A Study on Epistemic Uncertainty Estimation in Reliability Models

Dissertation submitted in partial fulfillment for the
degree of Ph.D. of Advanced Science and Engineering

Jiahao Zhang

Under the supervision of
Professor Hiroyuki Okamura

Dependable Systems Laboratory,
Graduate School of Advanced Science and Engineering,
Hiroshima University, Higashi-Hiroshima, Japan

January 2023

Abstract

This thesis discusses the epistemic uncertainty estimation in reliability models such as fault trees, Markov models and their hybrid models. The epistemic uncertainty is the uncertainty of output of model propagated from the uncertainty of input parameters. The uncertainty of input parameters are caused by the statistical errors in estimating model parameters due to limited sample size.

In Chapter 2, we develop the moment-based approach for estimating the epistemic uncertainty in hierarchical reliability models. The main point of the epistemic uncertainty estimation is how to determine the distribution of output of model. There are two approaches to obtain the distribution of output. One approach is to get a closed form solution of the distribution based on mathematical analysis. However, it is not always that the solution has a closed form. Another approach is based on sampling. By collecting outputs of model with the changes of input parameters, we obtain the distribution of output of model numerically. But, such approaches require much computation cost, i.e., the computation time tends to be longer when we want to get the highly-accurate distribution. The moment-based approach for the epistemic uncertainty estimation is known as the method to make a balance between computation speed and accuracy. Since the moment-based needs the information on local sensitivity, i.e., the first two derivatives of output of model, it has been discussed in a monolithic Markov chain whose first two derivatives are computed easily. This thesis extends the applicability of moment-based approach to the hierarchical reliability model. The hierarchical reliability model is defined as a hybrid model with fault trees and Markov chains, and they are used in reliability evaluation for complex system. The main idea behind our approach is to use the automatic differentiation for BDD (binary decision diagram) representation of fault trees and Markov chains. Numerical experiments are exhibited to estimate the uncertainty propagation in both simple and complicated hierarchical models. By comparing them with Bayes approach, we discuss the accuracy of moment-based approximation.

In Chapter 3, we focus on the global sensitivity of epistemic uncertainty. Although the local sensitivity is to estimate the magnitude of variation when the input parameters are varied, the global sensitivity is an approach to reveal

the main effect to produce the uncertainty. The well-known global sensitivity is the variance-based sensitivity analysis that is similar to the variance analysis in statistical models. Our main idea is to apply the moment-based approach to obtain the variance-based sensitivity measure in Markov models. The presented approach is much faster than the existing approach using Monte Carlo (MC) simulation.

In Chapter 4, we discuss the computation method of information matrix for phase-type (PH) approximation. The PH approximation is applied in non-exponential models that involve the state transitions following non-exponential distributions, and is to approximate the original model by replacing their non-exponential distributions with PH distributions. Since PH distributions are defined by Markov chains, the approximated model is regarded as a Markov model. In this scheme, we need to determine the model parameters of PH distributions that fits to the non-exponential distribution. Since this is based on the statistical estimation, the determined parameters of PH distribution involve the uncertainty. In the epistemic uncertainty estimation, the uncertainty of input parameters are estimated beforehand. In the context of PH approximation, the uncertainty of PH parameters should also be estimated. One of the approach to estimate the uncertainty is the evaluation of information matrix. By computing the information matrix with the estimated parameters, we obtain the characteristics of uncertainty of parameters. However, the computation of information matrix requires much computation effort. Especially, since PH distributions have a lot of parameters, it is necessary to develop an effective computation approach on information matrix. In Chapter 4, we propose an effective algorithm for computing the information matrix of PH distribution based on the uniformization technique.

Finally, we summary the contributions of this thesis are; (i) enhancement of the moment-based approach for estimating epistemic uncertainty so that it can be applied to in hierarchical models with the automatic differential technique. (ii) application of moment-based approach to obtain the global sensitivity measure of epistemic uncertainty with the variance-based sensitivity analysis. (iii) development of an effective algorithm for computing the information matrix of PH distributions that is useful to estimate the uncertainty of input parameters.

Acknowledgements

First and foremost, I would like to extend my sincere gratitude to Professor Hiroyuku Okamura, the supervisor of my study, for his valuable guidance, kind advice in every stage of the writing of this thesis. Without his continuous encouragement and impressive patience, I could not have completed my thesis.

Also, many thanks go to Professor Tadashi Dohi, Professor Yasuhiko Morimoto, Professor Shaoying Liu and Professor Tadashi Shima for their invaluable comments, useful suggestions and warm encouragement.

Finally, it is my special pleasure to acknowledge the hospitality and encouragement of the past and present members of the Dependable Systems Laboratory, Graduate School of Advanced Science and Engineering, Hiroshima University. I also feel gratitude for my family's support and my wife's accompany and encouragement for my Ph.D. graduation.

Contents

Chapter 1

Introduction

Model-based dependability evaluation is to estimate quantitative dependability measures such as system reliability and availability by using stochastic models and is quite useful for determining the system configuration in the design phase. It is well-known that the Markov chain and fault tree (FT) play central roles in the model-based dependability evaluation [1]. The drawback of model-based dependability evaluation is to determine model parameters, because the model parameters are usually estimated from empirical data or the knowledge of experts. However, since the estimated model parameters themselves involve uncertainty, it is important to evaluate how much effect the uncertainty of model parameters is on the dependability measures, which is called uncertainty propagation. More specifically, the uncertainty propagation is categorized into aleatory uncertainty and epistemic uncertainty [2]. The aleatory uncertainty generally results from the effect of inherent randomness or unpredictable variability of the modeled phenomenon. On the other hand, the epistemic uncertainty is due to the imperfect knowledge or incomplete information regarding values of parameters of the underlying model [3]. There are several papers to achieve the estimation of uncertainty propagation [4–8,10]. Singpurwalla *et al* proposed Bayes estimation for handling epistemic uncertainty, but the requirement of closed-form equation makes it only suitable for single structure model. Mishra *et al.* [7] considered the sampling-based approach for epistemic uncertainty evaluation, taking advantage of analysis without closed-form structure. However, the high computational cost generated by simulation method should be further considered. In particular, Yin *et al.* [5] introduced an approximation using Taylor series expansion of epistemic

uncertainty propagation for reliability models. Okamura *et al.* [10] presented a moment-based approximation approach using the Taylor series expansion for the uncertainty propagation when the model is given by general Markov chains. The moment-based approximation requires the first and second derivatives of output measures of system respect to model parameters and can well address the both closed-form and non-closed form situations [10] with efficiency. However, the real systems are usually composed of complex structure, making it difficult to evaluate the effects of uncertainty propagation. Since the traditional method can not well be applied on complex system, we focus on the moment-based approximation for epistemic uncertainty propagation and improve the efficiency of computing the first and second derivatives of complex system for this method.

In model-based dependability evaluation, the structure of complex system is usually displayed through dependability models, which are often described as hierarchical models [11, 15]. The hierarchical modeling is to combine FTs and Markov chains, such as continuous-time Markov chains (CTMCs). The FT represents the relationship between system failure and component failures, and the Markov chain represents the dynamic behavior of components. Compared with the monolithic Markov model representing the behavior of whole systems, the hierarchical model has an advantage in reducing the number of system states. In this dissertation, we first focus on the epistemic uncertainty propagation for a hierarchical model consisting of FTs and CTMCs, and discuss a moment-based approximation method to estimate the uncertainty propagation in the hierarchical model. Furthermore, we also discussed this topic in another paper which is under review [12].

Secondly, stochastic models including Markov chains can be highly complex, as a result, the relationship between input parameters and output measure might be poorly understood. Because the model parameters are usually estimated from empirical data or the knowledge of experts, that means the value of model parameters contains uncertainty and even affects the output measures. Due to the poor relationship between the inputs and outputs, it is difficult to realize the uncertainty of output affected by input parameters. In such a situation, it is not clear which component needs to be prioritized if we want to improve system dependability. To address the above issues, sensitivity

analysis, which acts as a critical role in quantifying uncertainty in the design of computer systems, has been widely considered. In particular, a variance-based global sensitivity analysis is often used to rank the importance of input factors, based on their contribution to the variance of the output measure of interest [13]. The variance-based sensitivity analysis relies on the computation of conditional variances, is sampling-based, and therefore usually applies simulation methods such as Monte Carlo simulation. That means, the traditional methods for variance-based sensitivity analysis based on simulation do not need the analytic structure of the model to be analyzed, which can be regarded as its major advantage. However, the simulation usually needs a huge number of realizations until obtaining stable results, resulting in an undesired high computational cost. Gan *et al.* [14] applied 10 different sensitivity analysis methods on the Sacramento Soil Moisture Accounting (SAC-SMA) model. The cost of methods is calculated by the number of samples, and most methods need more than 1000 samples to achieve the same estimation purpose. To overcome this problem, we consider the moment-based approximation, which can obtain the expectation and variance of the output measure to achieve the formulation of the variance-based sensitivity analysis.

On the other hand, epistemic uncertainty propagation occurs mainly from the parameter estimation which follows non-exponential distribution and exponential distribution [2]. Since non-exponential distribution is difficult to compute, a widely used solution is applying phase-type (PH) distribution for approximation. PH distribution is the distribution for an absorbing time in a CTMC, and it is the fundamental part of PH expansion. In general, PH expansion [44] is widely used to solve the problem that non-exponential distribution is difficult to calculate. The main idea of PH expansion is replacing the non-exponential distribution with PH distribution with high accuracy. And the calculation problem will turn to the calculation of the underlying CTMC of PH distribution. In PH expansion, PH fitting [45] is considered to fit the PH parameters from the target distribution. In general, PH parameters are estimated from (i) independent and identically distributed (IID) samples or (ii) weighted samples from the non-exponential distribution via the maximum likelihood (ML) estimation [46]. With PH fitting, the fitted PH distribution can be a very accurate approxi-

mation of the original distribution. After expansion, the non-Markovian model containing non-exponential distributions becomes PH expanded model, which is essentially a CTMC. The PH expanded model can be used for evaluating system performance/dependability measures, such as system reliability and availability. However, model parameters typically involve estimation errors, so-called uncertainty, especially in the phase expanded models where the number of PH parameters and states of underlying CTMC grows significantly as an increasing phase [47]. It results in estimation error in the performance/dependability measure, so it is crucial to analyze the parameter uncertainty [10, 38]. To solve the above issue, one can use the Fisher information matrix to obtain the variance and covariance information of model parameters [48]. In the context of ML estimation [49], the Fisher information matrix is given by the second derivatives of log-likelihood function (LLF) in terms of parameters. The variance/covariance matrix is computed by the inverse of the Fisher information matrix. It is easy to obtain the Fisher information matrix when the model contains few parameters. But as we mentioned, PH fitting can bring a lot of parameters so that the calculation speed would be so low, and the computational cost would be extremely high. So, the motivation of this paper is to improve the calculation speed of the Fisher information matrix with a large number of model parameters.

The organization of this dissertation is as follows. The moment-based approximation for epistemic Uncertainty propagation in hierarchical reliability models is shown in Chapter 2. Chapter 3 introduces the details of variance-based sensitivity analysis for Markov models and the experiments of them. Chapter 4 describes the contents of the efficient computation of information matrix in phase-type fitting. Chapter 5 gives the conclusions and the future work.

Chapter 2

Moment-Based Approach for Estimating Epistemic Uncertainty

2.1 Introduction

It is well-known that model-based approach such as FT, Markov chain, Hierarchical model plays the center role in dependability evaluation. However, the usage of model-based approach requires the estimation of model parameters, which is usually derived from data samples and expert's knowledge. Due to the statistical errors called uncertainty in the model parameters, the output measure of system can be affected. So it is important to evaluate how much effect the uncertainty of model parameters is on the dependability measures, which is called uncertainty propagation.

In this chapter, we focus on the epistemic uncertainty propagation for a hierarchical model consisting of FTs and CTMCs, and discuss a moment-based approximation method to estimate the uncertainty propagation in the hierarchical model. The moment-based approximation requires the first and second derivatives of output measures concerning model parameters, but it is not easy to obtain them for an FT in practice. [38]. To address this issue, we present the binary decision diagram (BDD) approach for the computation of the first and second derivatives in FT and extend this method to hierarchical models. The presented approach is scalable with respect to the number of parameters. The main contributions of this chapter are summarized as follows:

- Proposal of a feasible approach for epistemic uncertainty propagation in hierarchical reliability models by using moment-based approximation method.

- BDD-based computation for the first and second derivatives of the FTs and hierarchical models,

- Validation and evaluation of the proposed approach on both simple and complicated hierarchical models.

2.2 Related Work

In reliability engineering, probabilistic models are used to evaluate the quantitative measures of a system, such as system reliability and availability, since the failure event involves randomness. In such a probabilistic model, the randomness is generally modeled by parametric distributions and is called *aleatory uncertainty*. These probabilistic models are represented with fixed parameter values of these aleatory distributions, and the outputs thus obtained clearly depend upon the values of the parameters used.

The parameters of the aleatory model are estimated from experimental data such as field failure data, maintenance logs, and other sources of observed data. Since the number of observations is finite, the sampling errors affect the estimated model parameter values. Also, even if the parameter values are given by expert guesses, the uncertainty may be included in the values. This parametric uncertainty arising out of incomplete information about the parameters is called *epistemic uncertainty* [2].

In the context of uncertainty analysis, Lei *et al.* [4] proposed a strategy for modeling uncertainty propagation. Compared with the typical uncertainty propagation methods such as first-order analysis, sensitivity analysis, statistical linearization, and Monte Carlo (MC) analysis, they introduced a pathway of parameter uncertainty propagation analysis based on validity, simplicity, and computational requirements. Jean *et al.* [6] considered an MC method to evaluate properly the influence of residual parameters with prior uncertainty on the covariance of fitting parameters. Their work showed that propagating the uncertainties is crucial to understanding the parameter relative influences on

Table 2.1: Epistemic uncertainty propagation for different aleatory model types

	Epistemic uncertainty propagation method			
Solutions	**Closed-form integration**	**Numerical integration**	**Moment-based**	**Sampling-based**
Closed-form solution	Applicable (simple expressions)	Applicable (a few parameters)	Applicable	Applicable
Analytic-numeric solution	Not applicable	Applicable (a few parameters)	Applicable	Applicable
Simulative solution	Not applicable	Applicable (a few parameters)	Applicable (numerical differentiation is needed)	Applicable

theoretical model calculations.

The natural way to handle epistemic uncertainty is to treat the parameters of the aleatory probabilistic model as random variables (r.v.s). This concept is closely related to Bayes theorem [16] and leads to some difficulties in terms of numerical computation. Mishra *et al.* [7, 8] classified the epistemic uncertainty evaluation techniques into three categories; analytic closed-form integration, numerical integration, and sampling-based method. Table 2.1 summarizes the applicability of various epistemic uncertainty propagation methods for different types of aleatory models. Specifically, the analytic closed-form integration can be applied in the case of aleatory models that can be analytically solved to get the model output as simple closed-form expressions of input parameters. Zhang*et al.* [9] considered the epistemic uncertainty propagation in fault tree, discussing the effects of uncertainty on the importance measure. For more complex expressions of model output, i.e., in the case where the value of output metric is evaluated numerically with software packages like SHARPE [17] or SPNP [18], numerical integration and sampling-based approximation [19, 20] are applicable. In [5], the moment-based approach for a simple reliability model was presented. Okamura *et al.* [10] extended the moment-based uncertainty computation to general CTMC models. Also, Harverkort *et al.* [21] discussed the uncertainty propagation with the moments of parameter distribution for general Markov-reward models. However, in the work by Harverkort *et al.* [21], the moment-based approach could not be applied to a complex Markov-reward model due to the limitation of computational power at that time.

Other analytic methods for parametric epistemic uncertainty propagation,

mostly based on algebraic manipulations of model output and exploiting properties and transformations of expectation and variance (for simple non-state space reliability models), have been studied by Sarkar [22], Leiberman and Ross [23], and Coit [24]. Several recent papers have applied these ideas to high-speed railways reliability models [25], power consumption models [26], and system on a chip [27]. Particularly, the paper considering the system on a chip also extended the analysis to the case of Weibull aleatory distributions.

2.3 Hierarchical Reliability Model

A hierarchical reliability model is a mixture of non-state-space models and state-space models. The non-state-space model is a combinatorial model representing a static relationship between component failures and system failure. The typical non-state-space model is the FT. On the other hand, the state-space model represents a dynamic behavior of the system state and a history of state transitions when the system has failed. The typical state-space model is the Markov chain.

In general, the representation of the state-space model covers one of the non-state-space models mathematically; thus, the state-space model may be enough to represent the system failure mechanism. However, when we model a complex system with many components, the state-space representation causes the state explosion. For example, for the system consisting of n components, each of which is modeled by the binary states; UP (the component works normally) and DOWN (the component is failed), the number of system states is 2^n and increases exponentially as the number of components increases.

The hierarchical reliability model mitigates such a state explosion problem in reliability evaluation. Concretely, the hierarchical reliability model is constructed by a non-state-space model, where a state-space model models the dynamics of each component. In this chapter, we focus on the typical hierarchical reliability model, composed of an FT at the system level and CTMCs at the component level.

2.3.1 Fault Tree

The system failure is modeled by an FT with n basic events. Each basic event indicates a probabilistic event occurrence related to the system failure, e.g.,

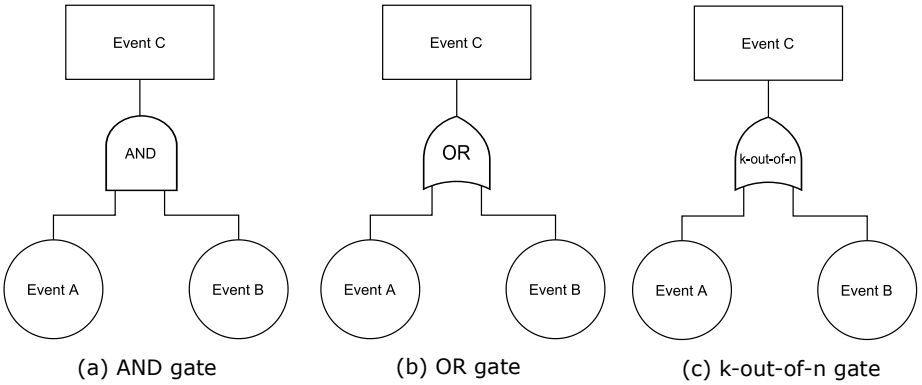

Figure 2.1: Example of Fault-tree.

component failure. Let X_i, $i = 1, \ldots, n$ be indicator r.v.s representing whether a respective event happens or not. The basic events are connected by logical gates and form a tree whose root (parent) event becomes a composite event triggered by basic events. Then the gate gives a condition to the root event occurrence. Typically, AND, OR, and k-out-of-n gates are widely-used in reliability assessment. For example, Fig. 2.1(a) shows an AND gate, which represents the relationship of Event C when Event A and Event B have occurred. Figure 2.1(b) is an OR gate, indicating that Event C occurs when at least one of Event A and Event B occurs. Also, Fig. 2.1(c) represents a 2-out-of-3 gate, where the root event occurs only when two or more basic events occur. In fact, the FT model has a hierarchical structure of AND/OR/k-out-of-n gates to describe the conditions of the top event occurrence.

From the mathematical point of view, an FT model can be represented by a structure function. The structure function is a 0-1 value function with an output and n inputs. The output means the state of top event occurrence, and the inputs are the states of basic events. Let x_i, $i = 1 \ldots, n$, and y be the binary states to represent n basic events and the top event, respectively, we have

$$y = F(x_1, \ldots, x_n). \tag{2.1}$$

In the above $F(\cdot)$ represents the structure function. For example, the structure functions of Figs. 2.1(a) and (b) are given by

$$y = x_A x_B, \tag{2.2}$$

$$y = 1 - (1 - x_A)(1 - x_B). \tag{2.3}$$

Using $F(\cdot)$, the top-event occurrence probability p_s is expressed as follows.

$$p_s = F(p_1, p_2, \cdots, p_n), \tag{2.4}$$

where p_i is a probability that the i-th basic event occurs.

2.3.2 Markov Chain

Since FT is a static model to express the relationship between the top event and basic events, it cannot basically represent the dynamic behavior, such as an order of basic event occurrences. For example, in Fig. 2.1(a), there is a case where the root event C occurs only when an order of basic event occurrences; A $\rightarrow$ B. To express such a dynamic behavior, the dynamic FT (DFT) has been presented [28]. The DFT is an extension of FT by introducing several dynamic gates, such as priority AND gate, which can indicate the order of basic event occurrences causing the root event. However, the representation ability of DFT is restricted, and there is the dynamic behavior that any DFT cannot express. In the chapter, we consider using the Markov chain to represent the system's dynamic behavior, i.e., the hierarchical model whose top level is given by a static FT, and the basic event occurrence follows a Markov chain.

A Markov chain is a stochastic process with discrete state space on either a discrete-time or continuous-time domain. The hierarchical model generally deals with discrete-time and continuous-time Markov chains. This chapter focuses only on the CTMC, i.e., the system behaves in the continuous-time domain. The same approach can be applied to the case of a discrete-time Markov chain.

Suppose that the state space associated with i-th basic event is defined by the discrete state; $\mathcal{S}_i$. In this chapter, these states are called phases. Besides, all the phases can be divided into two sets $\mathcal{U}_i$ and $\mathcal{D}_i$ where $\mathcal{U}_i \cap \mathcal{D}_i = \phi$ and $\mathcal{U}_i \cup \mathcal{D}_i = \mathcal{S}_i$. When the current phase is in $\mathcal{U}_i$, the state of i-th basic event is 0, i.e., the component i is working. Also, the phase is in $\mathcal{D}_i$ means the state of i-th basic event is 1, i.e., the component i is failed. Note that if both the sizes of phases $\mathcal{U}_i$ and $\mathcal{D}_i$ are 1, the hierarchical model is an ordinary FT model. Let $\{X_i(t); t \geq 0\}$ be the CTMC process on $\mathcal{S}_i$ and $\pi_n(t) = P(X_i(t) = n)$ denotes the state probability vector that the current phase is $n \in \mathcal{S}_i$ at time t. Then

the probability that i-th basic event occurs is given by

$$p_i = \sum_{n \in \mathcal{D}_i} \pi_n(t). \tag{2.5}$$

Define $\boldsymbol{Q}_i$ as an infinitesimal generator of $X_i(t)$. We have the following difference-differential equations (alternatively, Kolmogorov forward equation):

$$\frac{d}{dt}\boldsymbol{\pi}_i(t) = \boldsymbol{\pi}_i(t)\boldsymbol{Q}_i, \tag{2.6}$$

where $\boldsymbol{\pi}_i(t)$ is a row vector whose entries are $\pi_n(t)$, $n \in \mathcal{S}_i$. Similarly, we sometimes encounter the case where the basic event occurrence probability is given from the steady-state probability vector of CTMC, such as steady-state availability. In such a case, the i-th basic event occurrence probability is written as

$$p_i = \sum_{n \in \mathcal{D}_i} \tilde{\pi}_n, \tag{2.7}$$

where $\tilde{\pi}_n$ is the n-th entry of steady-state probability vector $\tilde{\boldsymbol{\pi}}$ satisfying

$$\tilde{\boldsymbol{\pi}}_i \boldsymbol{Q}_i = \boldsymbol{0}, \quad \sum_{n \in \mathcal{S}_i} \tilde{\pi}_n = 1. \tag{2.8}$$

As an example of the hierarchical model, we consider the equivalent expression to the DFT. Figure 2.2 shows a DFT that uses a priority AND gate. The system consists of three non-repairable components; A, B, and C. Each component has two states; UP and DOWN. The system failure occurs only when components A and B fail in the order A $\rightarrow$ B or the failure of component C occurs. In the hierarchical model expression, we consider subsystem A$'$ composites the components A and B. The phases (i.e., states) to be considered for the subsystem A$'$ are given in Table 2.2. The corresponding infinitesimal generator matrices becomes

$$\boldsymbol{Q}_{A'} = \begin{pmatrix} -(\lambda_A + \lambda_B) & \lambda_A & \lambda_B & 0 \\ 0 & -\lambda_B & 0 & \lambda_B \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \tag{2.9}$$

$$\boldsymbol{Q}_C = \begin{pmatrix} -\lambda_C & \lambda_C \\ 0 & 0 \end{pmatrix}. \tag{2.10}$$

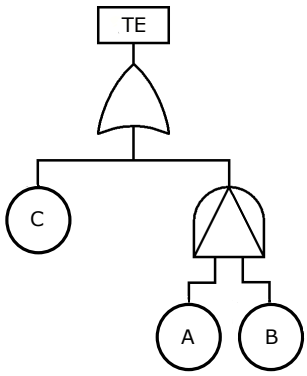

Figure 2.2: PAND gate DFT.

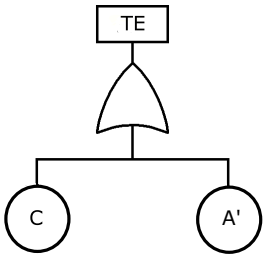

Figure 2.3: PAND gate DFT.

Table 2.2: Phases of the subsystem A′ compositing the components A and B

Phase	Description
(UP, UP)	Both components A and B are operational.
(UP, DOWN)	Component A is operational and B is failed.
(DOWN, UP)	Component A is failed and B is operational.
(DOWN, DOWN)	Both components A and B are failed.

In such a case, the DFT reduces to the FT in Fig. 2.3, whose basic events are represented by CTMCs in Eqs. (2.9) and (2.10).

Generally, we suppose a hierarchical model consisting of a static FT with n basic events and corresponding n CTMC models. Let $\boldsymbol{\theta}_i$ be a parameter vector of the i-th CTMC. Note that $\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_n$ are not necessarily mutually disjointed sets and may have common parameters. Also, we denote $p_i(\boldsymbol{\theta})$ as the event occurrence probability of the i-th basic event because it depends on the model parameter of the corresponding CTMC. Then the top-event occurrence

probability is given by

$$p_s(\boldsymbol{\theta}_s) = F(p_1(\boldsymbol{\theta}_1), p_2(\boldsymbol{\theta}_2), \ldots, p_n(\boldsymbol{\theta}_n)), \tag{2.11}$$

where $\boldsymbol{\theta}_s$ is a parameter vector of hierarchical model; that is, $\boldsymbol{\theta}_s = \cup_{i=1}^{n} \boldsymbol{\theta}_i$.

2.4 Moment-based Epistemic Uncertainty Evaluation

2.4.1 Epistemic Uncertainty Propagation

The epistemic uncertainty propagation is the phenomenon that the (statistical) errors of input parameters affect the output measure. Let $M(\boldsymbol{\theta})$ be an output measure of an aleatory probabilistic model, where $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_l)$ is a (column) vector of input parameters needed to compute the output measure. In the epistemic uncertainty propagation, the model parameters are assumed randomly distributed, i.e., the parameter vector is defined as a vector of random variables $\boldsymbol{\Theta} = (\Theta_1, \ldots, \Theta_l)$. When the joint epistemic density $f_{\boldsymbol{\Theta}}(\boldsymbol{\theta})$ of parameters is given, the cumulative distribution function (c.d.f.) of output measure is given by a l-dimensional integration:

$$F_M(m) = \int I(M(\boldsymbol{\Theta}) \leq m) f_{\boldsymbol{\Theta}}(\boldsymbol{\theta}) d\boldsymbol{\theta}, \tag{2.12}$$

where $I(E)$ is the indicator variable of an event E. Meanwhile, the integration in the above equation is generally the multiple integrations with respect to the parameter vector. Thus, the unconditional expected value of $M(\boldsymbol{\Theta})$ can be computed through the following equation:

$$\mathrm{E}[M(\boldsymbol{\Theta})] = \int M(\boldsymbol{\theta}) f_{\boldsymbol{\Theta}}(\boldsymbol{\theta}) d\boldsymbol{\theta}. \tag{2.13}$$

And the second moment of $M(\boldsymbol{\Theta})$ is given by

$$\mathrm{E}[M(\boldsymbol{\Theta})^2] = \int M(\boldsymbol{\theta})^2 f_{\boldsymbol{\Theta}}(\boldsymbol{\theta}) d\boldsymbol{\theta}. \tag{2.14}$$

In [16], the unconditional expected value of reliability has been termed as *survivability*.

There are two key issues in the epistemic uncertainty propagation: (i) how to determine the joint epistemic density $f_{\boldsymbol{\Theta}}(\boldsymbol{\theta})$, and (ii) how to compute the multiple integrations. The former issue is essentially a statistical problem; thus

well-known statistical approaches are available to determine $f_{\boldsymbol{\Theta}}(\boldsymbol{\theta})$. For example, denote $\mathcal{D}$ as observed data to estimate the model parameters. In the Bayes context, the joint epistemic density can be estimated as follows.

$$f_{\boldsymbol{\Theta}}(\boldsymbol{\theta}|\mathcal{D}) = \frac{p(\mathcal{D}|\boldsymbol{\theta}) f_{\boldsymbol{\Theta}}(\boldsymbol{\theta})}{\int p(\mathcal{D}|\boldsymbol{\theta}) f_{\boldsymbol{\Theta}}(\boldsymbol{\theta}) d\boldsymbol{\theta}}, \tag{2.15}$$

where $p(\mathcal{D}|\boldsymbol{\theta})$ is a probability density or mass function of $\mathcal{D}$ for a given parameter vector $\boldsymbol{\theta}$, which corresponds to a likelihood function of $\boldsymbol{\theta}$ for the given data $\mathcal{D}$. The densities $f_{\boldsymbol{\Theta}}(\boldsymbol{\theta}|\mathcal{D})$ and $f_{\boldsymbol{\Theta}}(\boldsymbol{\theta})$ are called posterior and prior, respectively. Since the posterior density is updated by using the observed data, $f_{\boldsymbol{\Theta}}(\boldsymbol{\theta}|\mathcal{D})$ is regarded as the estimated joint epistemic density. However, it is not common for analysts to know the observed data $\mathcal{D}$, and in general, the available information on model parameters may be a confidence interval. That is, the joint epistemic density should be determined from such incomplete information. In [7], based on the Bayes theorem, Mishra and Trivedi assumed that the posterior formed Erlang or beta densities and determined the parameters of these densities from the confidence intervals. Also, in [29], another approach to uncertainty propagation was proposed when a characteristic function gives the parameter distribution. On the other hand, there is also the case where some statistics, such as point estimates and their confidence interval, are available only instead of observed data. For such a case, it is difficult to obtain the joint density $f_{\boldsymbol{\Theta}}(\boldsymbol{\theta})$ itself.

The latter issue is a computation of multiple integrations. The methods to obtain a value of multiple integrations are classified into analytic and numerical methods. The analytic method is to obtain the integration symbolically and is generally applied to specific problems only. The work in [7] presented several instances where the integration can be solved analytically. In the numerical method, the sampling-based approximation approach, i.e., Monte Carlo integration, is well-known to be effective for solving such multiple integrations. Let $(\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_n)$ be a set of samples drawn from the joint epistemic density $f_{\boldsymbol{\Theta}}(\boldsymbol{\theta})$. The Monte Carlo integration is based on the following approximation:

$$\int M(\boldsymbol{\theta}) f_{\boldsymbol{\Theta}}(\boldsymbol{\theta}) d\boldsymbol{\theta} \approx \frac{1}{n} \sum_{i=1}^{n} M(\boldsymbol{\theta}_i). \tag{2.16}$$

The variants of Monte Carlo integration are the quasi-Monte Carlo method [30], the Markov chain Monte Carlo (MCMC) method [31], the Latin hypercube

sampling [32], and other variance reduction methods. In particular, the MCMC can combine the Bayes estimation of joint epistemic density and the computation of multiple integrations. The survey of sampling-based methods was presented by [33]. It should be noted that the simulation-based approach can be applied to the situation where the joint density $f_{\boldsymbol{\Theta}}(\boldsymbol{\theta})$ can be obtained.

2.4.2 Moment-based Approach

This chapter focuses on the moment-based approach for epistemic uncertainty evaluation in [10]. The main idea behind the approach is to use the first two moments of estimates of model parameters instead of the joint density of model parameters.

Suppose that the column vector of the point estimate of model parameters $\hat{\boldsymbol{\theta}} = (\hat{\theta}_1, \ldots, \hat{\theta}_m)^{\mathrm{T}}$ holds the unbiasedness $\hat{\boldsymbol{\theta}} = \mathrm{E}[\boldsymbol{\Theta}]$. By taking a Taylor series expansion of the expected value of the output measure at $\hat{\boldsymbol{\theta}}$, we have

$$\begin{aligned} \mathrm{E}[M(\boldsymbol{\Theta})] &= M(\hat{\boldsymbol{\theta}}) + \mathrm{E}[M'(\hat{\boldsymbol{\theta}})^{\mathrm{T}}(\boldsymbol{\Theta} - \hat{\boldsymbol{\theta}})] \\ &\quad + \frac{1}{2}\mathrm{E}[(\boldsymbol{\Theta} - \hat{\boldsymbol{\theta}})^{\mathrm{T}} M''(\hat{\boldsymbol{\theta}})(\boldsymbol{\Theta} - \boldsymbol{\theta})] + \cdots, \end{aligned} \tag{2.17}$$

where

$$\begin{aligned} M'(\boldsymbol{\theta}) &= \left.\frac{\partial M(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\right|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}} \\ &= \left(\left.\frac{\partial M(\boldsymbol{\theta})}{\partial \theta_1}\right|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}} \quad \cdots \quad \left.\frac{\partial M(\boldsymbol{\theta})}{\partial \theta_m}\right|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}} \right)^{\mathrm{T}}, \end{aligned} \tag{2.18}$$

and

$$\begin{aligned} M''(\boldsymbol{\theta}) &= \left.\frac{\partial^2 M(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}^2}\right|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}} \\ &= \begin{pmatrix} \left.\frac{\partial^2 M(\boldsymbol{\theta})}{\partial \theta_1^2}\right|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}} & \cdots & \left.\frac{\partial^2 M(\boldsymbol{\theta})}{\partial \theta_1 \partial \theta_m}\right|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}} \\ \vdots & \ddots & \vdots \\ \left.\frac{\partial^2 M(\boldsymbol{\theta})}{\partial \theta_m \partial \theta_1}\right|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}} & \cdots & \left.\frac{\partial^2 M(\boldsymbol{\theta})}{\partial \theta_m^2}\right|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}} \end{pmatrix}. \end{aligned} \tag{2.19}$$

In the above, $M'(\boldsymbol{\theta})$ and $M''(\boldsymbol{\theta})$ are the first and second derivatives of output measure with respect to model parameters, i.e., they mean the parametric local sensitivity. Finally, we obtain the approximation by considering the first two

moments of the above:

$$\begin{aligned}\mathrm{E}[M(\boldsymbol{\Theta})] &\approx M(\hat{\boldsymbol{\theta}}) + \frac{1}{2}\mathrm{E}[(\boldsymbol{\Theta}-\hat{\boldsymbol{\theta}})^{\mathrm{T}} M''(\hat{\boldsymbol{\theta}})(\boldsymbol{\Theta}-\boldsymbol{\theta})] \\ &= M(\hat{\boldsymbol{\theta}}) + \frac{1}{2}(\sum_{i=1}^{m} M''_{i,i}(\hat{\boldsymbol{\theta}})\mathrm{Var}[\Theta_i] \\ &+ 2\sum_{i=1}^{m}\sum_{j=1}^{i-1} M''_{i,j}(\hat{\boldsymbol{\theta}})\mathrm{Cov}[\Theta_i,\Theta_j]), \end{aligned} \tag{2.20}$$

where $M''_{i,j}(\hat{\boldsymbol{\theta}})$ is the (i,j)-entry of $M''(\hat{\boldsymbol{\theta}})$.

If $\Theta_1,\ldots,\Theta_m$ are mutually independent, then $\mathrm{Cov}[\Theta_i,\Theta_j]) = 0$ for $i \neq j$, we have

$$\mathrm{E}[M(\boldsymbol{\Theta})] \approx M(\hat{\boldsymbol{\theta}}) + \frac{1}{2}\sum_{i=1}^{n} M''_{i,i}(\hat{\boldsymbol{\theta}})\mathrm{Var}[\Theta_i], \tag{2.21}$$

and similarly, taking a Taylor series expansion of $M(\Theta)^2$ at $\hat{\boldsymbol{\theta}}$ leads to

$$\begin{aligned}&\mathrm{E}\left[M(\boldsymbol{\Theta})^2\right] \approx M(\hat{\boldsymbol{\theta}})^2 \\ &+\sum_{i=1}^{m}\left(M'_i(\hat{\boldsymbol{\theta}})^2 + M(\hat{\boldsymbol{\theta}})M''_{i,i}(\hat{\boldsymbol{\theta}})\right)\mathrm{Var}[\Theta_i] \\ &+2\sum_{i=1}^{m}\sum_{j=1}^{i-1}\left(M'_i(\hat{\boldsymbol{\theta}})M'_j(\hat{\boldsymbol{\theta}}) + M(\hat{\boldsymbol{\theta}})M''_{i,j}(\hat{\boldsymbol{\theta}})\right)\mathrm{Cov}[\Theta_i,\Theta_j]. \end{aligned} \tag{2.22}$$

Then the variance of output measure with respect to errors of estimates of model parameters can be simplified as follows.

$$\begin{aligned}&\mathrm{Var}[M(\boldsymbol{\Theta})] \approx \\ &\sum_{i=1}^{m} M'_i(\hat{\boldsymbol{\theta}})^2\mathrm{Var}[\Theta_i] - \frac{1}{4}\left(\sum_{i=1}^{m} M''_{i,i}(\hat{\boldsymbol{\theta}})\mathrm{Var}[\Theta_i]\right)^2. \end{aligned} \tag{2.23}$$

In the above, we need not the joint density of estimates but only the variance and covariances of estimates. Therefore it can be applied to the case where it is difficult to obtain the joint density such that we know point estimates and their confidence intervals. On the other hand, we need the information on the first two derivatives of the output measure with respect to the input parameters and efficient computation algorithms for these derivatives, i.e., the local sensitivity algorithms are required. In [10], the computation for the local sensitivity of the CTMC model has been discussed.

2.5 Local Sensitivity Algorithms

2.5.1 Automatic Differentiation for Hierarchical Model

From the analysis in Sec. 2.4.2, we know the first two derivatives of output measure with respect to model parameters are required for achieving the moment-based approximation. This section considers the automatic differentiation for the computation of the hierarchical model. Automatic differentiation is a computation of derivatives of a function based on the chain rule of differentiation. In general, there are two types of automatic differentiation; bottom-up and top-down methods [34]. This chapter focuses on the top-down type of automatic differentiation.

Suppose that the output measure is given by the event probability of the top event of FT in the hierarchical model. Given the structure function $F(\cdot)$ of FT, the output measure is expressed by Eq. (2.11). As aforementioned, $\boldsymbol{\theta}_i$, $i = 1, \ldots, n$, are CTMC parameters representing the dynamic behavior of the phases of i-th basic event, respectively, and $\boldsymbol{\theta}_s = \cup_{i=1}^{n} \boldsymbol{\theta}_i$. The main objectives are to compute the first and second derivatives of $p_s(\boldsymbol{\theta}_s)$; that is, $\partial p_s(\boldsymbol{\theta}_s)/\partial \boldsymbol{\theta}_s$ and $\partial^2 p_s(\boldsymbol{\theta}_s)/\partial \boldsymbol{\theta}_s^2$.

According to the chain rule of differentiation, the first derivative of $p_s(\boldsymbol{\theta}_s)$ with respect to a certain model parameter θ becomes

$$\frac{\partial}{\partial \theta} p_s(\boldsymbol{\theta}_s) = \sum_{i=1}^{n} \frac{\partial F(p_1(\boldsymbol{\theta}_1), p_2(\boldsymbol{\theta}_2), \ldots, p_n(\boldsymbol{\theta}_n))}{\partial p_i(\boldsymbol{\theta}_i)} \frac{\partial p_i(\boldsymbol{\theta}_i)}{\partial \theta}. \tag{2.24}$$

From Eq. (2.6), if $p_i(\boldsymbol{\theta}_i)$ is given by the steady-state probability vector, In the above, the first derivative $\partial p_i(\boldsymbol{\theta}_i)/\partial \theta$ is written in the form:

$$\begin{aligned} &\frac{\partial p_i(\boldsymbol{\theta}_i)}{\partial \theta} = \frac{\partial \boldsymbol{\pi}_i}{\partial \theta} \boldsymbol{r}_i, \\ &\frac{\partial \boldsymbol{\pi}_i}{\partial \theta} \boldsymbol{Q}_i + \boldsymbol{\pi}_i \frac{\partial \boldsymbol{Q}_i}{\partial \theta} = \mathbf{0}, \quad \frac{\partial \boldsymbol{\pi}_i}{\partial \theta} \mathbf{1} = \mathbf{0}, \end{aligned} \tag{2.25}$$

where $\boldsymbol{r}_i$ is a (column) reward vector with the following k-th entry

$$[\boldsymbol{r}_i]_k = \begin{cases} 0 & k \in \mathcal{U}_i, \\ 1 & k \in \mathcal{D}_i. \end{cases} \tag{2.26}$$

Equation (2.25) is essentially the local sensitivity of CTMC. Similarly, in the case of the transient solution of CTMC, we can apply the existing technique to compute the local sensitivity for the transient solution of CTMC [41].

On the other hand, in the same manner to Eq. (2.24), the second derivative of $p_s(\boldsymbol{\theta}_s)$ with respect to θ_x and θ_y is given by

$$\begin{aligned}\frac{\partial^2}{\partial\theta_x\partial\theta_y}p_s(\boldsymbol{\theta}_s) = &\sum_{i=1}^{n}\sum_{j\neq i}^{n}\frac{\partial^2}{\partial p_i(\boldsymbol{\theta}_i)\partial p_j(\boldsymbol{\theta}_j)}F(p_1(\boldsymbol{\theta}_1), p_2(\boldsymbol{\theta}_2),\\ &\ldots, p_n(\boldsymbol{\theta}_n))\frac{\partial p_i(\boldsymbol{\theta}_i)}{\partial\theta_x}\frac{\partial p_j(\boldsymbol{\theta}_j)}{\partial\theta_y}\\ &+\sum_{i=1}^{n}\frac{\partial}{\partial p_i(\boldsymbol{\theta}_i)}F(p_1(\boldsymbol{\theta}_1), p_2(\boldsymbol{\theta}_2),\\ &\ldots, p_n(\boldsymbol{\theta}_n))\frac{\partial^2 p_i(\boldsymbol{\theta}_i)}{\partial\theta_x\partial\theta_y},\end{aligned} \tag{2.27}$$

where the second derivative of $p_i(\boldsymbol{\theta}_i)$ is also obtained by solving the following linear equations when $\boldsymbol{\pi}_i$, $\partial\boldsymbol{\pi}_i/\partial\theta_x$ and $\partial\boldsymbol{\pi}_i/\partial\theta_y$ are given:

$$\begin{aligned}&\frac{\partial^2 p_i(\boldsymbol{\theta}_i)}{\partial\theta_x\partial\theta_y} = \frac{\partial^2\boldsymbol{\pi}_i}{\partial\theta_x\partial\theta_y}\boldsymbol{r}_i,\\ &\frac{\partial^2\boldsymbol{\pi}_i}{\partial\theta_x\partial\theta_y}\boldsymbol{Q}_i + \frac{\partial\boldsymbol{\pi}_i}{\partial\theta_x}\frac{\partial\boldsymbol{Q}_i}{\partial\theta_y} + \frac{\partial\boldsymbol{\pi}_i}{\partial\theta_y}\frac{\partial\boldsymbol{Q}_i}{\partial\theta_x} + \boldsymbol{\pi}_i\frac{\partial^2\boldsymbol{Q}_i}{\partial\theta_x\partial\theta_y} = \mathbf{0},\\ &\frac{\partial^2\boldsymbol{\pi}_i}{\partial\theta_x\partial\theta_y}\mathbf{1} = \mathbf{0}.\end{aligned} \tag{2.28}$$

The remaining problem is how to efficiently compute the local sensitivity for the part of FT because the computation cost to obtain the structure function of FT explicitly is not low. One main contribution of this chapter is to present the local sensitivity of FT with BDD representation.

2.5.2 BDD Representation of FT

A BDD is a compact representation of a Boolean function by a directed acyclic graph based on the Shannon decomposition [35]. Generally, we consider a Boolean function $\phi(x_1, \ldots, x_n)$. Using the Shannon decomposition, the Boolean function can be rewritten as

$$\begin{aligned}&\phi(x_1, \ldots, x_n) =\\ &\quad x_1\phi(1, x_2, \ldots, x_n) + (1 - x_1)\phi(0, x_2, \ldots, x_n).\end{aligned} \tag{2.29}$$

In the above, the Boolean functions $\phi(1, x_2, \ldots, x_n)$ and $\phi(0, x_2, \ldots, x_n)$ can also be decomposed by applying the Shannon decomposition to one of the remaining $n-1$ variables. Eventually, we make a binary tree in which the left and right edges from the i-th level (depth) node correspond to x_i and $1-x_i$, respectively,

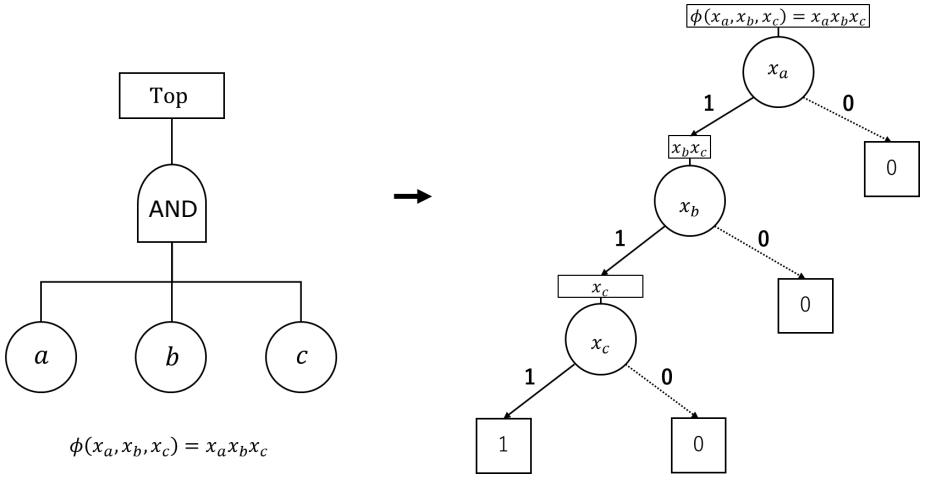

Figure 2.4: An example of a BDD for an AND gate.

and each node represents the decomposed Boolean function like $\phi(1, x_2, \ldots, x_n)$ and $\phi(0, x_2, \ldots, x_n)$. The fundamental idea of BDD is to merge any isomorphic subgraphs of such a binary tree and to eliminate any node whose two children are isomorphic so that it can provide a compact representation for a Boolean function. More details on the BDD can be found in [1]. Figure 2.4 illustrates an example of the BDD representation for a given FT composed by an AND gate.

Given a BDD representation of FT, we can compute the probability of top-event occurrence with an algorithm, which is to traverse all the nodes of BDD according to the following recursive formula:

$$\begin{aligned} p_s(p_1, \ldots, p_n) &= F(p_1, \ldots, p_n) \\ &= p_1 F_{\{p_1=1\}}(p_2, \ldots, p_n) \\ &\quad + (1 - p_1) F_{\{p_1=0\}}(p_2, \ldots, p_n), \end{aligned} \tag{2.30}$$

where $F_{\{A\}}(\cdot)$ is the structure function provided that the variables are assigned with A. The algorithm drastically reduces computation costs using a cache for the intermediate computation.

2.5.3 Automatic Differentiation of FT with BDD

Consider the automatic differentiation of BDD representation. The idea behind our algorithm is the differentiation of the recursive formula $p_s(\boldsymbol{\theta}_s)$. Using

Eq. (2.30), the first derivative of the recursive formula becomes

$$\begin{aligned}\frac{\partial}{\partial\theta}p_s(\boldsymbol{\theta}_s) &= \frac{\partial F(p_1(\boldsymbol{\theta}_1), p_2(\boldsymbol{\theta}_2), \ldots, p_n(\boldsymbol{\theta}_n))}{\partial\theta} \\ &= \frac{\partial p_1(\boldsymbol{\theta}_1)}{\partial\theta} F_{\{p_1=1\}}(p_2(\boldsymbol{\theta}_2), \ldots, p_n(\boldsymbol{\theta}_n)) \\ &\quad + p_1(\boldsymbol{\theta}_1)\frac{\partial}{\partial\theta} F_{\{p_1=1\}}(p_2(\boldsymbol{\theta}_2), \ldots, p_n(\boldsymbol{\theta}_n)) \\ &\quad - \frac{\partial p_1(\boldsymbol{\theta}_1)}{\partial\theta} F_{\{p_1=0\}}(p_2(\boldsymbol{\theta}_2), \ldots, p_n(\boldsymbol{\theta}_n)) \\ &\quad + (1 - p_1(\boldsymbol{\theta}_1))\frac{\partial}{\partial\theta} F_{\{p_1=0\}}(p_2(\boldsymbol{\theta}_2), \ldots, p_n(\boldsymbol{\theta}_n)),\end{aligned} \tag{2.31}$$

where $\partial p_i(\boldsymbol{\theta}_i)/\partial\theta$ is given by Eq. (2.25). The linear equation can be solved when the steady-state probability of CTMC $\boldsymbol{\pi}_i$ is given.

The second derivative of structure function in Eq. (2.27) is also computed based on the formula which is obtained by taking the differentiation of Eq. (2.31) under the BDD representation, that is

$$\begin{aligned}\frac{\partial^2}{\partial\theta_x\partial\theta_y}p_s(\boldsymbol{\theta}_s) &= \frac{\partial^2 p_1(\boldsymbol{\theta}_1)}{\partial\theta_x\partial\theta_y} F_{\{p_1=1\}}(p_2(\boldsymbol{\theta}_2), \ldots, p_n(\boldsymbol{\theta}_n)) \\ &+ \frac{\partial p_1(\boldsymbol{\theta}_1)}{\partial\theta_x}\frac{\partial}{\theta_y} F_{\{p_1=1\}}(p_2(\boldsymbol{\theta}_2), \ldots, p_n(\boldsymbol{\theta}_n)) \\ &+ \frac{\partial p_1(\boldsymbol{\theta}_1)}{\partial\theta_y}\frac{\partial}{\partial\theta_x} F_{\{p_1=1\}}(p_2(\boldsymbol{\theta}_2), \ldots, p_n(\boldsymbol{\theta}_n)) \\ &+ p_1(\boldsymbol{\theta}_1)\frac{\partial^2}{\partial\theta_x\partial\theta_y} F_{\{p_1=1\}}(p_2(\boldsymbol{\theta}_2), \ldots, p_n(\boldsymbol{\theta}_n)) \\ &- \frac{\partial^2 p_1(\boldsymbol{\theta}_1)}{\partial\theta_x\partial\theta_y} F_{\{p_1=0\}}(p_2(\boldsymbol{\theta}_2), \ldots, p_n(\boldsymbol{\theta}_n)) \\ &- \frac{\partial p_1(\boldsymbol{\theta}_1)}{\partial\theta_x}\frac{\partial}{\partial\theta_y} F_{\{p_1=0\}}(p_2(\boldsymbol{\theta}_2), \ldots, p_n(\boldsymbol{\theta}_n)) \\ &- \frac{\partial p_1(\boldsymbol{\theta}_1)}{\partial\theta_y}\frac{\partial}{\partial\theta_x} F_{\{p_1=0\}}(p_2(\boldsymbol{\theta}_2), \ldots, p_n(\boldsymbol{\theta}_n)) \\ &+ (1 - p_1(\boldsymbol{\theta}_1))\frac{\partial^2}{\partial\theta_x\partial\theta_y} F_{\{p_1=0\}}(p_2(\boldsymbol{\theta}_2), \ldots, p_n(\boldsymbol{\theta}_n)).\end{aligned} \tag{2.32}$$

Although the number of terms is large in the case of the second derivative formula, the computation cost can be drastically reduced by storing the computation results $\partial F_{\{A\}}(\cdot)/\partial\theta_x$ and $\partial F_{\{A\}}(\cdot)/\partial\theta_y$ as cache.

2.6 Numerical Experiments

This section illustrates two numerical experiments to solve uncertainty propagation in the hierarchical reliability models. The first experiment considers

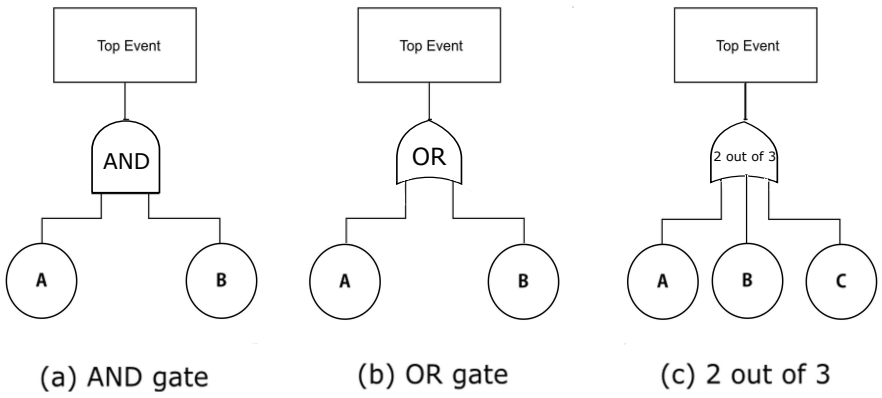

Figure 2.5: Three hierarchical reliability models.

uncertainty propagation in three simple hierarchical models and validates the accuracy of the proposed moment-based approximation method using Bayes estimation. In the second experiment, we apply the proposed approach to a real case study with the IBM SIP application server cluster [11]. These experiments are conducted on macOS 12.0 with the chip Apple M1 for computation.

2.6.1 Experiment I

Figure 2.5 depicts three simple hierarchical reliability models, whose upper levels are represented by FTs, and the lower levels are described by the CTMCs. The gates of the three models are AND gate, OR gate, and 2-out-of-3, respectively. In the lower level, each component is supposed to have two states; UP and DOWN. Note that the component will be replaced if failed in each hierarchical model.

In the experiment, the moment-based approximation is used to evaluate the uncertainty propagation in the hierarchical models. Also, to validate the accuracy of the proposed moment-based approach, the well-known uncertainty analysis method, Bayes estimation, is considered for comparison.

Assume that the time to failure of a single component follows the exponential distribution with parameter λ, thus the system reliability of the component at time t can be expressed by $R(t;\lambda) = e^{-\lambda t}$. Suppose that $\mathcal{D} = (t_1, \cdots, t_n)$ are

independent and identically distributed (i.i.d.) samples as observed failure times for the component and the prior density of the failure rate Λ follows a gamma distribution with shape parameter α and rate β as follows.

$$f_\Lambda(\lambda) = \frac{\beta^\alpha \lambda^{\alpha-1} e^{-\beta\lambda}}{\Gamma(\alpha)}. \tag{2.33}$$

By using Bayes theory, the posterior density of the failure rate $f_\Lambda(\lambda|\mathcal{D})$ is also a gamma distribution with shape parameters $n+\alpha$ and rate parameters $s+\beta$, where $s = \sum_{i=1}^{n} t_i$. Then the mean of $R(t;\lambda)$ becomes

$$\begin{aligned} \mathrm{E}[R(t;\lambda)] &= \int_0^\infty e^{-\lambda t} f_\Lambda(\lambda|\mathcal{D}) d\lambda \\ &= (1 + \frac{\hat{\lambda} t}{n+\alpha})^{-(n+\alpha)}, \end{aligned} \tag{2.34}$$

where $\hat{\lambda} = \mathrm{E}[\Lambda] = \dfrac{n+\alpha}{s+\beta}$. At the same time, the variance of $R(t;\lambda)$ is given by

$$\begin{aligned} \mathrm{Var}[R(t;\lambda)] = &(1 + \frac{2\hat{\lambda} t}{n+\alpha})^{-(n+\alpha)} \\ &- (1 + \frac{\hat{\lambda} t}{n+\alpha})^{-2(n+\alpha)}. \end{aligned} \tag{2.35}$$

On the other hand, using the moment-based method, we have the following approximation formula:

$$\begin{aligned} \mathrm{E}[R(t;\lambda)] &\approx e^{-\hat{\lambda} t} + \frac{1}{2} t^2 e^{-\hat{\lambda} t} \mathrm{Var}[\Lambda] \\ &= (1 + \frac{1}{2} t^2 \mathrm{Var}[\Lambda]) e^{-\hat{\lambda} t}, \end{aligned} \tag{2.36}$$

and

$$\begin{aligned} \mathrm{Var}[R(t;\lambda)] &\approx t^2 e^{-2\hat{\lambda} t} \mathrm{Var}[\Lambda] - \frac{1}{4} t^4 e^{-2\hat{\lambda} t} \mathrm{Var}[\Lambda]^2 \\ &= \mathrm{Var}[\Lambda](1 - \frac{1}{4} t^2 \mathrm{Var}[\Lambda]) t^2 e^{-2\hat{\lambda} t}. \end{aligned} \tag{2.37}$$

Besides, we set λ_a = 1.0e-5, λ_b = 2.5e-5, λ_c = 4.0e-5 as the real failure rates of components A, B, and C whose failure times follow exponential distributions. Commonly, uncertainty occurs when the sample size is limited, which can be reflected in variance. To control the variance of failure rates, five cases are considered for comparison; that is, sample size n = 2, 5, 10, 20, and 30. Thus, the mean value of outputs can be obtained through posterior density and the moment-based approximation method. In particular, prior hyperparameters are

Table 2.3: System reliabilities under different structures ($t = 10$)

n	AND			
	Exact	Plug-in	Moment	Bayes
2	9.98252e-1	9.96316e-1	9.96317e-1	9.96317e-1
5	9.98252e-1	9.97544e-1	9.97545e-1	9.97545e-1
10	9.98252e-1	9.97242e-1	9.97242e-1	9.97242e-1
20	9.98252e-1	9.98272e-1	9.98272e-1	9.98272e-1
30	9.98252e-1	9.98161e-1	9.98161e-1	9.98161e-1
n	OR			
	Exact	Plug-in	Moment	Bayes
2	9.99999e-1	9.99998e-1	9.99998e-1	9.99998e-1
5	9.99999e-1	9.99999e-1	9.99999e-1	9.99999e-1
10	9.99999e-1	9.99998e-1	9.99998e-1	9.99998e-1
20	9.99999e-1	9.99999e-1	9.99999e-1	9.99999e-1
30	9.99999e-1	9.99999e-1	9.99999e-1	9.99999e-1
n	2-out-of-3			
	Exact	Plug-in	Moment	Bayes
2	9.99996e-1	9.99986e-1	9.99987e-1	9.99985e-1
5	9.99996e-1	9.99993e-1	9.99996e-1	9.99993e-1
10	9.99996e-1	9.99995e-1	9.99995e-1	9.99995e-1
20	9.99996e-1	9.99996e-1	9.99996e-1	9.99996e-1
30	9.99996e-1	9.99996e-1	9.99996e-1	9.99996e-1

set as $\alpha = 0$ and $\beta = 0$ to make this similar to Jeffery's prior. Meanwhile, the time t is divided into 10 and 1,000 as system conditions under two different scenarios.

The results are shown in Table 2.3 and Table 2.4, where *Exact* means the system reliability computed by using the preset failure rates directly, *Plug-in* indicates the system reliability computed through point estimation, whereas *Moment* and *Bayes* correspond the system reliabilities obtained by moment-based approximation and Bayes estimation, respectively. In the case of $t = 10$, it seems that the results of moment-based approximation are very close to those

Table 2.4: System reliabilities under different structures ($t = 1000$)

n	AND			
	Exact	Plug-in	Moment	Bayes
2	9.65605e-1	9.52012e-1	9.52407e-1	9.52402e-1
5	9.65605e-1	9.60454e-1	9.60540e-1	9.60539e-1
10	9.65605e-1	9.46257e-1	9.46342e-1	9.46341e-1
20	9.65605e-1	9.66009e-1	9.66024e-1	9.66024e-1
30	9.65605e-1	9.63864e-1	9.63877e-1	9.63877e-1
n	OR			
	Exact	Plug-in	Moment	Bayes
2	9.99754e-1	9.99629e-1	9.99634e-1	9.99634e-1
5	9.99754e-1	9.99639e-1	9.99641e-1	9.99641e-1
10	9.99754e-1	9.99382e-1	9.99384e-1	9.99384e-1
20	9.99754e-1	9.99708e-1	9.99708e-1	9.99708e-1
30	9.99754e-1	9.99744e-1	9.99744e-1	9.99744e-1
n	2-out-of-3			
	Exact	Plug-in	Moment	Bayes
2	9.98415e-1	9.97409e-1	9.98351e-1	9.97456e-1
5	9.98415e-1	9.99082e-1	9.99190e-1	9.99086e-1
10	9.98415e-1	9.98258e-1	9.98366e-1	9.98263e-1
20	9.98415e-1	9.98350e-1	9.98406e-1	9.98353e-1
30	9.98415e-1	9.98774e-1	9.98780e-1	9.98775e-1

of Bayes estimation and has a slight difference from the exact results in the AND gate structure. With the increase of data samples, the results of *Plug-in*, *Moment*, and *Bayes* approximate the exact results. This indicates that the moment-based approximation method is highly accurate even when the sample size is small. Apart from AND gate structure, the uncertainty has minimal effects when the model is composed of the OR gate. In the OR gate structure, the results of each method in different sample sizes are very close to the exact results. While in the 2-out-of-3 structure, only when the sample size is 2 or 5, the differences in the results among all methods become small. However, when

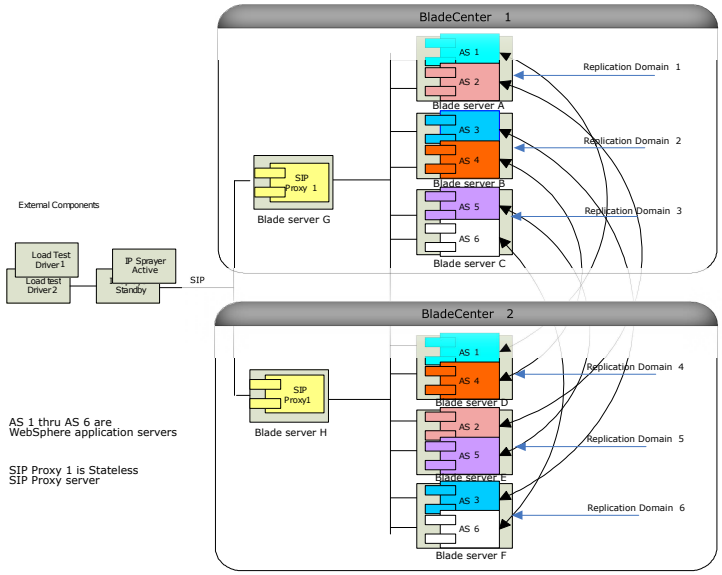

Figure 2.6: IBM SIP application server cluster [11].

the sample size increases, the results of these methods are close to the exact results gradually.

In the case of $t = 1000$, the effects of uncertainty become apparent. Similar to the case where $t = 10$, the computation results of the moment-based approximation are different from Bayes estimation when the sample size is small but close to each other with the sample size increase. In Table 2.3, Bayes estimation is closer to the exact result than the moment-based approximation in most situations. However, from Table 2.4, *Moment* and *Bayes* have a significant difference regardless of the sample size.

As a result, the moment-based approximation can evaluate the uncertainty in model parameters with high accuracy. Also, compared with the Bayes estimation, the moment-based approximation has another advantage; Note that in Bayes estimation, the closed-form of the system reliability is necessary, but in the moment-based approximation, the system reliability can be computed if the system structure is known, which will be discussed in the following experiment.

2.6.2 Experiment II

In this subsection, we deal with the hierarchical availability model for the IBM SIP application server cluster introduced by Trivedi *et al.* [11], and evaluate its

Table 2.5: Parameters for node operating system (OS) (4)

Description	Value
Mean time for OS failure	4,000 hr
Mean time to detect the OS failure	1 hr
Mean time for node reboot	10 min
Mean time to repair OS	1 hr

epistemic uncertainty propagation as a real case study. Since the accuracy of the moment-based approximation method has been validated through Experiment I, here we will discuss the application and performance of the proposed approach to a real system compared with the simulation method. The system configuration is shown in Fig. 2.6. The hierarchical availability model for the cluster consists of FTs and CTMCs, and can be divided into three levels; the top-level FT for the whole system failure, the middle-level FTs for chassis faults and node hardware failures, and the bottom level are CTMCs describing the dynamic behaviors of individual subsystems, such as the midplane, blade CPU, power domain, cooling subsystem, application server, proxy, etc. In this experiment, the minimum number of failed application servers for system unavailability is given by 6, and the model contains one 6-out-of-12 gate, 15 AND gates, and 71 OR gates representing the failure relationship between 182 leaf nodes, each node is represented by CTMCs with 3 to 10 states. The entire state of the whole system is 3.0e+56. More details on the hierarchical availability model can be referenced in [11]. Tables 2.5 through 2.8 summarize the model parameters, which are purely hypothetical (but reasonable) in nature [11]. The model parameters are from the application/proxy server, node OS, and hardware models. For example, 18 mean time-related parameters exist in the application and proxy servers and 23 in the hardware models. Particularly, the corresponding coverage factors are given in Table 2.8. The total number of model parameters is 58.

In the experiment, aiming to evaluate the performance of the proposed approach on the large-size and complicated hierarchical model, we compare the results of the moment-based approach with the commonly-used method, the

Table 2.6: Parameters for application (app.) and proxy servers (18)

Description	Value
Mean time to hangup (app.)	2,920 hr
Mean time to server failure (app.)	2,920 hr
Mean time for workload manager failure detection (app.)	2 s
Mean time for node agent failure detection (app.)	2 s
Mean time for manual failure detection (app.)	10 min
Mean time for automatic process restart (app.)	10 s
Mean time for manual process restart (app.)	60 s
Mean time for manual node reboot (app.)	10 min
Mean time for manual repair (app.)	8 hr
Mean time to hangup (proxy)	2,920 hr
Mean time to server failure (proxy)	2,920 hr
Mean time for workload manager failure detection (proxy)	2 s
Mean time for node agent failure detection (proxy)	2 s
Mean time for manual failure detection (proxy)	10 min
Mean time for automatic process restart (proxy)	10 s
Mean time for manual process restart (proxy)	60 s
Mean time for manual node reboot (proxy)	10 min
Mean time for manual repair (proxy)	8 hr

Monte Carlo (MC) simulation. Although MC simulation is a practical and flexible method in modeling and analyzing real-world systems and situations, the accuracy of this method strongly depends on the simulation times, which means achieving a highly accurate result usually incurs a high computational cost.

Firstly, like [11], we consider the case without considering the parameter uncertainty; that is, the values of model parameters in Tables 2.5 through 2.8 are directly applied for computing the system unavailability. The obtained system unavailability using the moment-based approach is 2.015e-6 with the computational time is 0.0697 seconds, which is close to the result evaluated in [11] (i.e., 2.2e-6).

Then, we discuss the cases where uncertainty propagation is considered. The

Table 2.7: Parameters for hardware models (23)

Description	Value
Mean time for mid-plane failure	10^6 hr
Mean time for blower failure	10^6 hr
Mean time for power module failure	10^6 hr
Mean time for processor failure	10^6 hr
Mean time for Base failure	10^6 hr
Mean time for ethernet switch failure	10^6 hr
Mean time for network interface card (NIC) failure	10^6 hr
Mean time for memory DIMM failure	10^6 hr
Mean time for hard disk failure	10^6 hr
Mean time for failure detection plus repair person arrival	2.5 hr
Mean time to repair mid-plane	1 hr
Mean time to repair blower	1 hr
Mean time to repair two blowers	1.5 hr
Mean time to repair power module	1 hr
Mean time to repair two power modules	1.5 hr
Mean time to repair processor	1 hr
Mean time to repair Base	1 hr
Mean time to repair the ethernet switch	1 hr
Mean time to repair NIC	1 hr
Mean time to repair memory bank	1 hr
Mean time to repair hard disk	1 hr
Mean time to repair two hard disks	1.5 hr
Mean time to copy disk data	10 min

simulation times are set at 10,000 to ensure the accuracy of the MC simulation method. In particular, the confidence interval at 95% is considered for MC simulation. Besides, the data sample size is set as $n = 20$. In the moment-based approximation, the computational time mainly depends on the computation of the second derivative of the output measure given in Eq. (2.19); in other words, the number of model parameters significantly affects the computational time.

Table 2.8: Coverage factors for application server, proxy server, node OS, and hardware models (13)

Description	Value
Coverage factor for WLM detection (app.)	0.95
Coverage factor for node agent detection (app.)	0.95
Coverage factor for auto process restart (app.)	0.95
Coverage factor for manual process restart (app.)	0.95
Coverage factor for manual node restart (app.)	0.95
Coverage factor for WLM detection (proxy)	0.95
Coverage factor for node agent detection (proxy)	0.95
Coverage factor for auto process restart (proxy)	0.95
Coverage factor for manual process restart (proxy)	0.95
Coverage factor for manual node restart (proxy)	0.95
Coverage factor for node reboot to recover OS	0.95
Coverage factor for power module failure	0.99
Probability of mid-plane common mode failure	0.001

Thus to investigate such an effect regarding parameter number, we consider the following four cases:

- Case I: Consider the uncertainty propagation in all model parameters except the coverage factors.

- Case II: Consider the uncertainty propagation in all model parameters except the coverage factors and the parameters related to the application server.

- Case III: Only consider the uncertainty propagation in both node OS and hardware models.

- Case IV: Only consider the uncertainty propagation in the hardware models.

Table 2.9: System unavailability and corresponding computational time

Cases	Number of uncertain parameters	System unavailability		Computational time	
		Moment	MC	Moment	MC
Case I	45	2.34421e-6	2.34421e-6 $\pm$ 0.01245e-6	25.27	35.07
Case II	36	2.34421e-6	2.35467e-6 $\pm$ 0.01222e-6	18.46	35.21
Case III	27	2.34421e-6	2.33183e-6 $\pm$ 0.01231e-6	12.51	35.15
Case IV	23	2.35269e-6	2.35922e-6 $\pm$ 0.01198e-6	8.85	35.12

The computational results are demonstrated in Table 2.9. Obviously, in all cases, the obtained system unavailabilities using the moment-based approximation method lay in the confidence interval range estimated from the MC simulation. That means the moment-based approximation method is well performed despite the decrease in uncertain parameters. Furthermore, as the number of uncertain model parameters decreases, the computational speed of the moment-based approach improves a lot; for example, in the case of 23 uncertain parameters, the computational time is 8.85 seconds, which is only one-third of the value for 46 parameters. However, the change in the parameter number brings almost no impact on the computational speed of the MC simulation method, and its computational time is around 35 seconds. In summary, the moment-based approach has a good performance with high accuracy, close to the estimation result of MC simulation, but significantly reduces the computational cost when the number of uncertain parameters is small compared with the MC simulation. For example, under Case IV, the computational time of the MC simulation is four times longer than the moment-based method.

Chapter 3

Variance-Based Sensitivity for Epistemic Uncertainty

3.1 Introduction

Variance-based global sensitivity analysis is often used to rank the importance of input factors, based on their contribution to the variance of the output measure of interest [13]. The variance-based sensitivity analysis relies on the computation of conditional variances, is sampling-based, and therefore usually applies simulation methods such as Monte Carlo simulation.

In this chapter, we present an analytic approach to compute the variance-based sensitivity based on moment approximation, this topic is also discussed in [39]. More specifically, we formulate the output measure of CTMC and investigate the relationship between input parameters and output measure through variance-based sensitivity analysis. In numerical experiments, the effects of model parameters in both parallel and series system configurations are evaluated to validate the proposed approach. In particular, the discussion of the approach on complex system is also taken into account.

3.2 Markov Reward Model (MRM)

Consider a CTMC $\{X(t); t \geq 0\}$ with discrete state space $\mathcal{S} = \{1, 2, \cdots, n\}$. The transient state probability (row) vector is defined as $\pi(t) = [P(X(t) = i)]_{i=1,\cdots,n}$. Given the initial state probability vector $\boldsymbol{\pi}(0)$, the state probability vector $\boldsymbol{\pi}(t)$ at any time t can be obtained by solving the following differential

equation:

$$\frac{d}{dt}\boldsymbol{\pi}(t) = \boldsymbol{\pi}(t)Q, \tag{3.1}$$

where $\boldsymbol{Q}$ is called the infinitesimal generator matrix and is a square matrix whose (i, j)-element represents the transition rate from state i to state j. By defining the reward vector $\boldsymbol{r}$ as below:

$$[\boldsymbol{r}]_i = \begin{cases} 1 & \text{System is up} \\ 0 & \text{System is failed} \end{cases} \tag{3.2}$$

in which $[\cdot]_i$ is the i-th element of $\boldsymbol{r}$. Then, one significant dependability measure of interest, that is the system reliability, can be obtained by

$$R(t) = \boldsymbol{\pi}(t)\boldsymbol{r}. \tag{3.3}$$

On the other hand, when time $t \to \infty$ the stationary state probability vector $\boldsymbol{\pi}_{ss} = \lim_{t\to\infty}\boldsymbol{\pi}(t)$ is obtained by solving the following linear equations:

$$\boldsymbol{\pi}_{ss}\boldsymbol{Q} = \mathbf{0}, \quad \boldsymbol{\pi}_{ss}\mathbf{1} = 1, \tag{3.4}$$

where $\mathbf{0}$ and $\mathbf{1}$ are row and column vectors whose entries are 0 and 1, respectively. Given the reward vector $\boldsymbol{r}$, the steady-state system availability A_{ss} is

$$A_{ss} = \boldsymbol{\pi}_{ss}\boldsymbol{r}. \tag{3.5}$$

3.3 Variance-based Sensitivity Analysis

Variance-based sensitivity analysis also called Sobol method is a popular measure for sensitivity analysis. The variance of output is decomposed into the conditional variance provided a set of input variables, which is similar to the analysis of variance [40]. Let $y = f(\boldsymbol{x})$ be a system where $\boldsymbol{x} = (x_1, \cdots, x_d)$ is a vector of input variables and y is an output measure. When the input variables are given by a random vector $X = (X_1, \cdots, X_d)$, the output measure also follows a random variable Y, i.e, $Y = f(X)$. For the sake of simplicity, suppose that X is uniformly distributed in a unit hypercube, i.e., X_i follows a uniform distribution. Now we assume that Y is decomposed as follows.

$$\begin{aligned} Y = g_0 &+ \sum_{i=1}^{d} g_i(X_i) + \sum_{i<j}^{d} g_{i,j}(X_i, X_j) \\ &+ \sum_{i<j<k}^{d} g_{i,j,k}(X_i, X_j, X_k) + \cdots + g_{1,2,\cdots,d}(X_1, \cdots, X_d), \end{aligned} \tag{3.6}$$

where

$$\int_0^1 g_i(X_i)dX_i = 0, \tag{3.7}$$

$$\int_0^1 g_{i,j}(X_i, X_j)dX_i dX_j = 0, \tag{3.8}$$

$$\vdots$$

By taking expectation, we have

$$g_0 = \mathrm{E}[Y], \tag{3.9}$$

$$g_i(X_i) = \mathrm{E}[Y|X_i] - g_0, \tag{3.10}$$

$$g_{i,j}(X_i, X_j) = \mathrm{E}[Y|X_i, X_j] - g_0 - g_i(X_i) - g_j(X_j), \tag{3.11}$$

$$\vdots$$

Based on these above equations, the variance of output Y can also be decomposed as follows.

$$\mathrm{Var}[Y] = \sum_{i=1}^{d} V_i + \sum_{i<j}^{d} V_{i,j} + \sum_{i<j<k}^{d} V_{i,j,k} + \cdots + V_{1,2,\cdots,d}, \tag{3.12}$$

where

$$V_i = \mathrm{E}[\mathrm{Var}[Y|X_i]], \tag{3.13}$$

$$V_{i,j} = \mathrm{E}[\mathrm{Var}[Y|X_i, X_j]] - V_i - V_j, \tag{3.14}$$

$$\vdots$$

One important index of the sensitivity analysis is the main effect or the first-order sensitivity defined below:

$$S_i = \frac{V_i}{\mathrm{Var}[Y]}, \tag{3.15}$$

which indicates the contribution to the output variance of the main effect of X_i. Also, in general, for random variables X and Y, it holds

$$\mathrm{Var}[Y] = \mathrm{E}[\mathrm{Var}[Y|X]] + \mathrm{Var}[\mathrm{E}[Y|X]]. \tag{3.16}$$

Then the main effect of X_i is given by

$$S_i = \frac{V_i}{\mathrm{Var}[Y]} = \frac{\mathrm{Var}[Y] - \mathrm{Var}[\mathrm{E}[Y|X_i]]}{\mathrm{Var}[Y]}. \tag{3.17}$$

The value of the main effect determines the importance of the input parameters. The larger the main effect is, the more important the parameter is. As an index to measure parameters, main effect also implies the contribution of parameters to the system dependability or performance. It is based on the conditional variance of other parameters, so when considering the higher order interaction main effects $S_{i,j}, S_{i,j,k}$ and so on, the following relationship is obtained,

$$\sum_{i=1}^{d} S_i + \sum_{i<j}^{d} S_{i,j} + \cdots + S_{1,2,\cdots,d} = 1. \tag{3.18}$$

In general, when input variables are independent with each other, the relationship between main effects can be simplified through $\sum_{i=1}^{d} S_i = 1$. Otherwise, the higher order interaction main effects should be taken into account. From Eq. (3.17), it is clear that the computation of $\mathrm{Var}[Y]$ and $\mathrm{E}[Y|X]$ are required for evaluating the main effect of X. As aforementioned in the introduction, the normal simulation is hard to fit a general formula. To address the above issue, this chapter considers the moment-based approximation method in the next section.

3.4 Moment-based Approximation Method

Let $M(\boldsymbol{\theta})$ be an output measure of an aleatory probability model, e.g., the dependability measures such as system reliability, where $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_l)$ is a column vector of input parameters needed to compute the output measure. Since the input parameters are estimated from data samples, there would be statistical errors so-called uncertainty exists in estimation. The traditional way is applying the Bayes theorem, which regards the input parameters as the random variables. Hereby, we suppose that these model parameters are randomly distributed, i.e., the parameter vector is defined as a vector of random variables $\boldsymbol{\Theta} = (\Theta_1, \ldots, \Theta_l)$. When the prior density functions of input parameters are given, the joint epistemic density which is also the posterior density $f_{\boldsymbol{\Theta}}(\boldsymbol{\theta})$ of parameters can be obtained with samples $\mathcal{D}$. Then the conditional expected value of $M(\boldsymbol{\Theta})$ can be obtained with

$$\mathrm{E}[M(\boldsymbol{\Theta})] = \int M(\boldsymbol{\theta}) f_{\boldsymbol{\Theta}}(\boldsymbol{\theta}|\mathcal{D}) d\boldsymbol{\theta}, \tag{3.19}$$

The Moment-based approximation who has the merit of simplifying the integral, by taking a Taylor series expansion of the expected value of the output measure, we have the following equation [10]:

$$\begin{aligned}\mathrm{E}[M(\boldsymbol{\Theta})] = M(\hat{\boldsymbol{\theta}}) + \mathrm{E}[M'(\hat{\boldsymbol{\theta}})^T(\boldsymbol{\theta}-\hat{\boldsymbol{\theta}})] \\ + \frac{1}{2}\mathrm{E}[(\boldsymbol{\theta}-\hat{\boldsymbol{\theta}})^T M''(\hat{\boldsymbol{\theta}})(\boldsymbol{\theta}-\hat{\boldsymbol{\theta}})] + \cdots,\end{aligned} \tag{3.20}$$

where $\hat{\boldsymbol{\theta}}$ is the point estimate value. Since $\hat{\boldsymbol{\theta}} = \mathrm{E}[\boldsymbol{\Theta}]$, the second term of Taylor series expansion becomes 0. We have the following approximation

$$\begin{aligned}\mathrm{E}[M(\boldsymbol{\Theta})] &\approx M(\hat{\boldsymbol{\theta}}) + \frac{1}{2}\mathrm{E}[(\boldsymbol{\theta}-\hat{\boldsymbol{\theta}})^T M''(\hat{\boldsymbol{\theta}})(\boldsymbol{\theta}-\hat{\boldsymbol{\theta}})] \\ &= M(\hat{\boldsymbol{\theta}}) + \frac{1}{2}\sum_{i=1}^{l} M''_{i,i}(\hat{\boldsymbol{\theta}})\mathrm{Var}[\Theta_i] \\ &\quad + \sum_{i=1}^{l}\sum_{j=1}^{i-1} M''_{i,j}(\hat{\boldsymbol{\theta}})\mathrm{Cov}[\Theta_i,\Theta_j].\end{aligned} \tag{3.21}$$

In particular, when $(\Theta_i, \cdots, \Theta_l)$ are mutually independent, that is $\mathrm{Cov}[\Theta_i, \Theta_j] = 0$ for $i \neq j$, the above equation becomes

$$\mathrm{E}[M(\boldsymbol{\Theta})] \approx M(\hat{\boldsymbol{\theta}}) + \frac{1}{2}\sum_{i=1}^{l} M''_{i,i}(\hat{\boldsymbol{\theta}})\mathrm{Var}[\Theta_i]. \tag{3.22}$$

Similarly, the Taylor series expansion of the second moment of $M(\boldsymbol{\Theta})$ can also be obtained as

$$\begin{aligned}\mathrm{E}[M(\boldsymbol{\Theta})^2] = M(\hat{\boldsymbol{\theta}})^2 + \mathrm{E}[2M(\hat{\boldsymbol{\theta}})M'(\hat{\boldsymbol{\theta}})^T(\boldsymbol{\theta}-\hat{\boldsymbol{\theta}})] \\ + \mathrm{E}[(\boldsymbol{\theta}-\hat{\boldsymbol{\theta}})^T(M'(\hat{\boldsymbol{\theta}})M'(\hat{\boldsymbol{\theta}})^T \\ + M(\hat{\boldsymbol{\theta}})M''(\hat{\boldsymbol{\theta}}))(\boldsymbol{\theta}-\hat{\boldsymbol{\theta}})] + \cdots,\end{aligned} \tag{3.23}$$

and it approximation is given by

$$\begin{aligned}\mathrm{E}[M(\boldsymbol{\Theta})^2] &\approx M(\hat{\boldsymbol{\theta}})^2 \\ &+ \sum_{i=1}^{l}\left(M'_i(\hat{\boldsymbol{\theta}})^2 + M(\hat{\boldsymbol{\theta}})M''_{i,i}(\hat{\boldsymbol{\theta}})\right)\mathrm{Var}[\Theta_i] \\ &+ 2\sum_{i=1}^{l}\sum_{j=1}^{i-1}\left(M'_i(\hat{\boldsymbol{\theta}})M'_j(\hat{\boldsymbol{\theta}}) + M(\hat{\boldsymbol{\theta}})M''_{i,j}(\hat{\boldsymbol{\theta}})\right)\mathrm{Cov}[\Theta_i,\Theta_j].\end{aligned} \tag{3.24}$$

In the case of mutually independent $(\Theta_i, \cdots, \Theta_l)$, the above equation reduces

$$\begin{aligned}\mathrm{E}[M(\boldsymbol{\Theta})^2] &\approx M(\hat{\boldsymbol{\theta}})^2 \\ &+ \sum_{i=1}^{l}\left(M'_i(\hat{\boldsymbol{\theta}})^2 + M(\hat{\boldsymbol{\theta}})M''_{i,i}(\hat{\boldsymbol{\theta}})\right)\mathrm{Var}[\Theta_i],\end{aligned} \tag{3.25}$$

where

$$M_i'(\hat{\boldsymbol{\theta}}) = \left.\frac{\partial M(\boldsymbol{\theta})}{\partial \theta_i}\right|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}}, \quad M_{i,j}''(\hat{\boldsymbol{\theta}}) = \left.\frac{\partial^2 M(\boldsymbol{\theta})}{\partial \theta_i \partial \theta_j}\right|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}}. \tag{3.26}$$

Moreover, since $\mathrm{Var}[M(\boldsymbol{\Theta})] = \mathrm{E}[M(\boldsymbol{\Theta})^2] - \mathrm{E}[M(\boldsymbol{\Theta})]^2$, we then have

$$\begin{aligned}\mathrm{Var}[M(\boldsymbol{\Theta})] \approx & \sum_{i=1}^{l} M_i'(\hat{\boldsymbol{\theta}})^2 \mathrm{Var}[\Theta_i] \\ & - \frac{1}{4}(\sum_{i=1}^{l} M_{i,i}''(\hat{\boldsymbol{\theta}})\mathrm{Var}[\Theta_i])^2. \end{aligned} \tag{3.27}$$

The above approximations require the first two derivatives of the output measure $M(\boldsymbol{\theta})$ with respect to the input parameters. They can be computed by parametric sensitivity analysis [41, 42]. On the other hand, the necessary information on the joint density are their expectations, variances, and covariances only, which means the actual form of density distribution is not necessary. Meanwhile, the variance-based sensitivity analysis needs previous information such as expectations and variances. Based on moment-based approximation method, the formulation of variance-based sensitivity analysis can be achieved.

3.5 Variance-based Sensitivity Analysis for MRMs

Suppose that $M(\boldsymbol{\theta})$ is the output measure such as the system reliability of an MRM and $\boldsymbol{\Theta} = (\Theta_1, \ldots, \Theta_n)$ is the vector of random variables for parameter vector $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_n)$. The conditional expectation of $M(\boldsymbol{\Theta})$ becomes

$$\begin{aligned}\mathrm{E}[M(\boldsymbol{\Theta})|\theta_l] \approx & M(\hat{\boldsymbol{\theta}}) + M_l'(\hat{\boldsymbol{\theta}})(\theta_l - \hat{\theta}_l) \\ & + \frac{1}{2}\sum_{i=1, i\neq l}^{n} M_{i,i}''(\hat{\boldsymbol{\theta}})\mathrm{Var}(\Theta_i) + \frac{1}{2}M_{l,l}''(\theta_l - \hat{\theta}_l)^2. \end{aligned} \tag{3.28}$$

Also we have

$$\mathrm{E}[M(\boldsymbol{\Theta})|\hat{\theta}_l] \approx M(\hat{\boldsymbol{\theta}}) + \frac{1}{2}\sum_{i=1, i\neq l}^{n} M_{i,i}''(\hat{\boldsymbol{\theta}})\mathrm{Var}(\Theta_i), \tag{3.29}$$

$$\left.\frac{\partial}{\partial \theta_l}\mathrm{E}[M(\boldsymbol{\Theta})|\boldsymbol{\theta}]\right|_{\theta_l=\hat{\theta}_l} \approx M_l'(\hat{\boldsymbol{\theta}}), \tag{3.30}$$

$$\left.\frac{\partial^2}{\partial \theta_l^2}\mathrm{E}[M(\boldsymbol{\Theta})|\boldsymbol{\theta}]\right|_{\theta_l=\hat{\theta}_l} \approx M_{l,l}''(\hat{\boldsymbol{\theta}}). \tag{3.31}$$

The second moment of $\mathrm{E}[M(\boldsymbol{\Theta})|\theta_l]$ is approximated by

$$\begin{aligned}
\mathrm{E}[\mathrm{E}[M(\boldsymbol{\Theta})|\theta_l]^2] \approx \mathrm{E}\Bigg[&\mathrm{E}[M(\boldsymbol{\Theta})|\hat{\theta}_l]^2 \\
&+ 2\mathrm{E}[M(\boldsymbol{\Theta})|\hat{\theta}_l]M_l'(\hat{\boldsymbol{\theta}})(\theta_l - \hat{\theta}_l) \\
&+ \Big(M_l'(\hat{\boldsymbol{\theta}})^2 + \mathrm{E}[M(\boldsymbol{\Theta})|\hat{\theta}_l]M_{l,l}''(\hat{\boldsymbol{\theta}})\Big)(\theta_l - \hat{\theta}_l)^2\Bigg] \\
= \mathrm{E}[M(\boldsymbol{\Theta})|\hat{\theta}_l]^2 &\\
&+ \Big(M_l'(\hat{\boldsymbol{\theta}})^2 + \mathrm{E}[M(\boldsymbol{\Theta})|\hat{\theta}_l]M_{l,l}''(\hat{\boldsymbol{\theta}})\Big)\mathrm{Var}[\Theta_l] \\
= (\mathrm{E}[M(\boldsymbol{\Theta})] - \frac{1}{2}M_{l,l}''(\hat{\boldsymbol{\theta}})\mathrm{Var}[\Theta_l])^2 &\\
&+ \Big(M_l'(\hat{\boldsymbol{\theta}})^2 + \mathrm{E}[M(\boldsymbol{\Theta})|\hat{\theta}_l]M_{l,l}''(\hat{\boldsymbol{\theta}})\Big)\mathrm{Var}[\Theta_l] \\
= \mathrm{E}[M(\boldsymbol{\Theta})]^2 - \mathrm{E}[M(\boldsymbol{\Theta})]&M_{l,l}''(\hat{\boldsymbol{\theta}})\mathrm{Var}[\Theta_l] \\
&+ \frac{1}{4}M_{l,l}''(\hat{\boldsymbol{\theta}})^2\mathrm{Var}[\Theta_l]^2 + \Big(M_l'(\hat{\boldsymbol{\theta}})^2 \\
&+ \mathrm{E}[M(\boldsymbol{\Theta})|\hat{\theta}_l]M_{l,l}''(\hat{\boldsymbol{\theta}})\Big)\mathrm{Var}[\Theta_l]. \qquad (3.32)
\end{aligned}$$

Therefore, the main effect can be obtained by

$$S_i = \frac{V_i}{\mathrm{Var}[M(\boldsymbol{\Theta})]} = \frac{\mathrm{Var}[M(\boldsymbol{\Theta})] - \mathrm{Var}[\mathrm{E}[M(\boldsymbol{\Theta})|\theta_l]]}{\mathrm{Var}[M(\boldsymbol{\Theta})]}. \qquad (3.33)$$

In the above equation $\mathrm{Var}[M(\boldsymbol{\Theta})]$ can be obtained by using Eq. (3.27). The main objective now is to compute $\mathrm{Var}[\mathrm{E}[M(\boldsymbol{\Theta})|\theta_l]]$. According to $\mathrm{Var}[M(\boldsymbol{\Theta})] = \mathrm{E}[M(\boldsymbol{\Theta})^2] - \mathrm{E}[M(\boldsymbol{\Theta})]^2$, we have

$$\begin{aligned}
\mathrm{Var}[\mathrm{E}[M(\boldsymbol{\Theta})|\theta_l]] &= \mathrm{E}[\mathrm{E}[M(\boldsymbol{\Theta})|\theta_l]^2] - \mathrm{E}[\mathrm{E}[M(\boldsymbol{\Theta})|\theta_l]]^2 \\
&= \mathrm{E}[\mathrm{E}[M(\boldsymbol{\Theta})|\boldsymbol{\theta}_l]^2] - \mathrm{E}[M(\boldsymbol{\Theta})]^2. \qquad (3.34)
\end{aligned}$$

In summary, in order to compute the main effect of system parameters, the information required are: (i) variance of model parameters; (ii) expectation and variance of output measure; and (iii) conditional expectation and variance of output measure with respect to model parameters.

3.6 Numerical Experiments

In this section, we illustrate the variance-based sensitivity analysis based on the moment-based approximation via two experiments below to evaluate the main effects of model parameters, aiming at investigating the relationship between

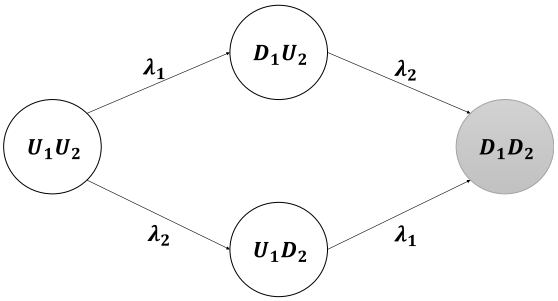

Figure 3.1: Parallel system with two single components.

input parameters and output measures and the application of the analytic approach. Note that the main effect here is the contribution of system parameters devoted to the system reliability with considering uncertainty. The experiments are divided as follows.

- Experiment I: Both parallel and series system configurations with two identical single components A and B, aiming at investigating the effects of a component in different configurations;
- Experiment II: A complex series-parallel system composed of five identical single components {A, B, C, D, E}, for the evaluation of the main effects of components in a complex system.

3.6.1 Experiment I

In this subsection, the failure rates of two components are given by $\lambda_1 = 1.0 \times 10^{-5}$ and $\lambda_2 = 2.0 \times 10^{-5}$, respectively. Note that the two systems contain the components with the same failure rates. For instance, component A in parallel system has the same failure rate as the one in series system. The dynamics of both parallel and series systems are captured by CTMCs, which are depicted in Figs. 3.1 and 3.2. In these figures, the white state means the system is available, whereas the gray state means the system is failed. Each state is denoted by two characters; if a component is available, its state is denoted by "U", otherwise it is "D". More details on the state notation are given in Tables 3.1 and 3.2.

To get the variance of failure rates, we apply the Bayes estimation as the estimation method. Basically, both the failure rates for components A and B are estimated by using data samples drawn from exponential distribution with pre-

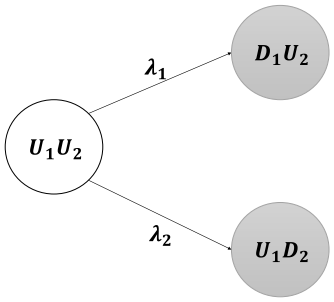

Figure 3.2: Series system with two single components.

Table 3.1: State notation in the CTMC (parallel system).

State	Description
U_1U_2	Both components A and B are operational.
D_1U_2	Component A is failed and the system is still available.
U_1D_2	Component B is failed and the system is still available.
D_1D_2	Both two components are failed and the system becomes unavailable.

Table 3.2: State notation in the CTMC (series system).

State	Description
U_1U_2	Both components A and B are operational.
D_1U_2	Component A is failed and the system becomes unavailable.
U_1D_2	Component B is failed and the system becomes unavailable.

set failure rates. For example, $\boldsymbol{D} = (t_1, \cdots, t_n)$ are defined as a set of observed failure times, which are identical and independently distributed (i.i.d.) samples drawn from exponential distribution. When the prior density of the failure rate is a gamma distribution with hyper parameters (α, β), the posterior distribution is also gamma with parameters $(\alpha + n, \beta + s)$ where $s = \sum_{i=1}^{n} t_i$. The mean and variance of estimated failure rates are $\hat{\lambda}_i = (\alpha + n)/(\beta + s)$ and $\mathrm{Var}[\hat{\lambda}_i] = (\alpha + n)/(\beta + s)^2$ for $i \in \{1, 2\}$. By applying the point estimation when sample size is 5, we obtain the variances of two components as $var1 = 9.5718 \times 10^{-12}$, $var2 = 6.7584 \times 10^{-11}$. The output measure we estimated in the experiments is the system reliability. When $t = 50$, the system reliability of parallel and series systems are $R_{parallel}(50; \lambda_1, \lambda_2) = 0.999999365$ and $R_{series}(50; \lambda_1, \lambda_2) = 0.998212759$. Because the parameters are independent from each other, for the main effect we can just consider the following relationship;

$$S_1 + S_2 = 1. \tag{3.35}$$

Table 3.3 demonstrates the main effects of two failure rates devoted to system reliability in both parallel and series configurations. From the table, it is clear that two components in the parallel system have the almost same contribution to the system reliability through main effect. However, in the case of series system, the effect of λ_1 is much larger than λ_2. This is because in the series configuration, the system failure occurs when any component fails, which means, the system reliability mainly depends on the component which has relatively higher reliability so that the large variation in the failure time of the component will affect remarkably the system reliability. On the other hand, since two systems have the components with the same failure rates, it is clarified that even if different systems are composed of the same components, components will make different contributions due to their different system structures. And in the parallel system, although the variance of two components has a relatively large difference it has no significant impact on main effect. But in the series system, the gap between the two components becomes more obvious. Based on the main effect, if we consider the uncertainty in parallel system, both the components should be considered. But for series system, we can give priority to the uncertainty analysis of component with relatively lower failure rate, to improve the system reliability effectively. Because the calculation results meet

Table 3.3: Main effects of failure rates in both parallel and series systems.

	Parallel	Series
λ_1	0.49962	0.87594
λ_2	0.50038	0.12406

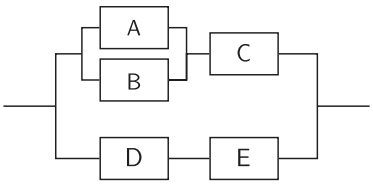

Figure 3.3: Series-parallel model with five single components.

the verification conditions of main effect, the formulaic method can be used normally and effectively.

In summary, the components with same failure rates in different configurations denote different contributions. When in the same configuration, component with large variance has more effects on system reliability.

3.6.2 Experiment II

Here we focus on a complex series-parallel system whose structure is depicted in Fig. 3.3. The system parameters are set as $\lambda_1 = 1.0 \times 10^{-5}$, $\lambda_2 = 2.0 \times 10^{-5}$, $\lambda_3 = 7.0 \times 10^{-5}$, $\lambda_4 = 6.0 \times 10^{-5}$, and $\lambda_5 = 5.0 \times 10^{-5}$, respectively. The dynamics of this system is also captured by the CTMC given in Fig. 3.4, with the state notation in Table 3.4. To inject uncertainty, the system parameters are estimated from data samples. The estimated variances are $var1 = 9.3047 \times 10^{-12}$, $var2 = 3.8422 \times 10^{-09}$, $var3 = 3.3522 \times 10^{-09}$, $var4 = 2.3712 \times 10^{-09}$, and $var5 = 5.6426 \times 10^{-10}$. Thus, the relationship of main effects changes to be;

$$S_1 + S_2 + S_3 + S_4 + S_5 = 1. \tag{3.36}$$

The main effects of model parameters for components are listed as in Table 3.5. From the table, it is obvious that component B has the largest value of main effect. Although series-parallel system contains the features of both parallel and series systems in configuration, the results of main effects with respect to parameters indicate that the contributions of components are different from

Table 3.4: State notation in the CTMC for the series-parallel system.

State	Description
$U_1U_2U_3U_4U_5$	All components are operational.
$D_1U_2U_3U_4U_5$	Component A is failed, system is still available.
$U_1D_2U_3U_4U_5$	Component B is failed, system is still available.
$U_1U_2D_3U_4U_5$	Component C is failed, system is still available.
$U_1U_2U_3D_4U_5$	Component D is failed, system is still available.
$U_1U_2U_3U_4D_5$	Component E is failed, system is still available.
$D_1D_2U_3U_4U_5$	Components A, B are failed, system is still available.
$D_1U_2D_3U_4U_5$	Components A, C are failed, system is still available.
$D_1U_2U_3D_4U_5$	Components A, D are failed, system is still available.
$D_1U_2U_3U_4D_5$	Components A, E are failed, system is still available.
$U_1D_2D_3U_4U_5$	Components B, C are failed, system is still available.
$U_1D_2U_3D_4U_5$	Components B, D are failed, system is still available.
$U_1D_2U_3U_4D_5$	Components B, E are failed, system is still available.
$U_1U_2D_3D_4U_5$	System is unavailable due to components C, D' failures.
$U_1U_2D_3U_4D_5$	System is unavailable due to components C, E' failures.
$D_1D_2D_3U_4U_5$	Components A, B, C failed, the system is still available.
$D_1D_2U_3D_4U_5$	System is unavailable due to components A, B, D' failures.
$D_1D_2U_3U_4D_5$	System is unavailable due to components A, B, E' failures.
$D_1U_2D_3D_4U_5$	System is unavailable due to components A, C, D' failures.
$D_1U_2D_3U_4D_5$	System is unavailable due to components A, C, E' failures.
$U_1D_2D_3D_4U_5$	System is unavailable due to components B, C, D' failures.
$U_1D_2D_3U_4D_5$	System is unavailable due to components B, C, E' failures.
$D_1D_2D_3D_4U_5$	System is unavailable due to components A, B, C, D' failures.
$D_1D_2D_3U_4D_5$	System is unavailable due to components A, B, C, E' failures.

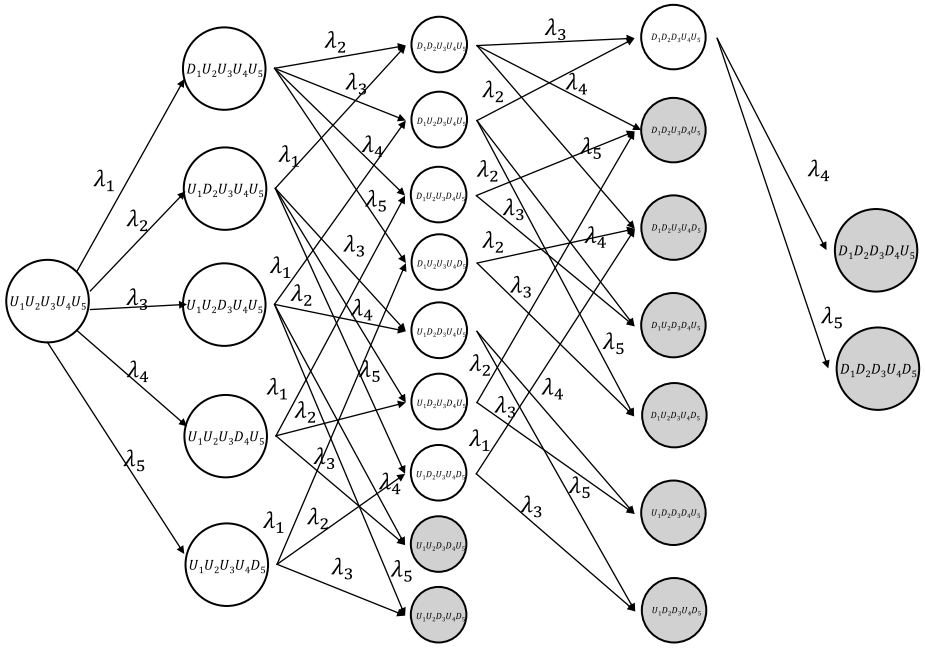

Figure 3.4: CTMC of Series-parallel system.

Table 3.5: Main effects of failure rates in series-parallel systems.

	Series-parallel
λ_1	0.00092
λ_2	0.37895
λ_3	0.33062
λ_4	0.23387
λ_5	0.05565

the single configuration. For example, although component A has a parallel relationship with component B, the main effect of component A devotes itself to having the smallest value. By comparing the variances of all parameters, we found component A with λ_1 has the smallest variance, which implies that the effects on the main effect are different from the situation in parallel system. Since component B has the largest value among all components, it should pay more attention to the estimation of λ_2 whose uncertainty will have a significant impact on the system reliability. Note that when the system consists of both series and parallel structures, the variance of components plays the central role in the final main effect. In other words, the variance of system parameters should be estimated well, especially for the components with large main effects. And combined with the results in experiment I, it is clear that components with the same failure rates have different contributions in the system with different structures, which means evaluating the main effects of model parameters is meaningful.

Chapter 4

Computation for Information Matrix of PH Distribution

4.1 Introduction

In the context of ML estimation [49], the Fisher information matrix is given by the second derivatives of log-likelihood function (LLF) in terms of parameters. The variance/covariance matrix is computed by the inverse of the Fisher information matrix. It is easy to obtain the Fisher information matrix when the model contains few parameters. But as we mentioned, PH fitting can bring a lot of parameters so that the calculation speed would be so low, and the computational cost would be extremely high. So, the motivation of this chapter is to improve the calculation speed of the Fisher information matrix with a large number of model parameters. In this chapter, we focus on the computation of the Fisher information matrix in PH fitting based on the uniformization method. The uniformization is to compute transient solutions of a finite-state CTMC by approximating the continuous process using a discrete-time Markov chain (DTMC) [50]. Also, this topic is discussed in other paper [43]. For brevity, the contributions of this chapter are summarized as follows.

- Computation of first and second derivatives of LLF in PH fitting with two types of information: IID samples and the probability density function (p.d.f.).
- Proposal of an efficient computation algorithm to enhance the computa-

tion speed of the information matrix.

4.2 Related Work

The ML estimation as one of the valuable methods contributes a lot to estimate PH parameters. The main idea of ML estimation is to estimate the parameters with maximum LLF. The LLF can be a function representing the probability, mass, or density of the observed data. Several papers discussed the ML estimation for PH distribution. Panchenko et al. [51] presented a PH fitting approach trying to aggregate the large data-trace to the small number of weighted samples based on ML estimation. And Bobbio et al. [52] attempted to obtain a standard form of ML estimation for PH distribution using optimization techniques. Another method called moment matching is also widely discussed. The major of moment matching is making moment equate match the empirical moment. But usually, the accuracy of moment matching would be based on the number of moments. In such a case, Osogami et al. [53] introduced a useful moment matching method based on three moments with a simple formula.

But unfortunately, the trade-off of accuracy and computation speed of PH fitting is difficult to avoid. One of the main issues is how to achieve PH fitting with high precision and speed. To address the above issue, the ML estimation is more worthy of consideration. Asmussen et al. [54] proposed an expectation-maximization (EM) algorithm for PH distributions. The EM algorithm is a general framework for calculating ML estimation under incomplete samples [55]. However, the proposed EM algorithm does not match the situation when there are many parameters. To overcome this problem, Okamura et al. [56] improved the EM algorithm of Asmussen et al. [54] from the aspect of scalability of the number of phases, which provides high computation speed for the estimation in the case of a large number of phases.

As abovementioned in Chapter 1, another aspect of improving the computation speed is improving the computation speed of the transient solution of the Markov chain. In reliability theory, the higher-order derivatives are like the sensitivity of the transient solution of CTMC. Ramesh et al. [41] considered the sensitivity of transient solutions of Markov models. Specifically, they provided a uniformization method for obtaining these transient solutions by converting

CTMC into DTMC. This chapter also applies the uniformization method to compute the information matrix in PH fitting efficiently.

4.3 PH Fitting

4.3.1 PH Distribution

The PH distribution is the distribution for an absorbing time in a finite Markov chain with an absorbing state. Basically, the classification of PH distribution would be continuous and discrete PH distributions. In this chapter, we focus on the continuous PH distribution. Without loss of generality, the infinitesimal generator $\boldsymbol{Q}$ of CTMC is defined as follows:

$$\boldsymbol{Q} = \left(\begin{array}{c|c} \boldsymbol{T} & \boldsymbol{\tau} \\ \hline \mathbf{0} & 0 \end{array} \right), \tag{4.1}$$

where $\boldsymbol{T}$ and $\boldsymbol{\tau}$ represent the transition rate between transients and the exit rates from the transient state to the absorbed state, respectively. Let $\boldsymbol{\alpha}$ be the initial probability vector of the transient states, the c.d.f. and p.d.f. of PH distribution can be obtained

$$F(t) = 1 - \boldsymbol{\alpha} \exp(\boldsymbol{T}t)\mathbf{1}, \quad f(t) = \boldsymbol{\alpha} \exp(\boldsymbol{T}t)\boldsymbol{\tau}, \tag{4.2}$$

where $\mathbf{1}$ is a column vector whose elements are all 1. The exit rate vector has the relationship can be obtained by $\boldsymbol{\tau} = -\boldsymbol{T}\mathbf{1}$. In particular, the transient states here are marked as phases.

There are several subclasses of PH distribution according to the structure of $\boldsymbol{T}$ (e.g., [57]). In particular, the acyclic PH distribution (APH) is the most extensive class of mathematically treatable PH distributions. Cumani [57] derived Canonical Forms (CFs) as the minimum representation of APH with the smallest number of free parameters. CF1 (canonical form 1) is defined as follows.

$$\boldsymbol{\alpha} = \begin{pmatrix} \alpha_1 & \alpha_2 & \cdots & \alpha_m \end{pmatrix}, \tag{4.3}$$

$$\boldsymbol{T} = \begin{pmatrix} -\beta_1 & \beta_1 & & & \boldsymbol{O} \\ & -\beta_2 & \beta_2 & & \\ & & \ddots & \ddots & \\ & & & -\beta_{m-1} & \beta_{m-1} \\ \boldsymbol{O} & & & & -\beta_m \end{pmatrix}, \tag{4.4}$$

where $\alpha_i \geq 0$, $\sum_i \alpha_i = 1$ and $0 < \beta_1 \leq \cdots \leq \beta_m$.

4.3.2 Parameter Estimation

The PH fitting is a parameter estimation of PH distribution so that it fits given information. In general, the loss function used for the estimation depends on the type of information. In this chapter, we introduce PH fitting with the following two types of information: IID samples and the p.d.f..

Suppose that $\mathcal{D} = \{t_1, t_2, \ldots, t_K\}$ are IID samples drawn from a general distribution. Without loss of generality, assuming that $0 = t_0 < t_1 < t_2 < \cdots < t_K$. The likelihood function is defined by

$$L(\boldsymbol{\alpha}, \boldsymbol{T}, \boldsymbol{\tau}; \boldsymbol{D}) = \prod_{n=1}^{N} f(t_n) = \prod_{n=1}^{N} \boldsymbol{\alpha} \exp(\boldsymbol{T} t_n) \boldsymbol{\tau}, \tag{4.5}$$

and the corresponding LLF is given by

$$\log L(\boldsymbol{\alpha}, \boldsymbol{T}, \boldsymbol{\tau}; \mathcal{D}) = \sum_{k=1}^{K} \log \boldsymbol{\alpha} \exp(\boldsymbol{T} t_k) \boldsymbol{\tau}. \tag{4.6}$$

The PH parameters can be obtained as the parameters maximizing the LLF.

In the case where the p.d.f. for a general distribution is explicitly given, we use Kullback-Leibler (KL) divergence to determine PH parameters. The KL divergence $KL(f, g)$ for p.d.f.'s $f(t)$ and $g(t)$ is given by

$$\begin{aligned} KL(f, g) &= \int_0^\infty f(t) \log \frac{f(t)}{g(t)} dt \\ &= \int_0^\infty f(t) \log f(t) dt - \int_0^\infty f(t) \log g(t) dt. \end{aligned} \tag{4.7}$$

In the context of PH fitting, $f(t)$ is a general distribution and $g(t)$ is a PH distribution approximating the general distribution. Then the purpose is finding $g(t)$ maximizing $\int_0^\infty f(t) \log g(t) dt$.

By applying a suitable numerical integration technique, we have the following relationship.

$$\int_0^\infty f(t) \log g(t) dt \approx \sum_{i=1}^{J} \nu_i f(t_i) \log g(t_i), \tag{4.8}$$

where w_i is a weight. The discretized points and their associated weights are determined by the numerical quadrature, e.g., the double exponential formula (see [58]). Eq. (4.8) implies that the PH parameters can be determined by

maximizing the LLF with the weighted samples $(t_1, \nu_1 f(t_i)), \ldots, (t_K, \nu_J f(t_J))$ [59].

Finally, to obtain concrete estimates of PH parameters, we use the algorithm presented in [59]. The algorithm in [59] is based on expectation-maximization (EM) algorithm for PH distribution which is originally discussed in [54]. The computation speed is further enhanced by using the sparsity of the generator matrix $\boldsymbol{T}$. Thus the algorithm in [59] can handle CF1 with a few hundred number of phases to provide the highly-accurate PH approximation.

4.4 Computation of Information Matrix in PH fitting

4.4.1 Uniformization

Before discussing the computation of the information matrix, we introduce numerical computation techniques for CTMC used in our algorithm.

Consider a CTMC with the infinitesimal generator $\boldsymbol{Q}$. Let $\boldsymbol{\pi}_0$ be the initial probability (row) vector and $\boldsymbol{\pi}(t)$ denote the transient probability vector at time t. Then $\boldsymbol{\pi}(t)$ can be computed by solving the following Kolmogorov differential equations

$$\frac{d\boldsymbol{\pi}(t)}{dt} = \boldsymbol{\pi}(t)\boldsymbol{Q}, \quad \boldsymbol{\pi}(0) = \boldsymbol{\pi}_0. \tag{4.9}$$

Equivalently, the expression with the matrix exponential becomes

$$\boldsymbol{\pi}(t) = \boldsymbol{\pi}_0 \exp(\boldsymbol{Q}t). \tag{4.10}$$

Uniformization is one of the methods used to perform the above transient analysis of CTMCs. In uniformization, the CTMC reduces to a DTMC with a uniformization rate q such that $q \geq \max_{i,j} |q_{i,j}|$ where $q_{i,j}$ is the (i,j)-th entry of $\boldsymbol{Q}$. Then the probability transition matrix of the uniformized DTMC can be written by

$$\boldsymbol{P} = \boldsymbol{I} + \frac{\boldsymbol{Q}}{q}, \tag{4.11}$$

where $\boldsymbol{I}$ is an identity matrix. The transient state probability vector at time t is given by

$$\boldsymbol{\pi}(t) = \pi_0 \sum_{i=0}^{\infty} \mathrm{Poi}(i; qt)\boldsymbol{P}^i, \tag{4.12}$$

where $\mathrm{Poi}(i; qt)$ is the Poisson distribution with mean qt. From the above formula, it can be seen that the computation time of uniformization is shorter as t is small.

Next, we consider the first and second derivatives of $\boldsymbol{\pi}(t)$ with respect to model parameters. For the simplification, we suppose that the infinitesimal generator $\boldsymbol{Q}$ and the initial vector $\boldsymbol{\pi}_0$ consisting of model parameters (θ_1, θ_2). By considering the derivative of Eq. (4.9) with respect to θ_1, we have

$$\frac{d}{dt}\frac{\partial}{\partial\theta_1}\boldsymbol{\pi}(t) = \frac{\partial}{\partial\theta_1}\boldsymbol{\pi}(t)\boldsymbol{Q} + \boldsymbol{\pi}(t)\frac{\partial}{\partial\theta_1}\boldsymbol{Q}, \quad \frac{\partial}{\partial\theta_1}\boldsymbol{\pi}(0) = \frac{\partial}{\partial\theta_1}\boldsymbol{\pi}_0. \tag{4.13}$$

Also the second derivative becomes

$$\begin{aligned}\frac{d}{dt}\frac{\partial^2}{\partial\theta_1\partial\theta_2}\boldsymbol{\pi}(t) =& \frac{\partial^2}{\partial\theta_1\partial\theta_2}\boldsymbol{\pi}(t)\boldsymbol{Q} + \frac{\partial}{\partial\theta_1}\boldsymbol{\pi}(t)\frac{\partial}{\partial\theta_2}\boldsymbol{Q} \\ &+ \frac{\partial}{\partial\theta_2}\boldsymbol{\pi}(t)\frac{\partial}{\partial\theta_1}\boldsymbol{Q} + \boldsymbol{\pi}(t)\frac{\partial^2}{\partial\theta_1\partial\theta_2}\boldsymbol{Q}, \quad \frac{\partial^2}{\partial\theta_1\partial\theta_2}\boldsymbol{\pi}(0) = \frac{\partial^2}{\partial\theta_1\partial\theta_2}\boldsymbol{\pi}_0\end{aligned} \tag{4.14}$$

Based on Eqs. (4.9), (4.13) and (4.14), we have the following differential equation

$$\frac{d}{dt}\boldsymbol{s}(t) = \boldsymbol{s}(t)\boldsymbol{\Lambda}, \quad \boldsymbol{s}(0) = \begin{pmatrix}\boldsymbol{\pi}_0 & \frac{\partial}{\partial\theta_1}\boldsymbol{\pi}_0 & \frac{\partial}{\partial\theta_2}\boldsymbol{\pi}_0 & \frac{\partial^2}{\partial\theta_1\partial\theta_2}\boldsymbol{\pi}_0\end{pmatrix}, \tag{4.15}$$

where

$$\boldsymbol{s}(t) = \begin{pmatrix}\boldsymbol{\pi}(t) & \frac{\partial}{\partial\theta_1}\boldsymbol{\pi}(t) & \frac{\partial}{\partial\theta_2}\boldsymbol{\pi}(t) & \frac{\partial^2}{\partial\theta_1\partial\theta_2}\boldsymbol{\pi}(t)\end{pmatrix}, \tag{4.16}$$

$$\boldsymbol{\Lambda} = \begin{pmatrix}\boldsymbol{Q} & \frac{\partial}{\partial\theta_1}\boldsymbol{Q} & \frac{\partial}{\partial\theta_2}\boldsymbol{Q} & \frac{\partial^2}{\partial\theta_1\partial\theta_2}\boldsymbol{Q} \\ & \boldsymbol{Q} & & \frac{\partial}{\partial\theta_2}\boldsymbol{Q} \\ & & \boldsymbol{Q} & \frac{\partial}{\partial\theta_1}\boldsymbol{Q} \\ & & & \boldsymbol{Q}\end{pmatrix}. \tag{4.17}$$

The above differential equation is similar to Eq. (4.9). Thus we can apply the uniformization to obtain the first and second derivatives of $\boldsymbol{\pi}(t)$ concerning model parameters.

4.4.2 Computation of LLF and Fisher Information Matrix

Here we present the computation algorithm of the Fisher information matrix in PH fitting. In the context of ML estimation, the Fisher information matrix is often given by the second derivatives of LLF. In other words, the computation of LLF is fundamental even for calculating the Fisher information matrix.

Consider the computation of LLF with weighed IID samples. The computation of LLF with IID samples can be obtained by substituting 1 into all the weights. Suppose that $\mathcal{D} = \{(t_1, w_1), (t_2, w_2), \ldots, (t_K, w_K)\}$ are weighted IID samples where $0 = t_0 < t_1 < t_2 < \cdots < t_K$ are ordered time points and $w_1, \ldots, w_K$ are corresponding weights. Then the LLF is defined by

$$\log L(\boldsymbol{\alpha}, \boldsymbol{T}, \boldsymbol{\tau}; \boldsymbol{D}) = \sum_{k=1}^{K} w_k \log \boldsymbol{\alpha} \exp(\boldsymbol{T} t_k) \boldsymbol{\tau}. \tag{4.18}$$

As shown in the uniformization, the computation cost becomes less if t_n is short. Therefore, the chapter focuses on the time difference between two time points, i.e., $\Delta t_k = t_k - t_{k-1}$. Then the LLF can be rewritten by

$$\log L(\boldsymbol{\alpha}, \boldsymbol{T}, \boldsymbol{\tau}; \mathcal{D}) = \sum_{k=1}^{K} w_k \log \boldsymbol{f}_k \boldsymbol{\tau}, \tag{4.19}$$

where $\boldsymbol{f}_0 = \boldsymbol{\alpha}$, $\boldsymbol{f}_k = \boldsymbol{f}_{k-1} \exp(\boldsymbol{T} \Delta t_k)$. Based on this formula, the pseudo-code for the computation of LLF with uniformization is given as follows.

```
1: llf = 0
2: f[0] = alpha
3: for k = 1:K
4:     f[k] = unif(f[k-1], T, Delta[k])
5:     llf += w[k] * log(f[k] * tau)
6: end
```

where `alpha`, `T`, `tau`, `w[k]`, `Delta[k]` are, $\boldsymbol{\alpha}$, $\boldsymbol{T}$, $\boldsymbol{\tau}$, w_k, Δt_k. Also, the function `unif(f[k-1], T, Delta[k])` corresponds to the uniformization algorithm to compute $\boldsymbol{f}_k = \boldsymbol{f}_{k-1} \exp(\boldsymbol{T} \Delta t_k)$.

Next we present the computation algorithm for the information matrix, i.e., the second derivatives of LLF.

$$\begin{aligned}
&\frac{\partial^2}{\partial \theta_i \partial \theta_j} \log L(\boldsymbol{\alpha}, \boldsymbol{T}, \boldsymbol{\tau}; \mathcal{D}) \\
&\quad = \sum_{k=1}^{K} w_k \frac{\frac{\partial^2}{\partial \theta_i \partial \theta_j} \boldsymbol{f}_k \boldsymbol{\tau} + \frac{\partial}{\partial \theta_i} \boldsymbol{f}_k \frac{\partial}{\partial \theta_j} \boldsymbol{\tau} + \frac{\partial}{\partial \theta_j} \boldsymbol{f}_k \frac{\partial}{\partial \theta_i} \boldsymbol{\tau} + \boldsymbol{f}_k \frac{\partial^2}{\partial \theta_i \partial \theta_j} \boldsymbol{\tau}}{\boldsymbol{f}_k \boldsymbol{\tau}} \\
&\quad - \sum_{k=1}^{K} w_k \frac{(\frac{\partial}{\partial \theta_i} \boldsymbol{f}_k \boldsymbol{\tau} + \boldsymbol{f}_k \frac{\partial}{\partial \theta_i} \boldsymbol{\tau})(\frac{\partial}{\partial \theta_j} \boldsymbol{f}_k \boldsymbol{\tau} + \boldsymbol{f}_k \frac{\partial}{\partial \theta_j} \boldsymbol{\tau})}{(\boldsymbol{f}_k \boldsymbol{\tau})^2}.
\end{aligned} \tag{4.20}$$

Note that the first and second derivatives of $\boldsymbol{f}_k$ can be computed with Eq. (4.15).

Finally, the pseudo-code for the computation of the information matrix is given as follows.

```
1: llf = 0
2: f[0] = alpha
3: for i = 1:n
4:     s[i] = 0;  df[0][i] = dalpha[i]
5:     for j = 1:i
6:         F[i,j] = 0;  ddf[0][i,j] = ddalpha[i,j]
7:     end
8: end
9: f[0], df[0], ddf[0] = alpha, dalpha, ddalpha
10: for k = 1:K
11:     f[k], df[k], ddf[k] = unif(f[k-1],df[k-1],ddf[k-1); Delta[k])
12:     llf += w[k] * log(f[k] * tau)
13:     for i = 1:n
14:         s[i] += w[k] * (f[k] * dtau[i] + df[i][k] * tau) \
/ (f[k] * tau)
15:         for j = 1:i
16:             F[i,j] += w[k] * (ddf[k][i,j] * tau \
+ df[k][i] * dtau[j] + df[k][j] * dtau[i] \
+ f[k] * ddtau[i,j]) / (f[k] * tau) \
- w[k] * (df[k][i] * tau + f[k] * dtau[i]) \
* (df[k][j] * tau + f[k] * dtau[j]) \
/ (f[k] * tau)^2
17:         end
18:     end
19: end
```

where `dalpha`, `ddalpha`, `dtau`, `ddtau` are the first and second derivatives of the initial probability vector $\boldsymbol{\alpha}$ and the exit vector $\boldsymbol{\tau}$. It should be noted that they are an array of vectors. Also `s` and `F` are the first and second derivatives of LLF, i.e., the score function and the information matrix. In the algorithm, the function `unif(f[k-1], df[k-1], ddf[k-1]; Delta[k])` is to provide the first and second derivatives of $\boldsymbol{f}_k$ from the first and second derivatives of $\boldsymbol{f}_{k-1}$.

In this function, the next step vectors can be computed by using the uniformed matrix consisting of the first and second derivatives of $\boldsymbol{T}$ like Eq. (4.17).

4.5 Numerical Experiments

In this section, we evaluate the computational time for calculating the information matrix in PH fitting with two types of information: IID samples and the p.d.f., by using the uniformization method. In the experiment, the IID samples of observed data are drawn from Weibull distribution with the probability density function as follows:

$$f(t;\beta,\eta) = \frac{\beta}{\eta}(\frac{x}{\eta})^{\beta-1}e^{-(\frac{x}{\eta})^{\beta}}, \tag{4.21}$$

where $\beta > 0$ and $\eta > 0$ are the shape parameter and scale parameter, respectively. In particular, we set $\beta = 1.5$, $\eta = 1.0$ and first generate 10 samples from Weibull distribution.

To investigate the effect of the number of phases on the computational time, two cases of phases are considered, i.e., phases are 2 and 4. As an example, when the number of phases is 2, the corresponding PH parameters after PH fitting are

$$\boldsymbol{\alpha} = \begin{pmatrix} \alpha_1 & 1-\alpha_1 \end{pmatrix}, \quad \boldsymbol{T} = \begin{pmatrix} -\beta_1 & \beta_1 \\ 0 & -\beta_2 \end{pmatrix}, \tag{4.22}$$

where $\boldsymbol{\alpha}$ is the initial probability vector of transition states and $\boldsymbol{T}$ is the infinitesimal generator of the underlying CTMC. Then, according to Eq. (4.20), letting

$$\boldsymbol{I}(\boldsymbol{\theta}) = \begin{pmatrix} \frac{\partial^2}{\partial\alpha_1^2}\log L(\boldsymbol{\alpha},\boldsymbol{T},\boldsymbol{\tau};\mathcal{D}) & \frac{\partial^2}{\partial\alpha_1\partial\beta_1}\log L(\boldsymbol{\alpha},\boldsymbol{T},\boldsymbol{\tau};\mathcal{D}) & \frac{\partial^2}{\partial\alpha_1\partial\beta_2}\log L(\boldsymbol{\alpha},\boldsymbol{T},\boldsymbol{\tau};\mathcal{D}) \\ \frac{\partial^2}{\partial\beta_1\partial\alpha_1}\log L(\boldsymbol{\alpha},\boldsymbol{T},\boldsymbol{\tau};\mathcal{D}) & \frac{\partial^2}{\partial\beta_1^2}\log L(\boldsymbol{\alpha},\boldsymbol{T},\boldsymbol{\tau};\mathcal{D}) & \frac{\partial^2}{\partial\beta_1\partial\beta_2}\log L(\boldsymbol{\alpha},\boldsymbol{T},\boldsymbol{\tau};\mathcal{D}) \\ \frac{\partial^2}{\partial\beta_2\partial\alpha_1}\log L(\boldsymbol{\alpha},\boldsymbol{T},\boldsymbol{\tau};\mathcal{D}) & \frac{\partial^2}{\partial\beta_2\partial\beta_1}\log L(\boldsymbol{\alpha},\boldsymbol{T},\boldsymbol{\tau};\mathcal{D}) & \frac{\partial^2}{\partial\beta_2^2}\log L(\boldsymbol{\alpha},\boldsymbol{T},\boldsymbol{\tau};\mathcal{D}) \end{pmatrix}, \tag{4.23}$$

the information matrix can be obtained by $\boldsymbol{I}(\boldsymbol{\theta})$, where $\boldsymbol{\theta} = \{\alpha_1, \beta_1, \beta_2\}$ and $\boldsymbol{A}$ is the inverse of matrix $\boldsymbol{I}(\boldsymbol{\theta})$. The variance of θ_i corresponds to the (i,i)-element of $\boldsymbol{A}$ while the covariances of θ_i and θ_j corresponds to the (i,j)-element. We

Table 4.1: Comparison of computational time between common method and uniformization-based method.

#. of phases	Common method	Uniformization
2	0.049969	0.002384
4	8.153741	0.005044

Table 4.2: Computational time of information matrix in PH fitting (5 phases).

# of samples	IID samples		the p.d.f.	
	first	second	first	second
50	0.000552	0.014600	0.005391	0.121645
100	0.000941	0.026586	0.008195	0.122906
500	0.004028	0.098346	0.030713	0.129557
1000	0.006594	0.174501	0.058552	0.130411

Table 4.3: Computational time of information matrix in PH fitting (10 phases).

# of samples	IID samples		the p.d.f.	
	first	second	first	second
50	0.001558	0.082790	0.029265	1.031466
100	0.002347	0.118382	0.039976	0.995568
500	0.010192	0.449630	0.108468	0.992346
1000	0.026296	0.807612	0.196954	0.994312

estimate the computational time based on common method and uniformization-based method. The main idea of common method is computing the second derivative of LLF concerning PH parameters by using matrix exponential (see Eq. (4.10)). The unit of computational time is sec in the experiment. The results of the Comparison are shown in Table 4.1. From the table, it can be seen that the uniformization-based method outperforms the common method with lower computational time, especially in the case of 4 phases, the computational time of uniformization-based method is about 0.005 seconds (s), which is only 0.061% of that of common method. This shows the uniformization-based method improved a lot of computational speed.

Next, to evaluate the effect of the number of data samples on the compu-

tational time of the uniformization-based method, we generate 50,100,500,1000 data samples from Weibull distribution for both 5 and 10 phases cases. Table 4.2 shows the results of computational time of calculating information matrix in the case of 5 phases, while Table 4.3 gives the results when the number of phases is 10. From these tables, first and second represent the computational time of the first derivative and second derivative of the information matrix, respectively. Specifically, for IID samples, in the case of 5 phases, the computational time is much lower than that in the case of 10 phases. And the computational time increases with the number of samples. At the same time, it takes more time to calculate the second partial derivative, compared with the first partial derivative. On the other hand, for the p.d.f., the computational time becomes much larger than that of IID samples case. However, with the increase in the number of data samples, the computational time of the second derivative of the p.d.f. was almost the same. Since the computation involves the PH fitting using EM algorithm, the above results indicate that the EM algorithm is more stable in finding quadratic partial derivatives. In addition, it is obvious that the computational time grows when the number of phases increases.

Chapter 5

Conclusion

This dissertation contains four folds, around the the effects and the evaluation of uncertainty propagation, we discussed the uncertainty propagation in fault tree and hierarchical reliability models, respectively. From a point view of variance-based sensitivity analysis, we proposed an analytical approach to evaluate the uncertainty in the model parameters through the variance from data samples. To evaluate the uncertainty in the non-exponential stochastic models, the PH fitting technique is considered for approximation and an efficient algorithm for efficiently computing the Fisher information matrix is proposed for getting the variance and covariance of estimators.

In Chapter 2, we introduced a moment-based approximation method for evaluating epistemic uncertainty propagation in hierarchical reliability models. The moment-based approximation requires the first and second derivatives of system dependability measure with respect to model parameters, but obtaining them for an FT in practice is not easy. To overcome such a limit, the BDD-based computation for the first and second derivatives of the FTs and the hierarchical models was presented. Two experiments with both simple and complicated systems were conducted to validate and evaluate the proposed moment-based approximation method. Specifically, Bayes estimation helped validate the effectiveness, and MC simulation was used to evaluate the performance of the proposed approach. Numerical results showed that the moment-based approximation not only achieved good performance with high accuracy but also significantly improved the computational speed compared with the existing approaches. In the future, we will automate the calculation of the proposed approach and provide

an efficient tool to assess the uncertainty propagation in hierarchical reliability models.

In Chapter 3, the variance-based sensitivity analysis for CTMCs was discussed. Since the sensitivity analysis requires the formulation of the target model and the CTMC model seldom has the formulation method, thus the moment-based approximation method was applied. With the aid of moment-based approximation, we provided an explicit formulation of variance-based sensitivity analysis and obtained the main effects of components by decomposing the variance of the model output into fractions attributed to inputs. Our first experiment illustrated the main effects of components with the same failure rates in parallel/series systems and the results showed that the same components with different system structures can devote different contributions to the system reliability. The second experiment indicated that the proposed approach can also be applied to complex system. In particular, components in the system composed of both series and parallel structure own the relationship different from the single parallel or series system. To figure out the contribution of each component, the importance of system parameters can be judged via their main effects, so as to know the components that need to be focused on. The importance rank of components in terms of main effects can bring insight on the effective system reliability improvement. At the same time, if we consider the impact of parameter uncertainty on the system reliability, we can also give priority to the components with high contribution. In the future, we will apply this formulation method to a real system to consider the uncertainty propagation.

In Chapter 4, we proposed an efficient algorithm for computing the information matrix in PH fitting using uniformization. In particular, two types of PH fitting were considered; that is, PH parameters were estimated from IID samples and the p.d.f. information, respectively. The uniformization method was applied for simplifying the computation process of the matrix exponential of the underlying CTMC of PH distribution. In the experiments, we compared the computational time between the common method and the uniformization-based method. To investigate the effects of the number of phases on the computational time, two cases with different numbers of phases were taken into account. In particular, we further evaluated the effect of the number of data samples. Our

numerical results showed that the uniformization-based method improved a lot of the computational speed and can be considered an efficient and promising method for computing the information matrix in the PH fitting. However, the data samples we used in this paper were all point data. Actually, in addition to point data, the group data are also widely used for PH fitting. In the case of group data, the situation will be different. In the future, we will consider the computation of the information matrix based on group data.

The main contributions of this dissertation are proposing several analytical methods of solving uncertainty propagation. We initially considered the uncertainty propagation in the FT, using the BDD representation help the computation of first and second derivatives. As the extension of the previous topic, we discussed the uncertainty propagation in the hierarchical reliability models through the moment-based approximation. Though the experiments, the accuracy of moment-based approximation had been proven and the computation efficiency is proved better than simulation method especially when the number of parameters is small.

Bibliography

[1] K. S. Trivedi and A. Bobbio, Reliability and Availability Engineering, Modeling, Analysis, and Applications, Cambridge University Press, 2017.

[2] M. Stamatelatos, G. Apostolakis, H. Dezfuli, C. Everline, S. Guarro, P. Moeini, A. Mosleh, T. Paulos, and R. Youngblood, "Probabilistic risk assessment procedures guide for nasa managers and practitioners," in *http://www.hq.nasa.gov/office/codeq/doctree/praguide.pdf*, 2011.

[3] A. Volkanovski, "Impact of Component Unavailability Uncertainty on Safety Systems Unavailability," *Nuclear Engineering and Design*, vol. 283, pp. 193–201, 2015.

[4] J. Lei and W. Schilling, "Parameter uncertainty propagation analysis for urban rainfall runoff modelling," *Water Science and Technology*, vol. 29, no. (1–2), pp. 145–154, 1994.

[5] L. Yin, M. Smith, and K. S. Trivedi, "Uncertainty analysis in reliability modeling," in *Proceedings of International Symposium on Product Quality and Integrity*, IEEE, pp. 229–234, 2001.

[6] C. D. S. Jean, G. Noguere, B. Habert, and B. Looss, "A Monte Carlo approach to nuclear model parameter uncertainties," *Nuclear Engineering and Design*, vol. 161, no. 3, pp. 363–370, 2009.

[7] K. Mishra and K. S. Trivedi, "Uncertainty propagation through software dependability models," in *Proceedings of 2011 IEEE 22nd International Symposium on Software Reliability Engineering*, IEEE, pp. 80–89, 2011.

[8] K. Mishra, K. S. Trivedi and R. R. Some, "Uncertainty analysis of the remote exploration and experimentation system," *Journal of Spacecraft and Rockets*, vol. 49, pp. 1032–1042, 2012.

[9] J. Zhang, J. Zheng, H. Okamura and T. Dohi, "Uncertainty propagation in importance analysis of fault trees," in *Proceedings of the 26th ISSAT International Conference on Reliability and Quality in Design (ISSAT RQD 2021)*, 2021, 5 pages.

[10] H. Okamura, T. Dohi, and K. S. Trivedi, 2018. "Parametric uncertainty propagation through dependability models," in *Proceedings of 2018 Eighth Latin-American Symposium on Dependable Computing (LADC '18)*, IEEE, pp. 10–18, 2018.

[11] K. S. Trivedi, D. Wang, D. J. Hunt, A. Rindos, W. E. Smith and B. Vashaw, "Availability modeling of SIP protocol on IBM© Websphere©," in *Proceedings of 2008 14th IEEE Pacific Rim International Symposium on Dependable Computing (PRDC '08)*, IEEE, pp. 323–330, 2008.

[12] J. Zhang, J. Zheng, H. Okamura, T. Dohi and K. S. Trivedi, "Moment-Based Approximation for Epistemic Uncertainty Propagation in Hierarchical Reliability Models," *IEEE Transactions on Reliability*, under review, 2022.

[13] S. Marzban and T. Lahmer, "Conceptual implementation of the variance-based sensitivity analysis for the calculation of the first-order effects," *Journal of Statistical Theory and Practice*, vol. 10, no. 4, pp. 589–611, 2016.

[14] Y. Gan, Q. Duan, W. Gong, C. Tong, Y. Sun, W. Chu, A. Ye, C. Miao and Z. Di, "A comprehensive evaluation of various sensitivity analysis methods: A case study with a hydrological model," *Environmental Modelling & Software*, vol. 51, pp. 269–285, 2014.

[15] J. Zheng, H. Okamura, and T. Dohi, 2015. "Availability importance measures for virtualized system with live migration," *Applied Mathematics*, vol. 6, pp. 359–372, 2015.

[16] N. D. Singpurwalla, *Reliability and Risk: A Bayesian Perspective*, John Wiley & Sons, 1 edition, 2006.

[17] K. S. Trivedi and R. Sahner, "Sharpe at the age of twenty two," *ACM SIGMETRICS Performance Evaluation Review*, vol. 36, no. 4, pp. 52–57, 2009.

[18] C. Hirel, B. Tuffin and K. S. Trivedi, "Spnp: Stochastic petri nets. version 6.0.," in *International Conference on Modelling Techniques and Tools for Computer Performance Evaluation.*, Springer Berlin / Heidelberg, vol. 1786, pp. 354–357, 2000.

[19] A. Devaraj, K. Mishra and K. S. Trivedi, "Uncertainty propagation in analytic availability models," in *Proceedings of 2010 29th IEEE Symposium on Reliable Distributed Systems*, IEEE, pp. 121–130, 2010.

[20] K. Mishra and K. S. Trivedi, "A non-obtrusive method for uncertainty propagation in analytic dependability models," in *Proceedings of 2011 IEEE 22nd International Symposium on Software Reliability Engineering*, 2010.

[21] B. Harverkort and A. M. H. Meeusissen, "Sensitivity and uncertainty analysis of Markov-reward models," *IEEE Transaction on Reliability*, vol. 44, no. 1, pp. 147–154, 1995.

[22] T. K. Sarkar, "An exact lower confidence bound for the reliability of a series system where each component has an exponential time to failure," *Technometrics*, vol. 13, no. 3, pp. 535–546, 1971.

[23] G. J. Leiberman and S. M. Ross, "Confidence intervals for independent exponential series systems," *Journal of the American Statistical Association*, vol.66, no. 336, pp. 837–840, 1971.

[24] D. W. Coit, "System reliability confidence intervals for complex systems with estimated component reliability," *IEEE Transactions on Reliability*, vol. 46, no. 4, pp. 487–493, 1997.

[25] R. Pinciroli, K. S. Trivedi and A. Bobbio, "Parametric sensitivity and uncertainty propagation in dependability models," in *Proceedings of 10th EAI International Conference on Performance Evaluation Methodologies and Tools*, pp. 44–51, 2017.

[26] M. Gribaudo, R. Pinciroli and K. S. Trivedi, "Epistemic uncertainty propagation in power models," *Electronic Notes in Theoretical Computer Science*, vol. 337, pp. 67–86, 2018.

[27] R. Pinciroli, A. Bobbio, C. Boichini, D. Cerotti, M. Gribaudo, A. Miele and K. S. Trivedi, "Epistemic uncertainty propagation in a weibull environment for a two-core system-on-chip," in *Proceedings of 2017 2nd International Conference on System Reliability and Safety (ICSRS)*, IEEE, pp. 516–520, 2017.

[28] G. J. Jiang and Z. Liu, Z. Y. Li , G. Qiao , H. X. Chen, H. B. Li and H. H. Sun, "Reliability Analysis of Dynamic Fault Tree Based on Binary Decision Diagrams for Explosive Vehicle," *Mathematical Problems in Engineering*, vol. 2021, 2021.

[29] P. Glasserman and Z. Liu, "Sensitivity estimates from characteristic functions," *Operations Research*, vol. 58, no. 6, pp. 932–940, 2010.

[30] S. Asmussen and P. W. Glynn, *Stochastic Simulation: Algorithms and Analysis*, Springer, 2007.

[31] M. Jerrum and A. Sinclair, "The Markov chain Monte Carlo method: an approach to approximate counting and integration," *Approximation Algorithms for NP-hard problems*, PWS Publishing, 1996.

[32] J. C. Helton and F. J. Davis, "Latin hypercube sampling and propagation of uncertainty in analysis of complex systems," *Reliability Engineering & System Safety*, vol. 81, no. 1, pp. 23–69, 2003.

[33] J. C. Helton, J. D. Johnson, C. J. Sallaberry and C. B. Storlie, "Survey of sampling-based methods for uncertainty and sensitivity analysis," *Reliability Engineering & System Safety*, vol. 91, no. 10–11, pp. 1175–1209, 2006.

[34] Y. Pinto, A. R. van der Leij, I. G. Sligte, V. A. Lamme and H. S. Scholte, "Bottom-up and top-down attention are independent," *Journal of vision*, vol. 13, no. 3, pp. 16-16, 2013.

[35] B. Akers, "Binary decision diagrams," *IEEE Transactions on computers*, vol. C-27, no. 6, pp. 509–516, 1978.

[36] M. Grottke and K. S. Trivedi, "Fighting bugs: Remove, retry, replicate, and rejuvenation," *IEEE Computer* vol. 40, no. 2, pp. 107–109, 2007.

[37] K. S. Trivedi, *Probability and Statistics with Reliability, Queuing and Computer Science Applications*, J. Wiley & Sons, New York, 2001.

[38] J. Zhang, J. Zheng, H. Okamura and T. Dohi, "Moment-based approximation for uncertainty propagation in fault trees," in *Proceedings of 2019 IEEE 24th Pacific Rim International Symposium on Dependable Computing (PRDC 2019)*, IEEE, 2019, pp. 120–121.

[39] J. Zhang, J. Zheng, H. Okamura and T. Dohi, "Variance-Based Sensitivity Analysis for Markov Models using Moment Approximation," *International Journal of Performability Engineering*, vol. 18, no. 3, pp. 329-337, May 2022.

[40] P. Jadun, L. J. Vimmerstedt, B. W. Bush, D. Inman and S. Peterson, "Application of a variance-based sensitivity analysis method to the Biomass Scenario Learning model," *System Dynamics Review,* vol. 33, no. 3-4, pp. 311–335, 2017.

[41] A. V. Ramesh and K. S. Trivedi, "On the sensitivity of transient solutions of Markov models," in *Proceedings of 1993 ACM SIGMETRICS Conference on Measurement and Modeling of Computer Systems*, 1993, pp. 122–134.

[42] J. T. Blake, A. L. Reibman and K. S. Trivedi, "Sensitivity analysis of reliability and performability measures for multiprocessor systems," in *Proceedings of 1988 ACM SIGMETRICS Conference on Measurement and Modeling of Computer Systems*, 1988, pp. 177–186.

[43] J. Zhang, J. Zheng, H. Okamura and T. Dohi, "An efficient algorithm for computation of information matrix in phase-type fitting," *International Journal for Computational Methods in Engineering Science and Mechanics*, vol. 22, iss. 3, pp. 193-199, April 2021.

[44] A. Bobbio and M. Telek, "A benchmark for PH estimation algorithms: result for acyclic-PH," *Stochastic models*, vol. 10, no. 3, pp. 661–677, 1994. DOI: 10.1080/15326349408807315

[45] S. Asmussen, P. J. Laub and H. Yang, "Phase-Type Models in Life Insurance: Fitting and Valuation of Equity-Linked Benefits," *Risks*, vol. 7, no. 1, pp. 17, 2019. DOI: 10.3390/risks7010017

[46] H. Okamura and T. Dohi, "Ph fitting algorithm and its application to reliability engineering," *Journal of the Operations Research Society of Japan*, vol. 59, no. 1, pp. 72-109, 2016. DOI: 10.15807/jorsj.59.72

[47] S. McClean, J. Gillespie, L. Garg, M. Barton, B. Scotney and K. Kullerton, "Using phase-type models to cost stroke patient care across health, social and community services," *European Journal of Operational Research*, vol. 236, no. 1, pp. 190-199, 2014. DOI: 10.1016/j.ejor.2014.01.063

[48] S. Retout and F. Mentré, "Further developments of the Fisher information matrix in nonlinear mixed effects models with evaluation in population pharmacokinetics," *Journal of Biopharmaceutical Statistics*, vol. 13, no. 2, pp. 209-227, 2003. DOI: 10.1081/BIP-120019267

[49] W. A. Scott, "Maximum likelihood estimation using the empirical fisher information matrix," *Journal of Statistical Computation and Simulation*, vol. 72, no. 8, pp. 599-611, 2002. DOI: 10.1080/00949650213744

[50] O. Ibe, *Markov Processes for Stochastic Modeling*. Elsevier, 2013.

[51] A. Panchenko and A. Thümmler, "Efficient phase-type fitting with aggregated traffic traces," *Performance Evaluation*, vol. 64, no. 7-8, pp. 629–645, 2007. DOI: 10.1016/j.peva.2006.09.002

[52] A. Bobbio and A. Cumani, "ML estimation of the parameters of a PH distribution in triangular canonical form," *Computer performance evaluation*, vol. 22, pp. 33–46, 1992.

[53] T. Osogami and M. Harchol-Balter, "Closed form solutions for mapping general distributions to quasi-minimal PH distributions," *Per-*

formance Evaluation, vol. 63, no. 6, pp. 524–552, 2006. DOI: 10.1016/j.peva.2005.06.002

[54] S. Asmussen, O. Nerman and M. Olsson, "Fitting phase-type distributions via the EM algorithm," *Scandinavian Journal of Statistics*, vol. 23, no. 4, pp. 419–441, 1996.

[55] A. P. Dempster, N. M. Laird and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 39, no. 1, pp. 1–22, 1977. DOI: 10.1111/j.2517-6161.1977.tb01600.x

[56] H. Okamura, T. Dohi and K. S. Trivedi, "A refined EM algorithm for PH distributions," *Performance Evaluation*, vol. 68, no. 10, pp. 938–954, 2011. DOI: 10.1016/j.peva.2011.04.001

[57] A. Cumani, "On the canonical representation of homogeneous Markov processes modelling failure-time distributions," *Microelectronics Reliability*, vol. 22, no. 3, pp. 583–602, 1982. DOI: 10.1016/0026-2714(82)90033-6

[58] H. Takahasi and M. Mori, "Double exponential formulas for numerical integration," *Publications of the Research Institute for Mathematical Sciences*, vol. 9, no. 3, pp. 721–741, 1974. DOI: 10.2977/prims/1195192451

[59] H. Okamura, Y. Kamahara and T. Dohi, "Estimating Markov-modulated compound Poisson processes," in *Proceedings of 2nd International ICST Conference on Performance Evaluation Methodologies and Tools*, 2007, pp. 1–8. DOI: 10.1145/1345263.1345299

Publication List of the Author

Publications in this dissertation

[J-1] J. Zhang, J. Zheng, H. Okamura and T. Dohi, "An efficient algorithm for computation of information matrix in phase-type fitting," International Journal for Computational Methods in Engineering Science and Mechanics, vol. 22, iss. 3, pp. 193-199, April 2021.

[J-2] J. Zhang, J. Zheng, H. Okamura and T. Dohi, "Variance-Based Sensitivity Analysis for Markov Models using Moment Approximation," International Journal of Performability Engineering, vol. 18, no. 3, pp. 329-337, May 2022.

Referred Conferences

[C-1] J. Zhang, J. Zheng, H. Okamura and T. Dohi, "Moment-based approximation for uncertainty propagation in fault trees," in Proceedings of the 24th Pacific Rim International Symposium on Dependable Computing (PRDC 2019), IEEE CPS, 2019, pp. 120-121.

[C-2] J. Zhang, J. Zheng, H. Okamura and T. Dohi, "Uncertainty propagation in importance analysis of fault trees," in Proceedings of the 26th ISSAT International Conference on Reliability and Quality in Design (ISSAT RQD 2021), 2021, 5 pages.

[C-3] J. Zhang, J. Zheng, H. Okamura and T. Dohi, "Uncertainty analysis of virtualized systems with live migration," in Proceedings of the 11th

International Conference on Quality, Reliability, Risk, Maintenance, and Safety Engineering (QR2MSE 2021), 2022, 5 pages.

[C-4] J. Zhang, J. Zheng, H. Okamura and T. Dohi, "A Note on Epistemic Uncertainty Propagation in Generalized Stochastic Petri Nets," in Proceedings of the 10th Asia Pacific International Symposium on Advanced Reliability and Maintenance Modeling (APARM 2022), 2022, 5 pages.