/m^2), the results show the trend of transition from nucleate boiling to film boiling. For both cases, it can be found the DFNN prediction captures the global

boiling pattern with good accuracy. This suggests that the DFNN relies on local features can not only give good prediction on local boiling process, but also captures the global boiling pattern.

Here a hypothesis is proposed to explain the good performance of DFNN on the extrapolation prediction. One the one hand, the DFNN in this work relies on the local features of flow and heating surface as inputs. Those local features can be characterized with a certain pattern, given that their global conditions are in the same regime. For example, in fully turbulent flow, the near wall flow can be characterized by the boundary layer theory, no matter the Reynolds number is 50,000 or 80,000. This means the wall function based on boundary layer theory can be extrapolated to predict the near wall flow for any fully turbulent flow case. However, traditional regression methods, such as Gaussian process, cannot identify such pattern (like boundary layer theory), with even unlimited data. Because these regression methods have limited capability in both expressiveness and generalization. Limited expressiveness capability means these methods cannot approximate complex functional forms if not specified first. Limited generalization capability means these methods cannot identify the intrinsic pattern of a certain problem from a large dataset. In this sense, these traditional regression methods cannot be trusted for extrapolative prediction, as has been already suggested in various practices.

On the other hand, the DNN has demonstrated good capability on both expressiveness and generalization. As being discussed in Section 1, a properly trained DNN can approximate any form of continuous function. Also, the DNN has demonstrated good generalization property in the application of natural language processing and computer vision. In this sense, the DNN has the potential to identify the intrinsic pattern of a real physical problem from a large set of data, and describe this pattern with enough accuracy. From this point of view, the DNN can serve as not only a statistical tool to replace certain closure relation, but also a promising tool to help discover insights of complex physical problems. The latter depends on the progress of the interpretability of DNN, which means the logic of DNN predictions hidden inside the network should be made understandable by people. This is a very challenging topic but already attracts interests of many researchers [117].

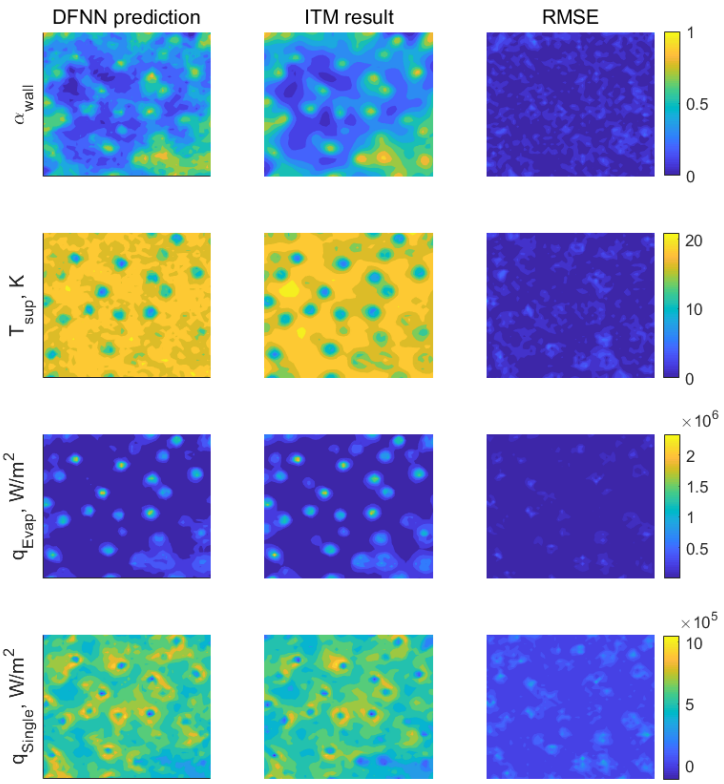

Figure 53. Visual comparison of DFNN predictions and ITM results (Case 2).

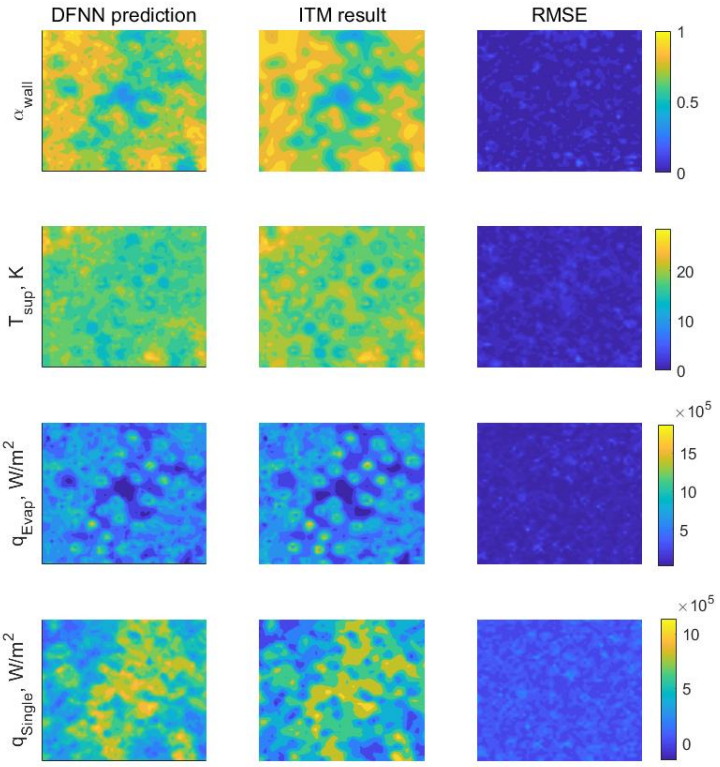

Figure 54. Visual comparison of DFNN predictions and ITM results (Case 4).

6.4 Summary remarks

In this chapter, the data driven approach is studied which takes local flow surface features as inputs to predict the heat transfer behavior in pool boiling using deep feedforward neural networks (DFNN). The networks are trained on data extracted from high fidelity pool boiling simulations with interface tracking method (ITM). The accuracy of the networks is tested through four case studies, including both interpolation and extrapolation cases. Reasonably good agreement between the DFNN predictions and the original ITM simulations are found for both the interpolation case and the extrapolation case. Moreover, the global boiling pattern over the heating surface can also be captured with the DFNN. Especially, the boiling patterns can be captured through DFNN, even for extrapolation case. This indicates the deep network trained with local flow features have good generalization property and thus can be trusted to be extended to unknown conditions. The results demonstrate the deep neural networks can be a promising tool to help improve the predicative capability of Multiphase-CFD solver. Further, a hypothesis is proposed to explain the good performance of DFNN on the extrapolation prediction. Based on this hypothesis, the DNN can also be a promising tool to help discover insights of complex physical problems.

On the other hand, there are still limitations of this work. Firstly, the networks studied in this work are trained with pool boiling simulation results, while for industrial problems, the flow boiling is usually the focus. Due to the significantly different flow and boiling patterns between the two scenarios, the DFNN trained with pool boiling data is questionable to be applied to flow boiling scenarios. Moreover, the flow features chose in this work is based on the terms of conservative equations, most of which are scale and geometry variant. If a network is designed to have universal predicative capabilities, its features should be scale invariant, as demonstrated in [37]. It worth noting that, a most recent work showed promising results to use convolutional neural network to automatically extract physics based features [118]. Finally, the accurate prediction for boiling heat transfer depends on the accurate input of flow features. Such premise cannot be guaranteed with flow features calculated from a MCFD solver. In this sense, to incorporate the deep network in a MCFD solver, the network should also be coupled with that solver in the training process, as is suggested in [119].

CHAPTER 7. CONCLUSIONS

Although being considered as one of the most promising analysis tools for industrial two-phase flow and boiling systems, the Eulerian-Eulerian two-fluid model based MCFD solver still has an unresolved issue that undermines its predictive capability: the significant uncertainty within the closure relations of the solver is not well quantified.

In contrast to the efforts to develop new closure relations that aim to describe the still unclear physical process, this dissertation proposes two approaches to address this issue within the data-driven analysis framework. The first is to perform validation and uncertainty quantification (VUQ) for the MCFD solver. This process is based on the idea of total data-model integration (TDMI) that treat solver, closure relations, and all available datasets in an integrated manner within the Bayesian framework. The second is to develop data-driven closure relations based on deep neural networks (DNNs) with local flow features for the MCFD solver. With the good expressiveness and generalization property of the DNN, the obtained closure relation can have minimized model form uncertainty and can be extrapolated to untested conditions.

The major contribution of the dissertation is the exploration of this new data-driven paradigm, which the author sees of great potential, to analyze complex physical processes of a system such as boiling. In contrast to the traditional analysis paradigm which is mainly driven by expert experience, this new paradigm is directly driven by data, thus leverages the power of increasingly rich data resources from high-resolution experiments and high-fidelity simulations.

This chapter concludes the dissertation by summarizing the proposed framework, highlighting the contributions, and outlining future works on this topic.

7.1 Summary

In this dissertation, a novel approach was presented for uncertainty quantification and model development in boiling problems using a data-driven framework with state-of-the-art statistical algorithms. The framework is applied to MCFD solver and serves for two important practical purposes: 1). quantifying and reducing the uncertainties of the MCFD

solver for boiling simulations; 2). quantitatively validating the solver predictions against available datasets.

First, a data processing and storage procedure is developed as a component of the framework. For high fidelity simulations results, the time and spatial average process is applied to obtain two-fluid model compatible data. For high-resolution experimental measurements obtained by IR camera, the hierarchical clustering algorithm is applied to identify the active nucleation sites and record the corresponding nucleation information. The obtained data are then stored in virtual containers and can be incorporated into the following VUQ and data-driven modeling work.

Second, a six-step data-driven VUQ procedure is developed for the MCFD solver. The procedure builds a surrogate model for the MCFD solver predictions. Based on the surrogate, a subset of parameters is selected and went through a modular Bayesian inference. With the provided datasets, the model form uncertainties are evaluated, and the parameter uncertainties are inversely quantified. The obtained uncertainties are then propagated through the solver to obtain the uncertainties of the solver predictions. Validation metrics are calculated as a quantitative measurement of the agreement between solver predictions and the datasets. Two cases studies with MCFD solver are performed based on the procedure, the obtained results demonstrate the applicability of the procedure.

Last, the data-driven approach based on deep feedforward network is studied which learns from high fidelity simulations to predict the boiling heat transfer. The proposed network takes the local flow features of the boiling system as inputs to predict the boiling heat transfer behavior. The networks are tested in both interpolation case and extrapolation case, which both demonstrate reasonable agreement with the original simulation results. A more promising observation is the trained network can accurately capture the global boiling patterns, even for extrapolation case.

7.2 Contributions

As stated, this dissertation explored the new data-driven paradigm in boiling problem analysis, which are reflected by the following contributions:

1). **The exploration of machine learning based automatic data processing, compared to the traditional human effort based data processing.** Traditionally, the nucleation events are manually identified and recorded from consecutive boiling images. Such work is generally tedious and time consuming. In this dissertation, the hierarchical clustering algorithm is applied to boiling images which can automatically identify the nucleation events with high accuracy and efficiency. Such an efficient data processing procedure is crucial for the data-driven paradigm which usually requires a significant amount of data for the analysis purpose.

2). **The application of Bayesian inference to inversely quantify the parameter uncertainty, compared to the expert judgment based parameter tuning.** Running simulation for boiling problems using MCFD solver involves many parameters. Traditionally, the uncertainties associated with these parameters are not well characterized and only roughly evaluated with expert judgment. In this dissertation, the Bayesian inference is applied to inversely quantify the uncertainties of those parameters, with the support of QoI datasets. Moreover, the model form uncertainty is also evaluated through this modular Bayesian approach. This process provides a rigorous estimate of the uncertainty of QoIs, which is essential for the validation process.

3). **The application of validation metrics to quantitatively measure the agreement between solver predictions and the datasets, compared to the traditional "graphical comparison".** Validation is a necessary step to evaluate the predictive capability of a simulation tool. The most commonly used approach in engineering community is to directly display the simulation results and data on a graph. Such "graphical comparison" cannot generate a quantitative measurement of the simulation-data agreement and can hardly lead to a reasonable evaluation of the solver. In this dissertation, two validation metrics, the area metric and confidence interval, are applied as a quantitative measure of the agreement between data and solver prediction. This type of validation metric considers the full uncertainty distribution of the solver predictions as well as the datasets, thus can be regarded as a comprehensive evaluation.

4). **The exploration of data-driven modeling for boiling problems using deep learning algorithms, compared to the expert experience based modeling approach.**

The closure relations in MCFD solver has a significant influence on the solver predictions. A closure relation is developed mainly based on expert experience and has limited applicable range within the condition it has been tested. In this thesis, the data-driven approach in the form of deep neural networks for closure relation development is studied. This approach leverages the "big data" from high-resolution experiments and high-fidelity simulations. The deep neural network has good expressiveness and generalization property, thus a well-trained deep network based closure relation should minimize the model form uncertainty and can be extrapolated to untested conditions. The results obtained in this dissertation demonstrates the network trained by local features cannot only predict local boiling heat transfer behavior with good accuracy but also capture the global boiling pattern, even for the extrapolation case.

7.3 Future work

As discussed in several chapters, some improvements can be made to the work discussed in this dissertation such as:

- Performing the VUQ procedure on a comprehensive case study, which including both boiling and two-phase flow QoIs.
- Using validation metrics such as p-box that can differentiate the epistemic uncertainty and aleatory uncertainty.

Besides those relatively minor improvements, there are three major areas that the author considers important for future work and the extension of this dissertation.

In the data processing work, current practice is to obtain different QoIs from experimental measurements of one test facility. However, it is not possible to measure all the QoIs of a boiling system in one single facility. A comprehensive data-driven analysis thus requires exploiting multiple data sources and heterogeneous data from different experimental facilities. To achieve this, rigorous scaling criteria for two-phase flow and boiling should be developed. Moreover, an evaluation procedure for the value of datasets, depending on their relevance and scaling distortion, should be developed.

In the VUQ work, MCFD simulations are performed, results over the global computation domain are obtained. Yet only a few local points are extracted to perform

Bayesian inference. The reason to do so is there lacks global measurement for the whole flow and heat transfer field. A desired improvement in future is to perform VUQ on the whole flow field. Two challenges need to be addressed to achieve this goal. The first is to develop an advanced image processing method to reconstruct the whole flow field from the images from high speed camera. Comparing to current boiling images (which represents the two-dimensional heating surface), this is more difficult since the obtained two-dimensional image is an incomplete representation of the three-dimensional real world. The second is the surrogate modeling for the whole flow field, which could have millions of outputs depending on the meshes of the computation domain.

In the data-driven modeling work, the deep feedforward network takes local input features determined by the author. Such determination may not be an "optimized" choice, since some of the features may not be important to the QoIs. More importantly, some features that are significant to the QoIs could be neglected. A desired improvement is to have an approach that can automatically identify and extract the important and efficient features for the deep neural network.

In the ultimate picture, the flow and boiling images obtained from different high-resolution experiments can be automatically processed to produce the averaged flow fields that compatible with the two-fluid model. This obtained flow field will be used for the VUQ of the MCFD solver, which is supported by the closure relations in the form of deep networks with efficient input features.

REFERENCES

1. US NRC, 2014. TRAC/RELAP Advanced Computational Engine (TRACE) V5.840 USER'S MANUAL Volume 1: Input Specification. Tech. Rep. Division of Safety Analysis, Office of Nuclear Regulatory Research, U. S. Nuclear Regulatory Commission,

2. Salko R.K. and Avramova M., 2014. CTF theory manual. Tech. Rep. The Pennsylvania State University,

3. Krepper E., Končar B., Egorov Y., 2007. CFD modelling of subcooled boiling—concept, validation and application to fuel assembly design. Nucl. Eng. Des. 237(7), 716-731.

4. Sugrue R., Magolan B., Lubchenko N., Baglietto E., 2017. Assessment of a simplified set of momentum closure relations for low volume fraction regimes in STAR-CCM and OpenFOAM. Ann. Nucl. Energy. 110, 79-87.

5. Mimouni S., Baudry C., Guingo M., Lavieville J., Merigoux N., Mechitoua N., 2016. Computational multi-fluid dynamics predictions of critical heat flux in boiling flow. Nucl. Eng. Des. 299, 28-36.

6. Yadigaroglu G., 2014. CMFD and the critical-heat-flux grand challenge in nuclear thermal–hydraulics–A letter to the Editor of this special issue. Int. J. Multiphase Flow. 67, 3-12.

7. Roy C.J. and Oberkampf W.L., 2011. A comprehensive framework for verification, validation, and uncertainty quantification in scientific computing. Comput. Methods Appl. Mech. Eng. 200(25), 2131-2144.

8. Krepper E. and Rzehak R., 2011. CFD for subcooled flow boiling: Simulation of DEBORA experiments. Nucl. Eng. Des. 241(9), 3851-3866.

9. Rzehak R. and Krepper E., 2013. CFD for subcooled flow boiling: Parametric variations. Sci. Technol. Nucl. Install. 2013

10. Wang Q. and Yao W., 2016. Computation and validation of the interphase force models for bubbly flow. Int. J. Heat Mass Transfer. 98, 799-813.

11. AIAA, 1998. Guide for the Verification and Validation of Computational Fluid Dynamics Simulations. American Institute of Aeronautics and Astronautics, AIAA-G-077-1998.

12. Oberkampf W.L. and Smith B.L., 2017. Assessment Criteria for Computational Fluid Dynamics Model Validation Experiments. Journal of Verification, Validation and Uncertainty Quantification. 2(3), 1-14.

13. Hoang N.H., Song C., Chu I., Euh D., 2017. A bubble dynamics-based model for wall heat flux partitioning during nucleate flow boiling. Int. J. Heat Mass Transfer. 112, 454-464.

14. Basu N., Warrier G.R., Dhir V.K., 2005. Wall heat flux partitioning during subcooled flow boiling: Part 1—model development. Journal of heat Transfer. 127(2), 131-140.

15. Gilman L. and Baglietto E., 2017. A self-consistent, physics-based boiling heat transfer modeling framework for use in computational fluid dynamics. Int. J. Multiphase Flow. 95, 35-53.

16. Kocamustafaogullari G. and Ishii M., 1995. Foundation of the interfacial area transport equation and its closure relations. Int. J. Heat Mass Transfer. 38(3), 481-493.

17. Brooks C.S., Ozar B., Hibiki T., Ishii M., 2014. Interfacial area transport of subcooled boiling flow in a vertical annulus. Nucl. Eng. Des. 268, 152-163.

18. Park H., Lee T., Hibiki T., Baek W., Ishii M., 2007. Modeling of the condensation sink term in an interfacial area transport equation. Int. J. Heat Mass Transfer. 50(25-26), 5041-5053.

19. Hibiki T. and Ishii M., 2000. One-group interfacial area transport of bubbly flows in vertical round tubes. Int. J. Heat Mass Transfer. 43(15), 2711-2726.

20. Sun X., Kim S., Ishii M., Beus S.G., 2004. Modeling of bubble coalescence and disintegration in confined upward two-phase flow. Nucl. Eng. Des. 230(1-3), 3-26.

21. Colombo M. and Fairweather M., 2016. Accuracy of Eulerian–Eulerian, two-fluid CFD boiling models of subcooled boiling flows. Int. J. Heat Mass Transfer. 103, 28-44.

22. Estrada-Perez C.E. and Hassan Y.A., 2010. PTV experiments of subcooled boiling flow through a vertical rectangular channel. Int. J. Multiphase Flow. 36(9), 691-706.

23. Hassan Y.A., Estrada-Perez C.E., Yoo J.S., 2014. Measurement of subcooled flow boiling using Particle Tracking Velocimetry and infrared thermographic technique. Nucl. Eng. Des. 268, 185-190.

24. Estrada-Perez C.E., Yoo J., Hassan Y.A., 2015. Feasibility investigation of experimental visualization techniques to study subcooled boiling flow. Int. J. Multiphase Flow. 73, 17-33.

25. Theofanous T., 2002. The boiling crisis phenomenon Part I: nucleation and nucleate boiling heat transfer. Experimental Thermal and Fluid Science. 26(6-7), 775-792.

26. Theofanous T.G., Dinh T., Tu J.P., Dinh A.T., 2002. The boiling crisis phenomenon: Part II: dryout dynamics and burnout. Exp. Therm. Fluid Sci. 26(6-7), 793-810.

27. Richenderfer A. et al., 2017. The Application of Modern Experimental Techniques to The Study of Subcooled Nucleate Flow Boiling Including Critical Heat Flux. in: Proc. 17th International Topical Meeting on Nuclear Reactor Thermal Hydraulics, Xi'an, China.

28. Feng J. and Bolotnov I.A., 2017. Evaluation of bubble-induced turbulence using direct numerical simulation. Int. J. Multiphase Flow. 93, 92-107.

29. Feng J. and Bolotnov I.A., 2017. Effect of the wall presence on the bubble interfacial forces in a shear flow field. Int. J. Multiphase Flow. 99, 73-85.

30. Sato Y. and Ničeno B., 2013. A sharp-interface phase change model for a mass-conservative interface tracking method. J Comput Phys. 249, 127-161.

31. Sato Y. and Niceno B., 2017. Nucleate pool boiling simulations using the interface tracking method: Boiling regime from discrete bubble to vapor mushroom region. Int. J. Heat Mass Transfer. 105, 505-524.

32. Lewis A., Smith R., Williams B., Figueroa V., 2016. An information theoretic approach to use high-fidelity codes to calibrate low-fidelity codes. J Comput Phys. 324, 24-43.

33. LeCun Y., Bengio Y., Hinton G., 2015. Deep learning. Nature. 521(7553), 436-444.

34. Krizhevsky A., Sutskever I., Hinton G.E., 2012. Imagenet classification with deep convolutional neural networks. in: Proc. 26th Annual Conference on Neural Information Processing Systems

35. Hinton G. et al., 2012. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. IEEE Signal Process. Mag. 29(6), 82-97.

36. Silver D. et al., 2017. Mastering the game of go without human knowledge. Nature. 550(7676), 354-359.

37. Ling J., Kurzawski A., Templeton J., 2016. Reynolds averaged turbulence modelling using deep neural networks with embedded invariance. J. Fluid Mech. 807, 155-166.

38. Kutz J.N., 2017. Deep learning in fluid dynamics. J. Fluid Mech. 814, 1-4.

39. Ma M., Lu J., Tryggvason G., 2016. Using statistical learning to close two-fluid multiphase flow equations for bubbly flows in vertical channels. Int. J. Multiphase Flow. 85, 336-347.

40. Oberkampf W.L. and Roy C.J., 2010. Verification and validation in scientific computing. Cambridge University Press.

41. Oberkampf W.L. and Barone M.F., 2006. Measures of agreement between computation and experiment: validation metrics. Journal of Computational Physics. 217(1), 5-36.

42. Ferson S., Oberkampf W.L., Ginzburg L., 2008. Model validation and predictive capability for the thermal challenge problem. Comput. Methods Appl. Mech. Eng. 197(29), 2408-2430.

43. Ferson S. and Oberkampf W.L., 2009. Validation of imprecise probability models. International Journal of Reliability and Safety. 3(1-3), 3-22.

44. Kennedy M.C. and O'Hagan A., 2001. Bayesian calibration of computer models. J. Royal Stat. Soc: Series B (Statistical Methodology). 63(3), 425-464.

45. Coyle C., O'Hanley H., Phillips B., Buongiorno J., McKrell T., 2013. Effects of hydrophobic surface patterning on boiling heat transfer and critical heat flux of water at atmospheric pressure. in: Proc.

46. Sato Y., Smith B., Niceno B., 2018. An Interface Tracking Method for Pool Boiling from Isolated Bubble Regime to Critical Heat Flux. Book chapter in: Encyclopedia of Two-Phase Heat Transfer and Flow III (Vol.I). World Scientific Pub Co Inc.

47. Drew D.A., 1983. Mathematical modeling of two-phase flow. Annu. Rev. Fluid Mech. 15(1), 261-291.

48. Ren W. and Valentine T., 2015. Nuclear Energy Knowledgebase and Simulation Code Validation System Development and Implementation Plan. Tech. Rep. Oak Ridge National Laboratories, ORNL/TM-2015/420.

49. Sato Y. and Sekoguchi K., 1975. Liquid velocity distribution in two-phase bubble flow. Int. J. Multiphase Flow. 2(1), 79-95.

50. Ishii M. and Zuber N., 1979. Drag coefficient and relative velocity in bubbly, droplet or particulate flows. AIChE J. 25(5), 843-855.

51. Tomiyama A., Kataoka I., Zun I., Sakaguchi T., 1998. Drag coefficients of single bubbles under normal and micro gravity conditions. JSME Int J., Ser. B. 41(2), 472-479.

52. Tomiyama A., 1998. Struggle with computational bubble dynamics. Multiph. Sci. Technol. 10(4), 369-405.

53. Lubchenko N., Magolan B., Sugrue R., Baglietto E., 2018. A more fundamental wall lubrication force from turbulent dispersion regularization for multiphase CFD applications. Int. J. Multiphase Flow. 98, 36-44.

54. Burns A.D., Frank T., Hamill I., Shi J., 2004. The Favre averaged drag model for turbulent dispersion in Eulerian multi-phase flows. in: Proc. 5th international conference on multiphase flow, Yokohama, Japan.

55. Antal S.P., Lahey R.T., Flaherty J.E., 1991. Analysis of phase distribution in fully developed laminar bubbly two-phase flow. Int. J. Multiphase Flow. 17(5), 635-652.

56. Gosman A.D., Lekakou C., Politis S., Issa R.I., Looney M.K., 1992. Multidimensional modeling of turbulent two-phase flows in stirred vessels. AIChE J. 38(12), 1946-1956.

57. Auton T.R., Hunt J., Prud'Homme M., 1988. The force exerted on a body in inviscid unsteady non-uniform rotational flow. J. Fluid Mech. 197, 241-257.

58. Ranz W.E. and Marshall W.R., 1952. Evaporation from drops. Chem. Eng. Prog. 48(3), 141-146.

59. Chen Y.M. and Mayinger F., 1992. Measurement of heat transfer at the phase interface of condensing bubbles. Int. J. Multiphase Flow. 18(6), 877-890.

60. Anglart H., Nylund O., Kurul N., Podowski M.Z., 1997. CFD prediction of flow and phase distribution in fuel assemblies with spacers. Nucl. Eng. Des. 177(1), 215-228.

61. Hibiki T. and Ishii M., 2002. Development of one-group interfacial area transport equation in bubbly flow systems. Int. J. Heat Mass Transfer. 45(11), 2351-2372.

62. Yao W. and Morel C., 2004. Volumetric interfacial area prediction in upward bubbly two-phase flow. Int. J. Heat Mass Transfer. 47(2), 307-328.

63. Nguyen V., Song C., Bae B., Euh D., 2013. Modeling of bubble coalescence and break-up considering turbulent suppression phenomena in bubbly two-phase flow. Int. J. Multiphase Flow. 54, 31-42.

64. Krepper E., Lucas D., Frank T., Prasser H., Zwart P.J., 2008. The inhomogeneous MUSIG model for the simulation of polydispersed flows. Nucl. Eng. Des. 238(7), 1690-1702.

65. Kurul N. and Podowski M.Z., 1991. Multidimensional effects in forced convection subcooled boiling. in: Proc. 9th International Heat Transfer Conference, Jerusalem, Israel.

66. Del Valle V.H. and Kenning D., 1985. Subcooled flow boiling at high heat flux. Int. J. Heat Mass Transfer. 28(10), 1907-1920.

67. Cheung S., Vahaji S., Yeoh G.H., Tu J.Y., 2014. Modeling subcooled flow boiling in vertical channels at low pressures - Part 1: Assessment of empirical correlations. Int. J. Heat Mass Transfer. 75, 736-753.

68. Lemmert M. and Chawla J.M., 1977. Influence of flow velocity on surface boiling heat transfer coefficient. Heat Transfer in Boiling. , 237-247.

69. Wang C.H. and Dhir V.K., 1993. Effect of surface wettability on active nucleation site density during pool boiling of water on a vertical surface. J. Heat Transfer. 115(3), 659-669.

70. Yang S.R. and Kim R.H., 1988. A mathematical model of the pool boiling nucleation site density in terms of the surface characteristics. Int. J. Heat Mass Transfer. 31(6), 1127-1135.

71. Hibiki T. and Ishii M., 2003. Active nucleation site density in boiling systems. Int. J. Heat Mass Transfer. 46(14), 2587-2601.

72. Cole R. and Rohsenow W.M., 1969. Correlation of bubble departure diameters for boiling of saturated liquids. Chem. Eng. Prog. Symp. Ser. 65(92), 211-213.

73. Tolubinsky V.I. and Konstanchuk D.M., 1972. The rate of vapour-bubble growth in boiling of subcooled water. Heat Transfer-Soviet Research. 4(6), 7-12.

74. Kocamustafaogullari G., 1983. Pressure dependence of bubble departure diameter for water. Int. Commun. Heat Mass Transfer. 10(6), 501-509.

75. Zeng L.Z., Klausner J.F., Bernhard D.M., Mei R., 1993. A unified model for the prediction of bubble detachment diameters in boiling systems—II. Flow boiling. Int. J. Heat Mass Transfer. 36(9), 2271-2279.

76. Cole R., 1967. Bubble frequencies and departure volumes at subatmospheric pressures. AIChE J. 13(4), 779-783.

77. Podowski R.M., Drew D.A., Lahey Jr R.T., Podowski M.Z., 1997. A mechanistic model of the ebullition cycle in forced convection subcooled boiling. in: Proc. 8th international topical meeting on nuclear reactor thermal-hydraulics, Kyoto, Japan.

78. Yeoh G.H., Vahaji S., Cheung S., Tu J.Y., 2014. Modeling subcooled flow boiling in vertical channels at low pressures–Part 2: Evaluation of mechanistic approach. Int. J. Heat Mass Transfer. 75, 754-768.

79. Eca L. and Hoekstra M., 2009. Evaluation of numerical error estimation based on grid refinement studies with the method of the manufactured solutions. Comput. Fluids. 38(8), 1580-1591.

80. Stern F., Wilson R.V., Coleman H.W., Paterson E.G., 2001. Comprehensive approach to verification and validation of CFD simulations—part 1: methodology and procedures. J Fluids Eng. 123(4), 793-802.

81. Bui A., Williams B., Dinh N., Nourgaliev R., 2013. Statistical modeling support for calibration of a multiphysics model of subcooled boiling flows. in: Proc. International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering, LaGrange Park, IL.

82. Smith R.C., 2014. Uncertainty quantification: theory, implementation, and applications. SIAM.

83. Hosder S., Watson L.T., Grossman B., Mason W.H., Kim H., Haftka R.T., Cox S.E., 2001. Polynomial response surface approximations for the multidisciplinary design optimization of a high speed civil transport. Optim. Eng. 2(4), 431-452.

84. Wu X., Mui T., Hu G., Meidani H., Kozlowski T., 2017. Inverse uncertainty quantification of TRACE physical model parameters using sparse gird stochastic collocation surrogate model. Nucl. Eng. Des. 319, 185-200.

85. Constantine P.G., Dow E., Wang Q., 2014. Active subspace methods in theory and practice: applications to kriging surfaces. SIAM J. Sci. Comput. 36(4), A1524.

86. Saltelli A., 2002. Making best use of model evaluations to compute sensitivity indices. Comput. Phys. Commun. 145(2), 280-297.

87. Banks H.T., Cintron-Arias A., Kappel F., 2013. Parameter selection methods in inverse problem formulation. Book chapter in: Mathematical modeling and validation in physiology. Springer.

88. Higdon D., Gattiker J., Williams B., Rightley M., 2008. Computer model calibration using high-dimensional output. J. Am. Stat. Assoc. 103(482), 570-583.

89. Liu F., Bayarri M.J., Berger J.O., Paulo R., Sacks J., 2008. A Bayesian analysis of the thermal challenge problem. Comput. Methods Appl. Mech. Eng. 197(29), 2457-2466.

90. Haario H., Laine M., Mira A., Saksman E., 2006. DRAM: efficient adaptive MCMC. Stat Comput. 16(4), 339-354.

91. Weller H.G., Tabor G., Jasak H., Fureby C., 1998. A tensorial approach to computational continuum mechanics using object-oriented techniques. Comput. Phys. 12(6), 620-631.

92. Marelli S. and Sudret B., 2014. UQLab: A framework for uncertainty quantification in Matlab. in: Proc. 2nd International Conference on Vulnerability and Risk Analysis and Management, Liverpool, United Kingdom.

93. Bartolomei G.G. and Chanturiya V.M., 1967. Experimental study of true void fraction when boiling subcooled water in vertical tubes. Thermal Engineering. 14(2), 123-128.

94. Jung S. and Kim H., 2014. An experimental method to simultaneously measure the dynamics and heat transfer associated with a single bubble during nucleate boiling on a horizontal surface. Int. J. Heat Mass Transfer. 73, 365-375.

95. Liu Y. and Dinh N., 2016. Analysis of heat transfer under high heat flux nucleate boiling conditions. Kerntechnik. 81(3), 308-314.

96. Yoo J., Estrada-Perez C.E., Hassan Y.A., 2016. Experimental study on bubble dynamics and wall heat transfer arising from a single nucleation site at subcooled flow boiling conditions–Part 1: Experimental methods and data quality verification. Int. J. Multiphase Flow. 84, 315-324.

97. Bucci M., Richenderfer A., Su G., McKrell T., Buongiorno J., 2016. A mechanistic IR calibration technique for boiling heat transfer investigations. Int. J. Multiphase Flow. 83, 115-127.

98. Jayatilleke C., 1969. The influence of Prandtl number and surface roughness on the resistance of the laminar sublayer to momentum and heat transfer. Prog.Heat Mass Transfer. 1, 193-321.

99. Edeling W.N., Cinnella P., Dwight R.P., Bijl H., 2014. Bayesian estimates of parameter variability in the k-epsilon turbulence model. Journal of Computational Physics. 258, 73-94.

100. Gel A., Shahnam M., Musser J., Subramaniyan A.K., Dietiker J., 2016. Nonintrusive Uncertainty Quantification of Computational Fluid Dynamics Simulations of a Bench-Scale Fluidized-Bed Gasifier. Ind Eng Chem Res. 55(48), 12477-12490.

101. Liu T.J. and Bankoff S.G., 1993. Structure of air-water bubbly flow in a vertical pipe—II. Void fraction, bubble velocity and bubble size distribution. Int. J. Heat Mass Transfer. 36(4), 1061-1072.

102. Leung W.H., Revankar S.T., Ishii Y., Ishii M., 1995. Axial development of interfacial area and void concentration profiles measured by double-sensor probe method. Int. J. Heat Mass Transfer. 38(3), 445-453.

103. Lin C. and Hibiki T., 2014. Databases of interfacial area concentration in gas–liquid two-phase flow. Prog. Nuclear Energy. 74, 91-102.

104. Leung W.H., 1997. Modeling of interfacial area concentration and interfacial momentum transfer: Theoretical and experimental study (PhD Thesis). Purdue University, West Lafayette, IN.

105. Barron A.R., 1993. Universal approximation bounds for superpositions of a sigmoidal function. IEEE Trans. Inf. Theory. 39(3), 930-945.

106. Eldan R. and Shamir O., 2016. The power of depth for feedforward neural networks. in: Proc. Conference on Learning Theory 2016, New-York City, NY.

107. Li Y. and Yuan Y., 2017. Convergence analysis of two-layer neural networks with relu activation. in: Proc. 31st Annual Conference on Neural Information Processing Systems, Los Angeles, CA.

108. Zhang C., Bengio S., Hardt M., Recht B., Vinyals O., 2017. Understanding deep learning requires rethinking generalization. in: Proc. 5th International Conference on Learning Representations, Toulon, France.

109. Nair V. and Hinton G.E., 2010. Rectified linear units improve restricted boltzmann machines. in: Proc. 27th international conference on machine learning, Haifa, Israel.

110. Clevert D., Unterthiner T., Hochreiter S., 2016. Fast and accurate deep network learning by exponential linear units (elus). in: Proc. 4th International Conference on Learning Representations, San Juan, Puerto Rico.

111. Rumelhart D.E., Hinton G.E., Williams R.J., 1986. Learning representations by back-propagating errors. Nature. 323(6088), 533-536.

112. Kingma D.P. and Ba J., 2015. Adam: A method for stochastic optimization. in: Proc. 3rd International Conference on Learning Representations, San Diego, CA.

113. Gaertner R.F., 1965. Photographic study of nucleate pool boiling on a horizontal surface. Journal of Heat Transfer. 87(1), 17-27.

114. Yabuki T. and Nakabeppu O., 2014. Heat transfer mechanisms in isolated bubble boiling of water observed with MEMS sensor. Int. J. Heat Mass Transfer. 76, 286-297.

115. Paszke A. et al., 2017. Automatic differentiation in PyTorch. in: Proc. 31st Annual Conference on Neural Information Processing Systems, Los Angeles, CA.

116. Glorot X. and Bengio Y., 2010. Understanding the difficulty of training deep feedforward neural networks. in: Proc. 13th International Conference on Artificial Intelligence and Statistics, Sardinia, Italy.

117. Zhang Q. and Zhu S., 2018. Visual Interpretability for Deep Learning: a Survey. arXiv preprint arXiv:1802.00614.

118. Strofer C.M., Wu J., Xiao H., Paterson E., 2018. Data-Driven, Physics-Based Feature Extraction from Fluid Flow Fields. arXiv preprint arXiv:1802.00775.

119. Chang C. and Dinh N.T., 2018. Classification of Machine Learning Frameworks for Data-Driven Thermal Fluid Models. arXiv preprint arXiv:1801.06621.