RELIABILITY ESTIMATION OF SYSTEMS WITH SPATIALLY DISTRIBUTED UNITS

By

DINGGUO HUA

A dissertation submitted to the

Graduate School-New Brunswick

Rutgers, The State University of New Jersey

In partial fulfillment of the requirements

For the degree of

Doctor of Philosophy

Graduate Program in Industrial and Systems Engineering

Written under the direction of

Elsayed A. Elsayed

And approved by

_____

_____

_____

_____

New Brunswick, New Jersey

OCTOBER, 2016

ABSTRACT OF THE DISSERTATION

Reliability Estimation of Systems with Spatially Distributed Units

By DINGGUO HUA

Dissertation Director:

Elsayed A. Elsayed

Systems with spatially distributed units, e.g. Unmanned Aerial Vehicle (UAV), are emerging in aerospace and military industries. In this dissertation, we present approaches for the reliability estimation of such systems. In particular, we consider *k*-out-of-*n* pairs:G Balanced systems and weighted-*c*-out-of-*n* pairs:G Balanced systems with spatially distributed units which must meet balance requirements.

We first estimate the reliability metrics for *k*-out-of-*n* pairs:G Balanced systems by considering systems as failed when unbalanced system states occur. We further investigate such systems by balancing unbalanced states: When unbalanced states occur, the system is balanced by forcing down one or more operating pairs into standby. The reliability estimation is computationally expensive for such systems with a large number of units. Therefore, we develop an efficient approach for reliability approximation with high accuracy based on Monte Carlo simulation.

Also, we investigate the system reliability further by assuming that the units are subject to degradation. In many situations, units exhibit degradation that can be monitored. We model

the degradation path of any unit based on collected observations of the degradation indicator and its physics-based or statistics-based degradation rate. We consider the effect of units' operating conditions on their degradation paths.

Moreover, available system capacity is an important indicator of a system's condition. A system fails when its capacity drops below a minimum value. We estimate the reliability metrics of weighted-*c*-out-of-*n* pairs:G Balanced systems, which considers the capacities of individual units. We investigate the problem in two scenarios: First, we assume that the capacity of any unit has multiple levels. Second, we assume that the capacity of any unit has two levels (either working or failed) whereas different units may have different capacities. In the second scenario, we consider load-sharing effect.

Furthermore, optimal design for systems with spatially distributed units is the key to maximizing the reliability of the systems given the constraints such as the upper bound for the total number of units and load-sharing effect. We study the optimal configuration that maximizes the system reliability metrics.

ACKNOWLEDGEMENT

First of all, I would like to thank my advisor, Dr. Elsayed for his great support during my Ph.D. study at Rutgers University. His excellence and hard work always encouraged me to try harder, which helped me significantly in finishing this dissertation. I also want to thank the committee members of my dissertation, Dr. Albin, Dr. Cabrera, and Dr. Pham, for their comments and advice. Their questions on my proposal provided me with a lot of insights to further my research. Furthermore, I want to express my gratitude to the faculty and staff of the Department of Industrial and Systems Engineering for their assistance during my study. Finally, many thanks to other graduate students in the department. You all make my life on campus exciting and enjoyable.

This dissertation would not have been possible without the support of NPRP 04-631-2-233 grant from Qatar National Research Fund (QNRF).

DEDICATION

To my beloved family and friends.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Motivation of the Research

In practice, many systems have units arranged in a certain spatial configuration to perform their functions which we call systems with spatially distributed units. For example, a flash drive is composed of a number of memory cells in a cubic arrangement with connections in between; a three-dimensional CT image consists of thousands of pixels which form patterns that indicate the health condition of an individual; a supervision system has multiple cameras distributed spatially to monitor an area of concern [1]; an LED display has a number of LEDs arranged in arrays to display letters or digits; and an alternator has multiple field coils arranged evenly on a circle to provide symmetric magnetic flux. A system with spatially distributed units requires that units at certain locations must operate for the system to function properly. The spatial locations of units hence play an important role in the system's reliability estimation.

Such systems are most often redundant systems in the sense that not all units in the systems are required to operate for the systems' operation. For example, a disease will be diagnosed only when the pixels at particular coordinates of an CT image form some pattern; a supervision system will only fail when the cameras in certain spatial configurations fail so that the remaining operating cameras do not cover the monitored area; and an LED display

can still display recognizable letters with a few scattered failed LEDs though it cannot display letters correctly if more than a certain number of LEDs fail in a cluster. Generally, such systems can be modeled by multi-dimensional $\mathbf{k}$-out-of-$\mathbf{n}$:G/F systems.

The reliability estimation of multi-dimensional $\mathbf{k}$-out-of-$\mathbf{n}$ pairs:G/F systems has been studied by many researchers. A multi-dimensional $\mathbf{k}$-out-of-$\mathbf{n}$:G/F system is composed of units configured according to the vector $\mathbf{n}$, e.g. a cubic system composed of n_i units in the i^{th} dimension where $i = 1$, 2 and 3. The system operates/fails if and only if a group of units configured as the vector $\mathbf{k}$ operates/fails. For example, a (2, 2, 2)-out-of-(2, 2, 10):F system fails if and only if there is at least 8 units fail in a cube of size 2 units by 2 units by 2 units.

The reliability estimation of such systems is challenging due to the spatial configurations of units. For instance, compared with a k-out-of-n:F system, which fails when at least k units fail out of n units in total, it is more difficult to estimate the reliability of a consecutive-k-out-of-n:F system, which fails when at least k consecutively arranged units fail out of n units in total [2], due to the consideration of relative locations of units in the failure events. In other words, the cut set for reliability estimation cannot be determined without considering the spatial relationship of units. The problem becomes more complicated when the n units are in a circular arrangement [3].

A category of two-dimensional $\mathbf{k}$-out-of-$\mathbf{n}$:G system with units distributed evenly on a circle is fast emerging in aerospace and military industries. We name the systems as k-

out-of-n Pairs:G Balanced systems. One practical example of such systems is rotary Unmanned Aerial Vehicle (UAV) such as octocopters, which presents many potential uses in various areas such as military, commercial and scientific research. The failure of such systems may result in major consequences especially in areas with high population density. However, an extensive review of the literature reveals that research on reliability estimation of such systems is quite limited. Therefore, in this dissertation, we investigate the reliability estimation of a variety of k-out-of-n pairs:G Balanced systems.

1.2 Problem Definition and Assumptions

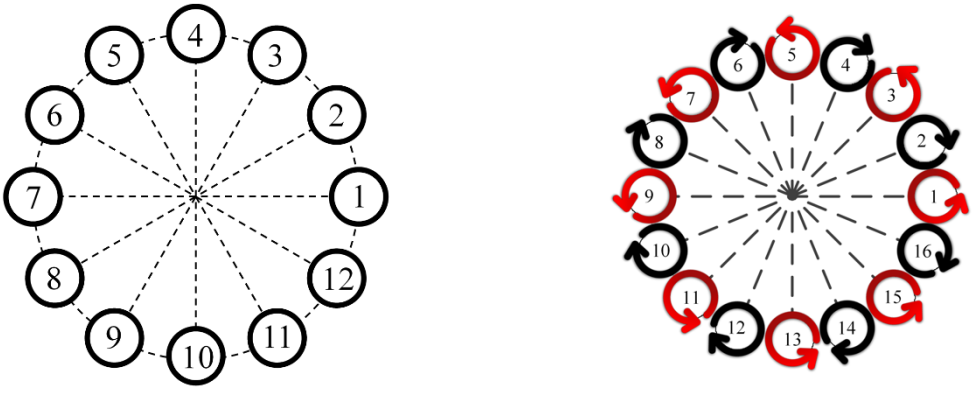
The reliability estimation of systems with spatially distributed units has been studied in the past two decades by many researchers due to its importance and wide applications. A special case of systems with spatially distributed units is the k-out-of-n pairs:G Balanced systems.

1.2.1 *System Description*

1.2.1.1 *k-out-of-n Pairs:G Balanced System*

A k-out-of-n pairs:G Balanced system has n pairs of units distributed evenly on a circle, as shown in Figure 1.1. Each pair of units is located along the same diameter of the circle. At least k out of n pairs should operate for the system to provide its desired function. Moreover, the system must maintain balance at all times. In the systems considered in this dissertation, all the remaining operating units in the system should be symmetric w.r.t. at least one pair of perpendicular axes of symmetry. This requirement also implies that when

a unit fails, the other unit of the same pair is forced down immediately. Other balance requirements can be imposed on the system as well. For instance, in some systems, the units may be required to be rotationally symmetric w.r.t. a specified angle. The individual units in the system can perform the same function, as in the planetary descending engine systems [4], or different functions, as in UAV where any two adjacent rotors rotate in opposite directions to provide the necessary lift for the UAV. Figure 1.1b shows a possible system configuration for an UAV with eight pairs of rotors where the arrows show the rotational directions, i.e. either clockwise or anticlockwise, of each rotor.



(a) (b)

Figure 1.1 Two examples of k-out-of-n pairs:G Balanced systems with n = 6 and 8

1.2.1.2 *Weighted-c-out-of-n Pairs:G Balanced System*

Weighted-c-out-of-n pairs:G Balanced system is a variant of k-out-of-n pairs:G Balanced system. In a weighted-c-out-of-n pairs:G Balanced system, we have n pairs of units distributed evenly on a circle as in k-out-of-n pairs:G Balanced systems. Any unit

has some capacity. The system requires at least a minimum capacity c to function while maintaining balance.

The capacities of individual units can be the same or different. In addition, the capacity of a unit has either multiple levels, e.g. full capacity, half capacity and zero capacity (failure), or two levels, i.e. full capacity and zero capacity. In this dissertation, we will investigate two scenarios of the reliability estimation of weighted-c-out-of-n pairs:G Balanced systems in Chapters 6 and 7, respectively.

1.2.2 *Definitions and Notations*

- ***Pair***. A *pair* is composed of two units that are located on the same diameter of the circular system arrangement.
- ***Unit identity number and pair identity number***. We identify units and pairs by using numbering system shown in Figure 1.1 throughout this dissertation unless stated otherwise. As shown, we start numbering the units from the unit on the extreme right hand side when we position one of the pairs horizontally. The numbering increases anticlockwise. In addition, each pair is identified by the smaller number in this pair but with an asterisk superscript, e.g. units 1 and 7 constitute pair 1^*. In this dissertation, we use letters such as i and j to index individual units and letters with an asterisk as superscript such as i^* and j^* to index individual pairs. A pair identity number also has an asterisk as superscript.

In addition, it is immediate that pair i^* is composed of units i and $(i+n)$. For example, units 1 and 7 compose pair 1^* as shown in Figure 1.1(a).

- ***State of a Unit.*** An individual unit has three possible states: operating, failed and forced-down. A unit is operating if it is performing its function. A unit is failed if either it fails or it is forced down permanently due to the failure of the other unit in the same pair. A unit is forced-down when it is forced down together with the other unit in the same pair for system balance while they are operating. In this dissertation, we consider two scenarios: (i) unbalanced systems are considered as failed, and (ii) unbalanced systems are rebalanced. Forced-down units do not resume operation in the first scenario, whereas they are in standby and can resume operation when necessary in the second scenario. The state of a unit is denoted as 1 if it is operating, 0 if it is failed, and –1 if it is forced-down.
- ***State of a Pair.*** Similarly, a pair of units has three possible states: operating, failed, and forced-down. A pair of units is operating when both units are operating properly. A pair is considered failed when one unit of the pair fails. A pair is forced-down when the pair is properly operating but is forced down to balance the system. Again, consider the two scenarios mentioned above. Forced-down pairs do not resume operation in the first scenario. However, in the second scenario, operating pairs are forced down into standby and can resume operation afterwards when necessary. Forced-down pairs in the second scenario are equivalent to standby pairs. The state of a pair is denoted as 1 if it is operating, 0 if it is considered as failed, and –1 if it is forced-down.

- ***State of a System.*** The state of a system is the combination of the states of the units in the system. The state of a system is denoted as a row vector of the states of individual units in ascending order of units' identity numbers.
- ***Weight of a Unit***. The weight of a unit is used for calculating the Moment Difference (MD) as introduced in Chapter 3. Specifically, a unit has weight 1 if it is operating and has weight 0 if it is failed (either it fails or it is forced down permanently due to the failure of the other unit in the same pair). A forced-down unit has weight 0 in the first scenario where a forced-down unit does not resume operation, but has weight 1 in the second scenario where a forced-down unit is in standby and can resume operation.

1.2.3 *Assumptions*

Throughout this dissertation, we assume the following unless stated otherwise:

- The units in a system, regardless of their functions, have *i.i.d.* lifetimes.
- The probability of two or more simultaneous failures is negligible.
- For a pair of units, whenever one unit fails, the other one of the same pair is forced down immediately and permanently; and the two units in the pair are always forced down simultaneously when they are operating but forced down for system balance.
- The cumulative failure rate of a standby pair does not change during the force down period. In other words, its cumulative failure rate immediately after resumption is the same as when it is forced down.
- A standby pair does not fail during the forced down period.

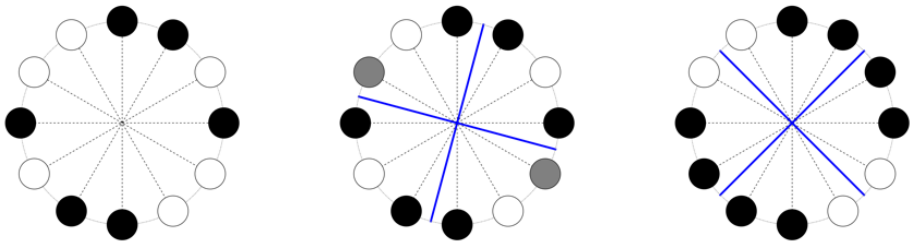
1.3 Reliability Estimation of *k*-out-of-*n* Pairs:G Balanced Systems

We first estimate reliability metrics of several types of k-out-of-n pairs:G Balanced systems with different configurations and balance requirements. We assume that units of the same type have *i.i.d.* lifetimes. Two scenarios are considered. In the first scenario, we consider a system as failed when the system reaches an unbalanced state. In the second scenario, as a system reaches an unbalanced state, it is balanced by forcing down operating pairs into standby. A standby pair can resume operation afterwards when needed for either balancing the system and/or providing an additional operating pair. Reliability estimation for k-out-of-n pairs:G Balanced systems presents three major challenges.

First, when n increases, the system's balance after the failure of a unit is not readily obvious. By forcing down the opposite unit of the failed unit is not necessarily sufficient to regain system's balance. In this case, some operating pairs must be forced down to balance the system. As one pair fails, we examine the states of the other pairs and determine which operating pairs to force down in order to maintain system's balance. It is important to keep the number of forced-down pairs as small as possible.

Second, locations and sequences of failures should be considered to obtain the set of successful events for reliability estimation. The failure of a system is determined by not only the number of failed units, but also their locations and sequences. For instance, consider the system shown in Figure 1.1(a) and assume $k = 3$. When pairs 1^*, 2^* and 3^* fail in any order, the system is balanced since the remaining operating pairs are symmetric w.r.t. a pair of perpendicular axes, i.e. the axes along pair 2^* and pair 5^*. When pairs 1^*,

3^* and 4^* fail in any order, the system is unbalanced since the remaining operating pairs are not symmetric w.r.t. any axes, as shown in Figure 1.2(a) where white pairs are considered operating and black pairs are considered failed. The system should be balanced by forcing down operating pair 6^* into standby, as in Figure 1.2(b) where gray pairs are considered in standby, which results in less than 3 operating pairs, hence a failed system. In some cases, the order of failures matters. For instance, consider the system shown in Figure 1.1(a) and assume $k = 2$. When pairs 1^*, 2^*, 3^* and 4^* fail sequentially, no standby is needed to balance the system since the system is always balanced with consecutively arranged failed pairs. When pairs 1^*, 3^* and 4^* fail sequentially, an unbalanced system is resulted, as shown in Figure 1.2(a), and standby is needed to bring the system back to balance, as shown in Figure 1.2(b). When pair 2^* fails afterwards, pair 6^* resumes operation since its resumption can bring an additional operating pair, as shown in Figure 1.2(c). The two events are successful since they both result in two operating pairs in the end, though they are two different events.



(a) (b) (c)

Figure 1.2 Example of standby pair in an unbalanced system and the resumption of standby pair

Third, it is difficult to estimate the probability of some successful events. As mentioned above, a successful event involves a sequence of failures and the order of failures matters. The estimation of the probability of an event with h failures requires an h^{th} order integral. If some operating pairs are forced down as standby to balance the system, it is quite difficult to obtain a closed form expression for such integral.

1.4 Reliability Approximation of *k*-out-of-*n* Pairs:G Balanced Systems

The reliability estimation of k-out-of-n pairs:G Balanced system is challenging in both successful event enumeration and event probability calculation as introduced in the previous section. The reliability estimation becomes extremely difficult when the system has a large number of units. It is very time-consuming, if indeed possible, to enumerate all system states and determine the complete set of successful events by enumeration. For a k-out-of-n pairs:G Balanced system, the number of unique successful events is approximately $\sum_{i=k}^{n} \frac{n!}{i!}$ by considering that we have $\left(n-\left(h-1\right)\right)$ options for the h^{th} failure to occur. When $n=30$ and $k=15$, the number of unique successful events is 2.1631×10^{20}. In addition, the determination of the probability of successful events, which involves multi-dimensional integration, requires expensive computational time.

Therefore, in this dissertation, we develop computationally efficient reliability approximation for k-out-of-n pairs:G Balanced systems. First, we investigate the reliability approximation of k-out-of-n pairs:G Balanced systems under the assumptions that unbalanced systems are considered as failed systems and that individual units have

i.i.d. lifetimes. Second, we investigate the reliability approximation of such systems under the assumptions that unbalanced systems are balanced by considering standby and that individual units have exponential *i.i.d.* lifetimes.

Monte Carlo simulation is effective in estimating the reliability of complex systems [5], [6]. In this dissertation, we use Monte Carlo simulation to approximate the reliability of k-out-of-n pairs:G Balanced systems. Utilizing simulation we can reduce the enumerations of successful events significantly by sampling a subset of events randomly as elaborated later in Chapter 4. In addition, we develop approximation for the multi-dimensional integral involved in the probability calculation of successful events. Numerical examples show that the reliability approximation approach proposed in this dissertation is effective and efficient.

1.5 Degradation Analysis of Systems with Spatially Distributed Units

In many situations, sensors monitor the degradation processes of critical units. The degradation measurements of individual units, which may be significantly affected by their operating conditions, can be used to enhance the accuracy of system reliability estimation. It is hence of great significance to develop a degradation model for spatially distributed units which considers the physics-based or statistics-based underlying degradation rate, captures the effect of operating conditions on the degradation rate, and the variabilities in the manufacturing of individual units. These factors bring more challenges in reliability estimation of systems with spatially distributed units.

In this dissertation, we model the degradation of individual units in k-out-of-n pairs:G Balanced systems. A unit fails when its degradation reaches a critical threshold. We assume that individual units of the same type have an identical baseline degradation rate, which is modeled as a function of time, e.g. power law. The effect of operating conditions on the degradation processes of individual units is modeled by stress-acceleration functions which govern the relationship between operating conditions (stresses) and acceleration factor for baseline degradation rate.

Assume that unbalanced systems are balanced by forcing down operating pairs into standby or resuming standby pairs back to operation, and assume that the units are not subject to degradation or failure during standby, this problem introduces two challenges:

First, successful events cannot be aggregated into groups to simplify computation. Under the assumption of *i.i.d.* lifetimes of individual units, successful events that have the same relative locations and sequences of failures occur with the same probability and hence can be aggregated into the same group to simplify computation. For instance, consider the system in Figure 1.1(a), the event that pairs 1^*, 2^* and 3^* fail sequentially and the event that pairs 2^*, 3^* and 4^* fail sequentially can be aggregated into one group. When individual units have different degradation paths due to different operating conditions, the lifetimes of individual units are no longer identically distributed. The probabilities of successful events should be obtained separately even if the relative locations and sequences of the failures in these events are the same.

Second, randomness in the degradation rates of units due to the operating conditions and manufacturing variations among the units.

1.6 Reliability Estimation of Weighted-*c*-out-of-*n* Pairs:G Balanced Systems

In many cases, the function of systems depends on the capacities of individual units, e.g., engines with certain horsepower, or generators with a certain output voltage. In previously mentioned topics, the capacities of units are considered as equal and ignored in reliability modeling. For example, in an UAV that consists of identical rotors, the rotors should be able to provide the same lift power. But the capacities of units can decrease and vary from unit to unit in some cases, and thus should be considered in reliability modeling.

In this dissertation, we investigate the reliability estimation of weighted-c-out-of-n pairs:G Balanced systems in two scenarios presented in Chapters 6 and 7 respectively.

1.7 Load-Sharing Effect on System Reliability

We investigate the effect of load-sharing on system reliability metrics by assuming that the load each unit shares, which depends on the number of remaining operating pairs in the system, affects its hazard rate. In a weighted-c-out-of-n pairs:G Balanced system with load-sharing effect, the load carried by failed units is re-distributed to the remaining operating units. When the operating units share more load, their hazard rates are affected in an adverse way, which decreases system reliability. In addition, the way load is

distributed to operating units should be considered in system reliability estimation especially when units have different capacities.

1.8 Optimal Design for Systems with Spatially Distributed Units

The *k*-out-of-*n* pairs:G Balanced systems exist in many applications such as the descent system of planetary vehicles [4] and UAV. The reliability of such systems has a major impact on the accomplishment of important missions, the cost that may occur when a failure happens, and public safety as in the case of UAV failure. The reliability optimization for such systems hence is significant in practice. Chapter 3 shows that there exists an optimal reliability design for *k*-out-of-*n* pairs:G Balanced system. Therefore, we investigate the optimal reliability design of a *k*-out-of-*n* pairs:G Balanced system in this dissertation.

1.9 Organization of the Dissertation

This dissertation is organized as follows. In Chapter 2 we present a comprehensive literature review of the related research. In Chapter 3 we present the procedures of estimating system reliability metrics for *k* -out-of- *n* pairs:G Balanced systems in two scenarios: (i) unbalanced systems considered as a failed systems and (ii) unbalanced systems are rebalanced by considering standby pairs. In this chapter, we also propose an approach for determining balance (symmetry) of a system and heuristics to determine standby pairs for unbalanced systems. In Chapter 4, we present a reliability approximation approach for the systems introduced in Chapter 3. In Chapter 5 we propose a degradation

model for spatially distributed units, based on which reliability metrics of k-out-of-n pairs:G Balanced systems are obtained. In Chapter 6 we investigate the reliability estimation of weighted-c-out-of-n pairs:G Balanced systems by assuming that the individual units are subject to multi-state capacity degradation. In Chapter 7, we investigate the load-sharing effect on the reliability of weighted-c-out-of-n pairs:G Balanced systems. In Chapter 8 we study the optimal design for k-out-of-n pairs:G Balanced systems. In Chapter 9 we present the conclusions and future research.

CHAPTER 2

LITERATURE REVIEW

2.1 Literature Review on Systems with Spatially Distributed Units

2.1.1 *Multi-Dimensional Consecutive-k-out-of-n:F Systems*

A type of well-known systems with spatially distributed units is the multi-dimensional consecutive-k-out-of-n:F systems. The applications of such systems range from electronic devices composed of cell units in squares or cubes [7], [8], TV supervision systems [9] and disease diagnosis based on X-ray [8].

A survey of the multi-dimensional systems derived from the one-dimensional consecutive-k-out-of-n:F system is found in [10]. Salvia and Lasher propose a two-dimensional consecutive-k-out-of-n:F system [7] by considering a square grid of units by side n, the system fails if there exists a failed square grid of units by side k. This is the first known multi-dimensional consecutive-k-out-of-n:F system considered in the literature. Koutras *et al.* [11], [12] provide estimates of the reliability of this system. Boehme *et al.* [9] propose a more generalized model, i.e. connected-X-out-of-(m,n):F lattice system, where the lattice of units can be rectangular, circular, and cylindrical. Boehme *et al.* also provide the reliability estimation of such a system. A special case of the connected-X-out-of-(m,n):F lattice system is the consecutive-(r,s)-out-of-(m,n):F system. Its reliability estimation is investigated further by Yamamoto and Miyakawa [13], Makri and Psillakis [14], Godbole

et al. [15], Hsieh and Chen [16], and Zhao *et al.* [17]. Godbole *et al*. [15] extend the result for d-dimensional system where $d \geq 3$. Other systems such as k-within-consecutive-(r,s)-out-of-(m,n):F systems [14], [18], [19] and consecutive-(r,s)-out-of-(m,n):F systems with constraints on the total number of operating units [20] are investigated.

Boushaba and Ghora [21] introduce the three-dimensional consecutive-k-out-of-n:F system and investigate its upper and lower reliability bounds. They state that "It is very difficult, probably impossible, to derive simple explicit formula for the reliability of a general three-dimensional consecutive-k-out-of-n:F system." Boushaba and Azouz [22] propose another method for estimating the lower reliability bound. Others [8], [23] attempt to address this research area but the contributions are limited due to the difficulty of the problem.

A thorough review of related work reveals that the reliability estimation of multi-dimensional consecutive-k-out-of-n:F systems is important yet challenging. The research is important since its applications can be found in many areas such as disease diagnosis by reading an X-ray [8] and other medical imagery [22], the failure model of three-dimensional flash memory cells [23], the failure model of thin film transistor liquid crystal display [19], scatter water area of a water sprinkler system [24], supervision system [9] and pattern recognition [7]. However, due to the difficulty of the problem, a large portion of the papers [7]-[12], [14]-[16], [21], [22], [25] only address the upper and lower bounds of system reliability without providing the exact values. In addition, very few papers [23] consider units with reliability functions.

2.1.2 *k-out-of-n Pairs:G Balanced Systems*

Another emerging system with spatially distributed units is the rotary winged Unmanned Air Vehicles (UAV) with multiple pairs of rotors [26]. UAV is set to play a major role in the future of the aerospace industry [27] and its use in many applications. For example, an UAV can collect more detailed geographic data than satellite [28]. Although UAVs have numerous potential applications, its flight is highly restricted now because its reliability is lower than manned aircraft [29]. No research is found related to the quantitative modeling and estimation of the multiple rotary UAV's reliability.

Multiple rotary UAV falls into the category of k-out-of-n pairs:G Balanced systems with units distributed spatially in a circular configuration. The reliability of such a system is difficult to estimate as it has the same nature of aforementioned multi-dimensional consecutive-k-out-of-n:F system. In addition, the consideration for system balance in k-out-of-n pairs:G Balanced systems adds to the difficulty in reliability estimation.

Attempts have been made for estimating the reliability of k-out-of-n pairs:G Balanced systems. Sarper and Sauer [4], [30] consider two balanced engine systems in planetary descent vehicles and estimate their reliability. The balanced engine system has four (or six) engines located evenly on a circular configuration to keep the descent vehicle in balance. Two (or three) engine pairs are formed along diameters of the circle. In each pair, when one engine fails, the second engine of the same pair is forced down to maintain balance. The system operates if and only if at least one (or two) engine pairs operate properly.

However, the two balanced engine systems with only two or three pairs of units are quite simple. The methods developed for estimating the reliability of these two systems hence cannot be used for more general systems.

2.2 Applications of *k*-out-of-*n* Pairs:G Balanced Systems

In practice, k-out-of-n pairs:G Balanced systems are already used in many applications. We provide some examples as follows:

2.2.1 *Engine Systems in Planetary Descent Vehicles*

Sarper and Sauer [4] present two balanced engine systems in planetary descent vehicles and estimate their reliability. The balanced engine system has four (or six) engines located evenly on a circle to keep the descent vehicle in balance. Two (or three) engine pairs are formed along diameters of the circle. When one engine fails, the second engine of the same pair is forced down to regain balance. The system operates if and only if at least one (or two) engine pairs operate properly depending on the requirements of the system.

2.2.2 *Unmanned Aerial Vehicles*

Unmanned Aerial Vehicles (UAVs), a.k.a. drones, are widely used in military and commercial applications. UAVs with multiple rotors, e.g. quadcopter, hexacopter, and octocopter, are used in border protection and are being prototyped for package delivery and other applications. UAVs with multiple rotors can be modeled as k-out-of-n pairs:G Balanced systems. The individual rotors rotate in two opposite directions to provide thrust

for UAVs, and at the same time, provide UAVs the ability to pitch, roll or yaw. Not all the rotors have to be rotating for an UAV to fly safely when more than 6 rotors are mounted in the system, though the number of working rotors should be above a critical number. An UAV also requires balance in the sense that the operating rotors should be symmetric, and the number of rotors rotating in the opposite directions should be equal.

2.2.3 *Generators and/or Alternators*

The stators and rotors of wind turbine generators are composed of multiple sets of three-phase windings. In each set of windings, the windings of different phases should be mounted evenly on a circle with $2/3\pi$ degrees between them. The multiple sets of windings are then mounted evenly with the other windings on the circle in a symmetric manner. The symmetry of windings is critical for the "health" condition of the generators [31], [32]. The symmetry among windings is necessary to avoid failure of the generator. The winding sets can be modeled as a k-out-of-n pairs:G Balanced system by considering each winding as a pair because each winding occupies the two ends of a diameter of a circle.

For an alternator, its salient pole rotor has multiple field coils arranged evenly on a circle to provide magnetic flux. The coils should be arranged in such a way that any two adjacent coils should provide opposite magnetic poles. The sets of coils can also be modeled as a k-out-of-n pairs:G Balanced system by considering each coil as a unit.

2.3 Literature Review on Symmetry Measure

The existing research on axes of symmetry, which exists in various areas such as chemistry and phytology, can be categorized into two major areas: seeking axes of symmetry for symmetric or approximately symmetric shapes or images [33]; and measuring asymmetry or symmetry of a shape or image and determining the minimum change needed to get the shape or image into a symmetric one [34]. The balance of the proposed k-out-of-n pairs:G Balanced system highly depends on its symmetry. The measure of symmetry and rebalancing an unbalanced system is a variant of the second research area. However, the minimum change involved in the second area is not to omit or add elements such as points or pixels, but to adjust the locations of currently existing elements. So the methods cannot be applied to the problem under study which involves forcing down operating pairs into standby and resuming standby pairs back to operation (omitting and adding elements temporarily from the system).

In this dissertation, we develop a measure of symmetry to determine the balance of k-out-of-n pairs:G Balanced systems. In the scenario where standby is considered, we develop heuristics to determine which operating pairs to force down into standby and which standby pairs to resume operation to rebalance unbalanced systems.

2.4 Literature Review on Monte Carlo Simulation-Based Reliability Approximation

In this section, we provide a literature review on Monte Carlo simulation-based reliability approximation. Monte Carlo simulation-based methods are widely used in reliability

approximation. Crude Monte Carlo simulation performs well in simple problems, though becomes time-consuming when problems become more complex. Therefore, many methods are developed for variance reductions to improve the efficiency of the Monte Carlo simulation, e.g. subset simulation and importance sampling.

A thorough review of the literature shows that the existing methods either lack the ability to approximate the reliability of k-out-of-n pairs:G Balanced systems or require significant effort to be applied to the systems. In this dissertation, we develop an efficient and effective reliability approximation method for k-out-of-n pairs:G Balanced systems based on Monte Carlo simulation. In addition, this method can be easily generalized to approximate the reliability of other systems with spatially distributed units.

2.4.1 *Crude Monte Carlo Simulation*

The Crude Monte Carlo simulation method repeatedly generates the realizations of system states by randomly drawing samples from the distributions of components' failure times. A system state is considered successful if the remaining operating components, the failure times of which are greater than the mission time, form a path. Approximate value of system reliability is obtained as the ratio of the number of successful system states over the total number of states generated through a large number of simulation runs. The approximate value converges to the true value as more simulation runs are implemented. Research in this area is found in [35], [36], [37], [38], and [39].

This method, though straightforward, tends to require a great amount of time to converge. Especially when the target system states occur with small probabilities, it requires a significant number of simulation runs to draw enough samples from the target region [40]. Thus many papers propose variance reduction methods to improve the simulation efficiency.

2.4.2 *Subset Simulation*

Subset simulation treats each system state as a result of a sequence of intermediate system states. The underlying idea is to express the probability of a target system state, which can be very small in many cases, as a product of probabilities conditional on some intermediate system states [40]. Thus a rare system state can be generated by a sequence of simulations of more frequent intermediate system states. This method can be used when it is possible to model system states as a vector of parameters so that target system state and its intermediate system states can be modeled as subsets of the universal set of the parameter vectors. Research in this area is found in [40], [41], [42], [43], and [44].

In this dissertation, we consider the k-out-of-n pairs:G Balanced systems. The function of such systems depends on the sequence and location of the failures; and it is very difficult, if not impossible, to model intermediate system states into subsets especially when there are system states with forcing-down and resumption of standby pairs.

2.4.3 *Importance Sampling*

Importance sampling is a widely used variance reduction method to improve simulation efficiency. The basic idea is to draw samples from an importance distribution that overweighs the target region instead of the original distribution, and then adjust the estimation with a likelihood ratio, namely the ratio of *pdf* of the original distribution over the *pdf* of the importance distribution. Due to the oversampling in the region of concern, the importance sampling method converges in significantly less simulation runs [45]. This method is especially used in the case of highly reliable systems and hence a small failure region. Research and survey in this area are found in [46], [47], [48], and [49].

The success of the method relies on a prudent choice of the importance distribution, which requires knowledge of the system in the failure region [40]. However, it is difficult, if not impossible, to determine the importance distribution for the failure region of k-out-of-n pairs:G Balanced systems.

2.4.4 *Other Variance Reduction Methods*

Other variance reduction methods are found in the literature. However, these methods require prior knowledge of the system, e.g. complete minimum cut set [50] or fault tree [51], lack the ability to model the effect of failure sequence [52], [53] or locations [54], [55], or are developed for specific systems, e.g. series and parallel systems [52], network systems [53], [56], linear sensor systems [57], or construction structures [58], [59], [60], [61]. It requires much extra effort to apply these methods to the reliability approximation of k-out-of-n pairs:G Balanced systems, if indeed possible.

2.5 Literature Review on Degradation Modeling of Individual Units

2.5.1 *Physics-Based Degradation Model*

Modeling the degradation process of products based on physics of failure is common for products with known failure mechanisms. Using physics-based degradation models results in a more accurate prediction for the reliability metrics of a batch of products.

A recent thesis by Kulkarni [62] investigates the physics degradation model for electrolytic capacitors. Tyaginov *et al.* [63] analyze and classify the existing hot-carrier degradation models and present a novel degradation model that includes all essential aspects of hot-carrier degradation. McPherson [64] presents three generally used physics-based degradation models: power law, exponential, and logarithmic. “These three forms were selected because they tend to occur rather frequently in nature.” [64]

Power law model is more frequently observed according to McPherson [64]. Hot carrier induced degradation of an SOI (Silicon on Insulator) MOSFET (Metal–Oxide–Semiconductor Field-Effect Transistor) is best described by power law βt^n where the exponent n is centered at around 0.25 or 0.5 depending on the stress conditions [65]. The threshold voltage of an PMOS (P-type Metal-Oxide-Semiconductor) transistors also has a degradation process in the form of βt^n [66], [67].

Exponential and logarithmic models are used when power law model does not fit the data. Drag-reduction-effect of polymer additive decreases following an exponential path [68]. Meeker and LuValle [69] describe an example of exponential degradation process on a circuit board. The remaining salts in printed circuit boards after manufacturing react with copper to generate copper chlorine compounds that may cause failure of circuit boards. The amount of copper chlorine compounds increases in an exponential form. Doyle [70] finds that the weight loss of a silicon resin in time follows a logarithmic model. He also states that the logarithmic process is encountered quite often not only in weight loss studies but also in some studies of moisture sorption and desorption, as well as loss of dielectric strength by solid polymers during heat-aging.

Physics-based degradation models utilize the information of the physics of the failure mechanisms of products and provide perspective on its degradation processes. However, the limitations of physics-based degradation models are also obvious. First, these models are much more difficult to construct. Extensive experiments must be carried out based on in-depth knowledge of physics, chemistry and metallurgical properties of the components. Second, the models are usually for simple electronic devices or mechanical components. The application of these models is limited for complex systems with multiple components. Third, physics-based degradation models tend to result in less accurate prediction for individual products due to the diversity of products and the stochastic nature of failures [71].

Statistics-based degradation models are used when physics-based degradation models are not adequate due to its limitations.

2.5.2 *Stochastic Process/Statistics-Based Model*

A general degradation path is proposed by Meeker and Escobar [72]. The degradation of unit i at time t_j is modeled as $y_{ij} = Y_{ij} + \varepsilon_{ij}$ where Y_{ij} is the actual path and ε_{ij}, which follows a normal distribution with zero mean, is the residual deviation. In addition, stochastic processes are commonly used to describe the degradation processes. "Stochastic models, as an alternative, are preferred whenever there is lack of experimental data or prior knowledge about the products." [73] Brownian motion and Gamma processes are two widely used stochastic processes for modeling the degradation paths.

Gebraeel *et al.* [74] model the degradation process of product using an exponential model where the randomness from unit to unit is modeled with a centered Brownian motion. According to Gebraeel *et al.* [74], [75], the randomness from unit to unit can also be modeled with a standard normal distribution. Wang *et al.* [76] propose a degradation-based remaining useful life prediction method where the degradation is modeled using a Brownian motion with drift. The drift parameter is adaptive to the history of condition monitoring information. Wu *et al.* [77] propose a degradation-based reliability estimation method with random failure threshold.

Marseguerra *et al.* [78] model the degradation of a maintainable system with a discrete time Markov chain that describes the condition of the system before or after maintenance.

Grall *et al.* [79] present condition based maintenance (CBM) approach where the condition (degradation) of single-unit-system is modeled with a Markovian stochastic process. Since the inspections are carried out at discrete instants of time when degradation increments occur, the model can also be considered as a cumulative damage model. The increments are considered to follow exponential distribution or Gamma distribution. Deloux *et al.* [80] study the maintenance optimization of a system subject to environmental stresses. The degradation of the system is modeled as a Gamma process. Liao *et al.* [81] use Gamma process to model the degradation of systems in numerical simulation due to the monotonous increments property of Gamma process.

2.5.3 *Degradation Modeling Considering Operating Conditions*

Many degradation models that consider operating conditions are found in the area of Accelerated Degradation Testing (ADT). Meeker *et al.* [82] provide an extensive introduction of ADT. Brownian motion or geometric Brownian motion with stress (operating condition) dependent drift coefficient and gamma process with stress-dependent shape parameter are widely used to model degradation processes considering operating conditions [83], [84], [85].

Liu *et al.* [86] present a degradation model for individual units considering the effect of operating conditions using a Brownian motion with time variant drift and diffusion coefficients. The model is composed of two parts: the deterministic part which deals with either physics-based or statistics-based degradation rate, and the stochastic part which considers the randomness in the degradation processes. The effect of operating conditions

on degradation rate is modeled by applying a stress-acceleration factor. Closed form expressions for *pdf* of lifetimes and reliability function of individual units are obtained.

2.5.4 *Stress-Acceleration Functions*

Stress-acceleration functions are widely used to model the effect of stresses (operating conditions) on degradation processes. Stress-acceleration functions can be categorized into experimental-based functions and empirical functions [87] as explained below.

2.5.4.1 *Experimental-Based Functions*

The Arrhenius equation is one of the most widely used stress-acceleration functions. It relates the rate of a simple (first order) chemical reaction rate (degradation rate) and temperature [88] as given in Eq. (2.1)

$$\mu = C \cdot \exp\left(-\frac{E_a}{K \cdot T} \right) \tag{2.1}$$

where μ is the reaction rate; C is a constant; E_a is the activation energy of the reaction, usually in electron-volts; T is the absolute Kelvin temperature; and K is Boltzmann constant.

Another widely used stress-acceleration function is the Eyring equation that describes the effect of temperature on the reaction rate of a process:

$$\mu = C \cdot f\left(T\right) \cdot \exp\left[-\frac{E_a}{K \cdot T} \right] \tag{2.2}$$

where $f(T)$ is a function of temperature determined by the specifics of the reaction dynamics. Applications in the literature typically assume $f(T) = T^m$ with a fixed value of m [87].

2.5.4.2 *Empirical Functions*

It is possible to obtain stress-acceleration functions via experimental observations or based on knowledge of physics or chemistry, yet it is time-consuming and sometimes infeasible to do so. Simple empirical functions have been used to describe the stress-degradation relationship.

In some Brownian motion based degradation models, the degradation rate (drift coefficient of the Brownian motion) μ is expressed as an exponential function of the stress vector $\mathbf{S}$:

$$\mu(\mathbf{S}) = \exp\left[a + b\varphi(\mathbf{S})\right] \tag{2.3}$$

where $\varphi(\mathbf{S})$ is a function of stress vector $\mathbf{S}$, and a and b are constants. Wang *et al.* [89] and Liao and Elsayed [83] use this model in ADT. Other empirical functions such as power law model are described in [73], [90], [91].

2.6 Literature Review on the Degradation Analysis of Systems with Multiple Units

2.6.1 *Systems with a Simple Configuration of Units*

Most of the research focuses on the degradation of simple systems such as series systems, parallel systems, the combinations of series and parallel systems, and k-out-of-n:G/F

systems where at least k out of n units must operate or fail. In such systems, only certain configurations of the units are required for the systems to operate. In other words, the spatial relationships of individual units have no effect on reliability analysis.

Pham *et al.* [92] investigate the reliability metrics of a k-out-of-n:G system with units that are subject to both discrete degradation stages and catastrophic failure. An individual unit either fails as it reaches the last stage of degradation or fails catastrophically. Song *et al.* [93] propose a degradation model for individual units considering the impact of shocks, which is shared by all units, on their degradation. The estimation of system reliability for series systems, parallel systems, and series-parallel systems is given. Gupta and Lawsirirat [94] present a general model for systems with multiple degrading units using Failure Modes and Effects Analysis (FMEA) to obtain interaction intensities ρ_{ij} between individual units i and j, the system degradation at time t, $D_{sys}(t)$ is obtained as

$$D_{sys}(t) = \sqrt{\sum_{i=1}^{N}\sum_{j=1}^{N} \rho_{ij} D_{i,t} D_{j,t} + J_{sys,t}} \tag{2.4}$$

where $D_{i,t}$ is the degradation of unit i ($i = 1, 2, \ldots, N$) at time t; and $J_{sys,t}$ is the damage caused to the system due to the failure of individual units. This model depends on the accuracy of ρ_{ij} obtained through FMEA, which is difficult to obtain for complex systems. Bian and Gebraeel [95] propose a degradation model for multiple units considering the dependence between the degradation rates of the units, which results in a more accurate prediction of the remaining useful life of individual units.

2.6.2 *Systems with Spatially Distributed Units*

Degradation analysis for systems with spatially distributed units is much more complicated due to the consideration of locations and sequences of failures. Work related to this area is sparse. Enright and Frangopol [96] investigate the reliability of a highway bridge with multiple spatially distributed girders. Series systems and series-parallel systems are used to model the bridge. The degradation of resistance of the girders is modeled with a deterministic polynomial function of time with a time variant random threshold as explained later. The overall load on the bridge, which follows a known distribution, occurs according to a Poisson process. This problem involves load redistribution whenever some girders fail, which is complicated due to the spatial distribution of the girders. The load shared by remaining girders, which can be considered as the critical thresholds for their resistance, depends on their spatial locations and the locations of failed girders. A failure tree describing the events with girders in different locations failing in different sequences is built to facilitate reliability estimation.

Marsh and Frangopol [97] investigate the reliability of reinforced concrete bridge decks by considering the corrosion of the reinforcing steels. Spatial correlation between corrosion of the reinforcing steels at different locations of the deck is considered in the model to enhance the accuracy of reliability estimation. The locations of reinforcing steels have a significant effect on the overall resistance capacity of the bridge deck to the applied load. No explicit expression for system reliability is obtained due to the modeling and computation challenges of the problem. Instead, Monte Carlo simulation is used to obtain system reliability.

2.7 Literature Review on Multi-State *k*-out-of-*n*:G/F Systems

El-Neweihi *et al.* [98] propose the first definition for a multi-state k-out-of-n system model, where the system state is defined as the state of the k^{th} best unit [99]. Boedigheimer and Kapur [100] define the multi-state k-out-of-n system in terms of lower and upper boundary points, which is considered as consistent [99] with the one proposed by El-Neweihi *et al.* [98]. Huang *et al.* [101] propose a generalized multi-state k-out-of-n:G system where the state space of each unit and the system is $\{1,2,\cdots,M\}$ and a lower state level represents a worse or equal performance of the unit or the system. The definition of the system is as follows:

An n-component system is called a generalized multi-state k-out-of-n:G system if $\phi(\mathbf{x}) \geq j\ (j=1,2,\cdots,M)$ whenever there exists an integer value $l\ (j \leq l \leq M)$ such that at least k_l components are in states at least as good as l. Where $\mathbf{x}=(x_1,x_2,\cdots,x_n)$ is an n-dimensional vector representing the states of all components, $\phi(\mathbf{x})$ is the state of the system, which is also called the structure function of the system.

A corresponding multi-state k-out-of-n:F system is defined as follows [102]:

An n-component system is called a generalized multi-state k-out-of-n:F system if $\phi(\mathbf{x}) < j\ (1 \leq j \leq M)$ whenever the states of at least k_l components are below l for all l such that $j \leq l \leq M$.

An example of k-out-of-n:G system is given by Zuo *et al.* [102] as follows:

> "Consider a power station with three generators. Each generator is treated as a component, and there are three components in the system. Each generator may be in three possible states: 0, 1, and 2. When a generator is in state 2, it is capable of generating 10 megawatts of power output; in state 1, 2 megawatts, and in state 0, 0 megawatt. The total power output of the system is equal to the sum of the power output from all three generators. We can describe this model as follows: The system is in state 2 whenever at least 1 component is in state 2; in state 1 or above whenever either at least 1 component is in state 2 or at least 2 components are in state 1 or above; and in state 0 otherwise."

The reliability metric estimation of the systems is discussed by [101], [103], [104], [105], [106] and [107]. A comparison of the method is given by Mo *et al.* [108]. Overall, the objective is to increase the efficiency of reliability estimation algorithm and to reduce computation time, or to provide bounds [104] and approximation [107] for system reliability metrics. Tian *et al.* [99] propose a new multi-state k-out-of-n:G system with the following definition:

An n-component system is called a multi-state k-out-of-n:G system if $\phi(\mathbf{x}) \geq j \left(j = 1, 2, \cdots, M \right)$ whenever at least k_l components are in state l or above *for all* l such that $1 \leq l \leq j$.

Similarly, there is a corresponding definition for multi-state k-out-of-n:F system. The reliability of such systems are discussed in [99], [107], and [108].

Another variant of the multi-state k-out-of-n:G/F systems is multi-state consecutive-k-out-of-n systems [109], [110], [111]. Explicit recursive formulas are provided for special systems and algorithms are developed for the generalized systems.

2.8 Literature Review on Weighted-*c*-out-of-*n*:G/F Systems

In the research related to the reliability estimation of redundant systems, it is assumed that each individual unit in the systems has its own integer weight, which can be considered as its capacity. The systems are called systems with weighted units.

Wu and Chen first propose weighted-c-out-of-n:G system [112] and consecutive-weighted-c-out-of-n:F system [113]. The systems consist of n units, each of which has its own integer weight. The former system operates if and only if the total weight of operating units is at least c, whereas the latter system fails if and only if the total weight of the consecutively failed units is at least c. Recursive algorithms are developed to estimate the reliability of these systems. Chang *et al.* [114] investigate circular consecutive-weighted-c-out-of-n:F system and develop an algorithm for estimating its reliability.

Eryilmaz [115] investigates the reliability of a k-out-of-n:G system with units of random integer weights. The system operates if and only if there are at least k operating units, and the total weight of all operating units is above a critical value c. Kamalja and Amrutkar [116] and Eryilmaz [117] provide reviews of systems with weighted units.

2.9 Literature Review on Load-Sharing Models

2.9.1 *Load-Sharing Rules*

Most of the research that address the load-sharing effect on system reliability utilize the *monotone load-sharing rule* where the load increment due to failures of some units is shared among the remaining operating units [118]. Two special cases of the monotone load-sharing rule are the *equal load-sharing rule* and *local load-sharing rule* [119]. Under the equal load-sharing rule, all operating units share overall system load equally [120]. Under the local load-sharing rule, the load carried by failed units is distributed only to their adjacent units.

2.9.2 *Load-Sharing Effect Models*

The most widely used models for load-sharing effect are Accelerated Failure Time Model (AFTM) [121], Tampered Failure Rate (TFR) model [120], Cumulative Effect (CE) model and Proportional Hazards Model (PHM).

2.9.3 *Baseline Hazard Rate Models*

Constant Baseline Hazard Rate Model. Constant baseline hazard rate model simplifies load-sharing problems and hence is widely used in literature. Scheuer [122] investigates the reliability of k-out-of-n:G system under two assumptions: (i) individual units have *i.i.d.* exponential lifetimes, which implies identical constant hazard rate, and (ii) the identical hazard rate of operating units increases when a failure occurs. Shao and Lamberson [123] study the problem by assuming repairable units and imperfect switching,

i.e. failure to detect and disconnect failed units. Lin *et al.* [124] investigate the problem in [122] further by assuming non-identical units.

Time-Variant Baseline Hazard Rate Model. Time-variant baseline hazard rate model is more general and realistic but complicates load-sharing problems. Hassett *et al.* [125] provide analytical results for the reliability and availability of systems with one or two units that have non-identical power law time-variant hazard rate and repair rate. Amari *et al.* [120] provide analytical results for k-out-of-n:G systems considering TFR load-sharing effect and generalize the results to a wide range of time-variant baseline hazard rate models.

2.9.4 *Other Research*

Yamamoto *et al.* [126] investigate the optimal load allocation (load-sharing rule) for k-out-of-n:F systems, and find that system lifetime is maximized by allocating the load to units according to their residual lifetimes. Huang and Xu [127] estimate the reliability of a k-out-of-n:G system with maximally m ($k \le m \le n$) active units at all times. Load (tasks) is assigned to units by a controller, and active units can be either busy or idle. Qi *et al.* [128] investigate the optimal maintenance policy for load-sharing computer systems with k-out-of-n:G redundant configuration.

2.10 Literature Review on Optimal Design for Systems' Reliability with Spatially Distributed Units

The system reliability optimization without considering maintenance can generally be categorized into redundancy allocation problem, reliability allocation problem and the combination of the two [129]. Redundancy allocation problem is to add redundant units to the system and/or to allocate redundant units to sub-systems in the form of cold, warm or hot standby. The sub-systems can take a variety of forms such as parallel systems or *k*-out-of-*n* systems. On the other hand, reliability allocation problem treats the reliability of the units allocated to any sub-system as a decision variable. Coit *et al.* [130] study the redundancy allocation problem for systems with multiple *k*-out-of-*n* sub-systems in series. Elegbede *et al.* [131] and Tian and Zuo [132] investigate the redundancy-reliability allocation problem considering parallel sub-systems in series. Reviews on the reliability optimization research are given in Kuo [129], [133].

The redundancy allocation and reliability allocation problems are proven to be NP-hard [134], [135]. As the complexity of the problems grows, heuristics become a common technique for solving the optimization problems [133], [136], [137]. The most popular heuristics include Ant Colony, Genetic Algorithm, Tabu Search, and Simulated Annealing [133].

The reliability optimization for *k*-out-of-*n*:F/G systems has been a subject of investigation. Yosi [138] studies the reliability optimization of *k*-out-of-*n* systems when the units have two failure modes. Zuo [139] investigates the reliability allocation for consecutive-*k*-out-

of-*n*:F/G systems. The objective functions include exact values of system reliability metrics, upper or lower bounds of system reliability metrics, and overall cost considering system reliability and cost of failure. A comprehensive review of reliability optimization for *k*-out-of-*n*:F/G systems is given in [140].

However, reliability optimization for multi-dimensional **k**-out-of-**n**:F/G systems is sparsely studied [140]. Zuo [141] investigates the reliability optimization for 2-D consecutive k-out-of-n:F systems. This dissertation is the first to investigate the reliability optimization for k-out-of-n pairs:G Balanced systems. We investigate both redundancy allocation and reliability allocation problems for *k*-out-of-*n* pairs:G Balanced systems. First, we determine the optimal number of pairs and optimal standby policy, which is a redundancy allocation problem. Second, we allocate pairs with different lifetime distributions to different locations in the system to optimize the overall system reliability metrics, which is a reliability allocation problem.

CHAPTER 3

RELIABILITY ESTIMATION OF *k*-OUT-OF-*n* PAIRS:G BALANCED SYSTEMS

3.1 Problem Definition and Assumptions

In this chapter, we present methods for reliability estimation of different types of *k*-out-of-*n* pairs:G Balanced systems in two scenarios: (i) unbalanced systems are considered as failed systems and (ii) unbalanced systems are rebalanced. We develop a systematic approach for enumerating the complete set of successful events, which are ordered sequences of failures described by system state transition paths, and obtain closed form expressions for calculating the probabilities of successful events. The developed methods can be generalized to other systems with spatially distributed units.

The assumptions stated in the Chapter 1 hold throughout this chapter. We also assume that the hazard rates of individual units are not affected by the total number of operating units.

3.2 Reliability Estimation of Systems Considering Unbalanced States as Failure of the System

3.2.1 *System Description*

In this section, we consider two types of k-out-of-n pairs:G Balanced systems with the constraint that a system fails once it becomes unbalanced even if there are still k or more

operating pairs remaining in the system. In the first type of systems, all units perform the same function; whereas in the second type, the units perform complementary functions with any two adjacent units performing different functions as explained later.

For clarity and simplicity, when we mention state pattern, typical state, and follow-up state, we mean the system state rather than the unit state or the pair state.

3.2.2 *Symmetry Determination*

If a system is balanced, it is symmetric in the sense that all the operating units in the system should be symmetric w.r.t. at least one pair of perpendicular axes. We introduce the concept of Moment Difference (MD) to determine the degree of symmetry of a system w.r.t. any candidate axes. The MD of a system w.r.t. candidate axis a is calculated as

$$M_a = \sum_{i \in U_a} w_i \sin \theta_{i,a} \tag{3.1}$$

where M_a is the MD of the system w.r.t. candidate axis a; U_a is the complete set of all the units within $\pm\pi/2$ of candidate axis a; w_i is the weight of unit i; and $\theta_{i,a}$ is the angle from the axis on which unit i is located to candidate axis a. The weight of each unit is as defined in the Introduction. The angle $\theta_{i,a}$ has a positive value if it is clockwise and negative value otherwise.

For k-out-of-n pairs:G Balanced systems considered in this chapter, a candidate axis of symmetry is either along a pair of units, or in the middle of two adjacent pairs. In a system with n pairs of units, there are $2n$ candidate axes which compose n pairs of the

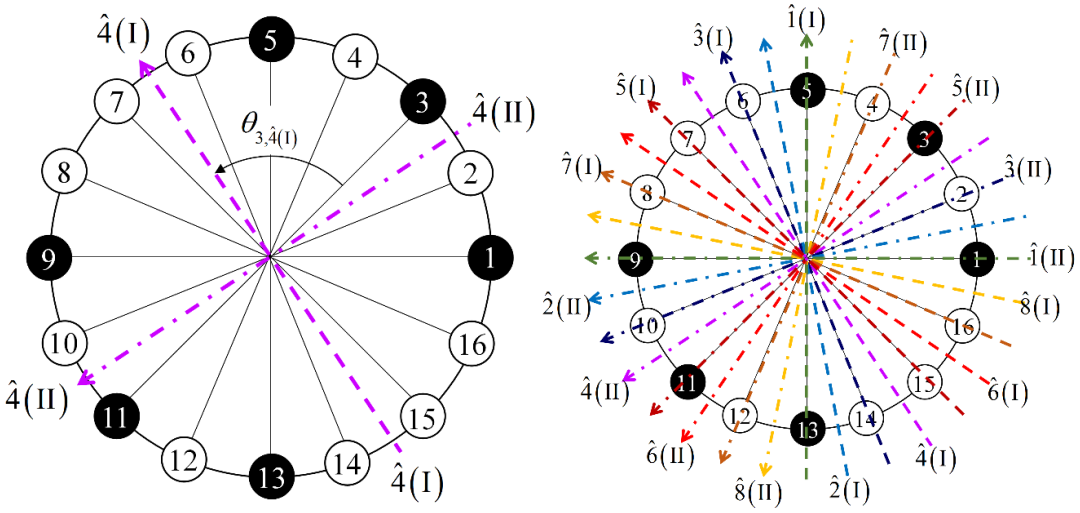
perpendicular candidate axes. We examine the symmetry of the system w.r.t. all n pairs of the perpendicular candidate axes by corresponding MD values.

Consider the system shown in Figure 3.1(a) which has $n=8$ pairs of units. The state of the system is (0101011101010111), i.e. the units in white have state 1 and the units in black have state 0 as mentioned in the Introduction. This system has $n=8$ pairs of perpendicular candidate axes. We index the candidate axes as shown in Figure 3.1(b). We simply index the 8 pairs of perpendicular candidate axes by numbers $\hat{1}$ to $\hat{8}$. In addition, to differentiate between the two perpendicular candidate axes in each axis pair, we index them with (I) and (II). For example, the vertical axis is indexed as $\hat{1}(\mathrm{I})$ and its perpendicular axis is indexed as $\hat{1}(\mathrm{II})$. Consider the candidate axis $\hat{4}(\mathrm{I})$ shown by the dashed line in Figure 3.1(a). The complete set of units within $\pm\pi/2$ of the axis is $U_{\hat{4}(\mathrm{I})}=\{3,4,\cdots,10\}$, the corresponding weights w_i for the units in $U_{\hat{4}(\mathrm{I})}$ are $[0,1,0,1,1,1,0,1]$, and corresponding angles $\theta_{i,\hat{4}(\mathrm{I})}$ are $\{\pi/8\cdot(-3.5,\ -2.5,\ -1.5,\ -0.5,\ 0.5,\ 1.5,\ 2.5,\ 3.5)\}$. For instance, in Figure 3.1(a) the angle from the axis on which unit 3 is located to the candidate axis $\hat{4}(\mathrm{I})$ is $\theta_{3,\hat{4}(\mathrm{I})}=-3.5\pi/8$. The MD value of the system w.r.t. this candidate axis is 0.7049. Similarly, the MD w.r.t. the perpendicular candidate axis, axis $\hat{4}(\mathrm{II})$ shown by dotted-dashed line, is –0.4710.

Starting from the vertical and horizontal candidate axes, i.e. axes $\hat{1}(\mathrm{I})$ and $\hat{1}(\mathrm{II})$ in Figure 3.1(b), we consider the $n=8$ pairs of perpendicular candidate axes anticlockwise. The MD

values are shown in Figure 3.2 where the Moment Difference I and II represent the MD values of the system w.r.t. the 8 candidate axes indexed with (I) and (II), respectively.

We then examine if there are any pairs of perpendicular candidate axes that have zero MD values w.r.t. both candidate axes. If so, the system is symmetric; otherwise, the system is asymmetric. For instance, in Figure 3.1(a) we can easily conclude that the system is symmetric since all the operating pairs are symmetric w.r.t. candidate axes $\hat{5}(\mathrm{I})$ and $\hat{5}(\mathrm{II})$. The MD values for the two candidate axes are both zero as shown in Figure 3.2. We can numerically validate that MD is effective in determining system symmetry when *n* is less than 30. When *n* is large, we add another condition for system symmetry as introduced in Section 4.2.1.



(a) (b)

Figure 3.1 Illustration of a system for Moment Difference calculation

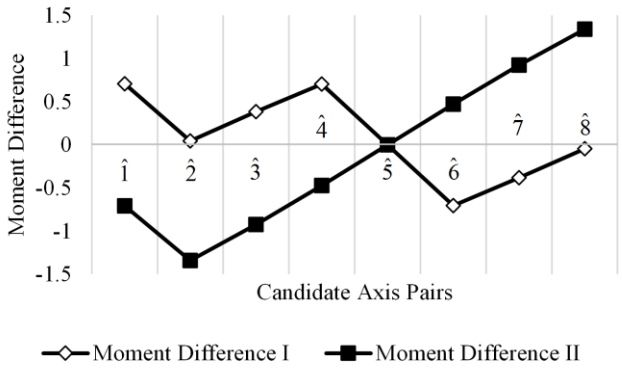


Figure 3.2 Illustration of Moment Difference for all candidate axes in a system

3.2.3 *Fundamental Method of Successful Event Enumeration*

In order to estimate the reliability of k-out-of-n pairs:G Balanced system, we first obtain all the successful events, i.e. system state transition paths that lead to operating system states, and estimate the probability of each event. The system state transition paths can be aggregated into several types, which reduces the computational time significantly. Moreover, the number of realizations of each type of system state transition paths is also determined.

First, we enumerate all the state patterns with a certain number, say h, of failed pairs. All the system states can be categorized into some state patterns. Consider the system in Figure 1.1(a), the two system states when pairs 1^*, 2^*, 3^* fail and when pairs 1^*, 5^*, 6^* fail can be categorized into the same state pattern with three consecutive failed pairs. We represent each state pattern with one typical state. A typical state of a state pattern is an arbitrary state of the system with failures arranged in the corresponding state pattern. Consider the system in Figure 1.1(a), the typical state of state pattern with three consecutive failed pairs

can be anyone of these states (000111000111), (100011100011), (110001110001), (111000111000), (011100011100) or (001110001110).

Then, we enumerate all the transitions from state patterns with h failed pairs to state patterns with $(h+1)$ failed pairs, where h increases from zero to a maximum allowed value by increment 1. We determine the transitions between state patterns as follows. An unbalanced state pattern has no output transitions although it may have input transitions. When we determine the transitions from a balanced state pattern with h failed pairs to state patterns with $(h+1)$ failed pairs, we enumerate the follow-up states derived from the typical state of the balanced state pattern with h failed pairs. A follow-up state of a typical state can be obtained by turning one operating pair of the typical state into failure. Each follow-up state is matched with one of the state patterns with $(h+1)$ failed pairs. If we find such a match, an allowable transition is made. Note there can be more than one realization of the transition between two state patterns. To match a state to a state pattern, we determine if the state is a repetition of the typical state of the state pattern. If so, then we find a match. In this dissertation, to determine if a row vector $\mathbf{b}$, e.g. a system state, is a repetition of another row vector $\mathbf{a}$, we compare them by searching $\mathbf{a}$ in the vector of $(\mathbf{b},\mathbf{b})$. If $\mathbf{a}$ can be found in $(\mathbf{b},\mathbf{b})$, then $\mathbf{a}$ is a repetition of $\mathbf{b}$.

We record the number of realizable transitions from balanced state patterns with h failed pairs to those with $(h+1)$ failed pairs in a transition matrix $\mathbf{Q}_{h,h+1}$ which has m_h rows and m_{h+1} columns where m_h is the number of balanced state patterns that has h failed pairs.

$\mathbf{Q}_{h,h+1}(i,j)$ is the number of realizable transitions from the i^{th} state pattern with h failed pairs to the j^{th} state pattern with $(h+1)$ failed pairs. The matrix does not include the transitions from balanced state patterns to unbalanced ones.

3.2.4 *Reliability Estimation of Systems with Units Performing Single Function*

In this section we discuss the reliability estimation of k-out-of-n pairs:G Balanced systems with all units performing the same function. Such a system is considered balanced if the operating units in the system are symmetric w.r.t. at least one pair of perpendicular axes. We use the Moment Difference to determine the symmetry of such systems.

3.2.4.1 *Successful Event Enumeration*

Each state pattern can be represented by a sequence of numbers of angles between failed units starting from a failed unit in a failed pair to the other failed unit in the same failed pair in an anticlockwise direction, which we call feature segment.

A feature segment of a state pattern is determined based on its typical state by listing the number of angles between failed units starting from a failed unit. Each angle equals π/n. Consider the system in Figure 1.1(a), when pairs 1^*, 2^*, and 3^* fail, starting from unit 1, the numbers of angles between failed units in the anticlockwise direction are (114114). We can choose any one of (114), (141), and (411) as the feature segment. If the system has a different state in the same state pattern, we can still find the feature segment in the sequence of numbers of angles between failed units. Again, consider the system in Figure 1.1(a),

when pairs 1^*, 5^*, and 6^* fail, the system is in the same state pattern as when pairs 1^*, 2^*, and 3^* fail, i.e. the state pattern with three consecutive failed pairs. Starting from unit 1, the number of angles between failed units in the anticlockwise direction are (411411) where we can find all three possible feature segments (114), (141), and (411).

All possible state patterns with a certain number, say h, of failed pairs, are obtained using their feature segments by enumerating all the permutations of h positive integers that sum up to n. Since different feature segments can represent the same state pattern, we eliminate the repetitions by the method introduced above to find the set of unique feature segments from which we obtain the corresponding state patterns and typical states.

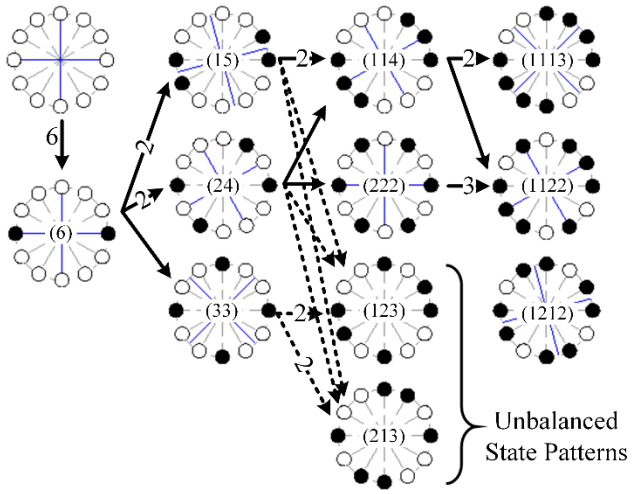


Figure 3.3 State transition diagram for 2-out-of-6 pairs:G Balanced system with units performing single function

An illustrative example of state transition diagram is shown in Figure 3.3 where we list all possible state patterns with zero to four failed pairs, balanced or unbalanced. Each state

pattern is represented by its typical state. Note that the white colored pairs are operating pairs and the black colored pairs are failed pairs. The number series in brackets are the corresponding feature segments. The solid links are transitions between balanced state patterns, and the dashed links are transitions from a balanced state pattern to an unbalanced state pattern. The number of realizations of a transition, which is recorded in transition matrix, is one unless a greater number is associated with the corresponding transition link. For instance, the transition matrix from balanced state patterns with 3 failed pairs to those with 4 failed pairs is $\mathbf{Q}_{34} = \begin{pmatrix} 2 & 1 & 0 \\ 0 & 3 & 0 \end{pmatrix}$ where, for instance, $\mathbf{Q}_{34}(1,2) = 1$ means there is only one possible transition from the 1st state pattern with three failed pairs and feature segment (114) to the 2nd state pattern with four failed pairs and feature segment (1122), as shown in Figure 3.3.

3.2.4.2 *System Reliability Estimation*

Let the probability density function (*pdf*) and the cumulative distribution function (CDF) of the life of any pair of units be f and F respectively, and let $\bar{F} = 1 - F$. When the *pdf* of an individual unit is g and CDF is G, then we have $f(t) = 2g(t)\left[1 - G(t)\right]$ and $\bar{F}(t) = \left[1 - G(t)\right]^2$. The reliability of a k-out-of-n pairs:G Balanced system can be obtained as

$$R_{sys}(t) = \sum_{h=0}^{n-k} P_h^n(t) \tag{3.2}$$

where $P_h^n(t) := \Pr\{h \text{ pairs fail in balanced state pattern by } t\}$ which can be obtained by

$$P_h^n(t) = \eta_h^n \int_{\tau_h=0}^{t} \cdots \int_{\tau_1=0}^{\tau_2} f(\tau_1) \cdots f(\tau_h)\, d\tau_1 \cdots d\tau_h \left[\bar{F}(t)\right]^{n-h}$$
$$= \frac{\eta_h^n}{h!}\left[F(t)\right]^h \left[\bar{F}(t)\right]^{n-h} \tag{3.3}$$

where η_h^n is the number of realizable system state transition paths that lead to balanced state patterns with h failed pairs. Let $\eta_0^n = 1$, and η_h^n when $1 \le h \le n-k$ is obtained as

$$\eta_h^n = \sum \prod_{i=0}^{h-1} \mathbf{Q}_{i,i+1} \tag{3.4}$$

For instance, η_3^6 for transition diagram in Figure 3.3 is

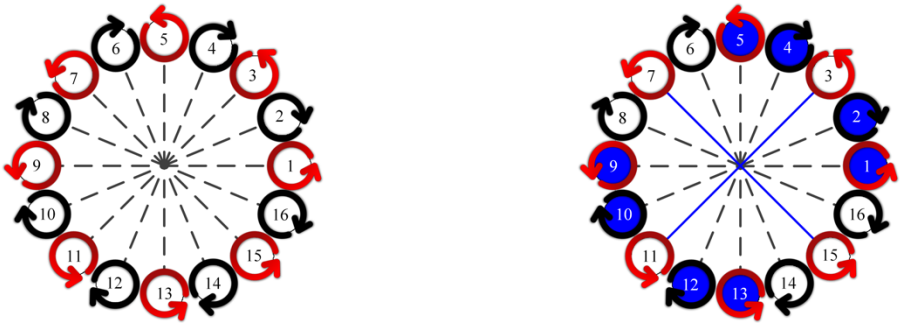
$$\eta_3^6 = \sum \mathbf{Q}_{01}\mathbf{Q}_{12}\mathbf{Q}_{23} = \sum 6 \begin{pmatrix} 2 & 2 & 1 \end{pmatrix} \begin{pmatrix} 2 & 0 \\ 1 & 1 \\ 0 & 0 \end{pmatrix} = 48 \tag{3.5}$$

3.2.5 *Reliability Estimation of Systems with Units Performing Complementary Functions*

In this section, we assume that any two adjacent units perform complementary functions. For example, any two adjacent rotors in an UVA have opposite rotational directions. We refer to any two adjacent rotors as units performing complementary functions as shown in Figure 3.4(a). We assume the units, regardless of their functions, have identical lifetime distributions. The balance requirements for such a system are: (i) the system should be symmetric in a sense that operating units should be symmetric w.r.t. at least a pair of perpendicular axes; (ii) any two adjacent operating pairs should perform complementary functions. Given the second balance requirement is satisfied, we determine the balance of such systems by examining the symmetry of the system using Moment Difference.

When n is odd (even), we should always have an odd (even) number of operating pairs to satisfy the second requirement, and hence corresponding k should be odd (even). An example of such system with all units operating is shown in Figure 3.4(a).

To meet the second balance requirement, we force down the operating pair closest to, not necessarily adjacent to, the failed pair on either side with probability 0.5. Consider the system in Figure 3.4(a), when only unit 1 fails, unit 9 is forced down permanently and either pair 2^* or pair 8^* is forced down with probability 0.5. Suppose that pair 2^* is forced down. When another unit actually fails, say unit 11, then unit 3 is forced down permanently, and either pair 4^* or pair 8^* is forced down.



(a) (b)

Figure 3.4 Examples of *k*-out-of-*n* pairs:G Balanced system with any two adjacent units performing complementary functions: n = 8 in this figure

Due to the balance requirements and the procedure for forcing down operating pairs, we conclude that (i) the system always has an even number of pairs that are either failed or

forced down; (ii) the failed pairs and forced-down pairs always form clusters which have even numbers of pairs, as shown in Figure 3.4(b) where pairs 1^*, 2^*, 4^*, and 5^* are either failed or forced down.

3.2.5.1 *Successful Event Enumeration*

Similarly, we obtain balanced state patterns with different numbers of failed pairs, the state transitions from balanced state patterns with h failed pairs to those with $(h+1)$ failed pairs, and then all realizable system state transition paths that lead to operating system states. In order to obtain all the balanced state patterns with h failed units, the following procedure is followed.

Step 1. Enumerate all the possible permutations of u positive integers that sum to $(n-h)$, where u is even and $2 \le u \le n-h$. Eliminate the repetitions as discussed earlier. Denote each unique permutation as a row vector $\mathbf{v}_u$.

Step 2. For each $\mathbf{v}_u$, sum the elements in odd indices. If the sum equals h, then multiply these elements by 2 to obtain vector $\mathbf{w}_u$. The elements in odd indices in $\mathbf{w}_u$ are the numbers of failed pairs and forced-down pairs, separated by the elements in even indices which are the numbers of operating pairs. We obtain a system state based on each $\mathbf{w}_u$ with the state of each unit denoted as [0/–1] if it is either failed or forced-down and denoted as 1 if it is operating. Note that a failed unit has state 0 and a forced-down unit has state –1, hence we denote the state of a unit that is either failed or forced down as [0/–1]. Denote

the system state as $\mathbf{s}_u$. For instance, let $n=8$, $h=2$, and $u=4$. One vector $\mathbf{v}_4=(1,1,1,3)$ and hence $\mathbf{w}_4=(2,1,2,3)$, based on which we can obtain the state $\mathbf{s}_u$ = ([0/–1] [0/–1] 1 [0/–1] [0/–1] 1 1 1, [0/–1] [0/–1] 1 [0/–1] [0/–1] 1 1 1) as shown in Figure 3.4(b) where the blue colored units are in state [0/–1] and white colored units are in state 1.

Step 3. We only consider the system states $\mathbf{s}_u$'s that are symmetric and hence balanced. We denote the state of each pair with the state of the units in the pair. For each balanced system state $\mathbf{s}_u$, half of the pairs in state [0/–1] are failed pairs and the other half are forced-down pairs. The pairs with state [0/–1] are in one or more clusters. For the pairs in each cluster, we let half of them be in state 0 and the other half in state –1. We enumerate all the combinations to obtain all the possible typical states. Continuing with the example in Figure 3.4(b), the pairs 1^*, 2^*, 4^*, 5^* are in state [0/–1]. There are two ways to assign the two different states, i.e. 0 and –1, to pairs 1^* and 2^*. It is the same with pairs 4^* and 5^*. So there are four typical states and four corresponding state patterns derived from the system state in Figure 3.4(b). The four typical states obtained are (–1 0 1 –1 0 1 1 1 –1 0 1 –1 0 1 1 1), (0 –1 1 –1 0 1 1 1 0 –1 1 –1 0 1 1 1), (–1 0 1 0 –1 1 1 1 –1 0 1 0 –1 1 1 1) and (0 –1 1 0 –1 1 1 1 0 –1 1 0 –1 1 1 1), shown as the typical states 7, 8, 9 and 10 in Figure 3.5.

Step 4. For the same vector $\mathbf{v}_u$ as in step 2, repeat the same procedure in steps 2 and 3 considering the elements in even indices instead of odd indices unless we obtain a state $\mathbf{s}_u$ that is a repetition of the one we found in step 2.

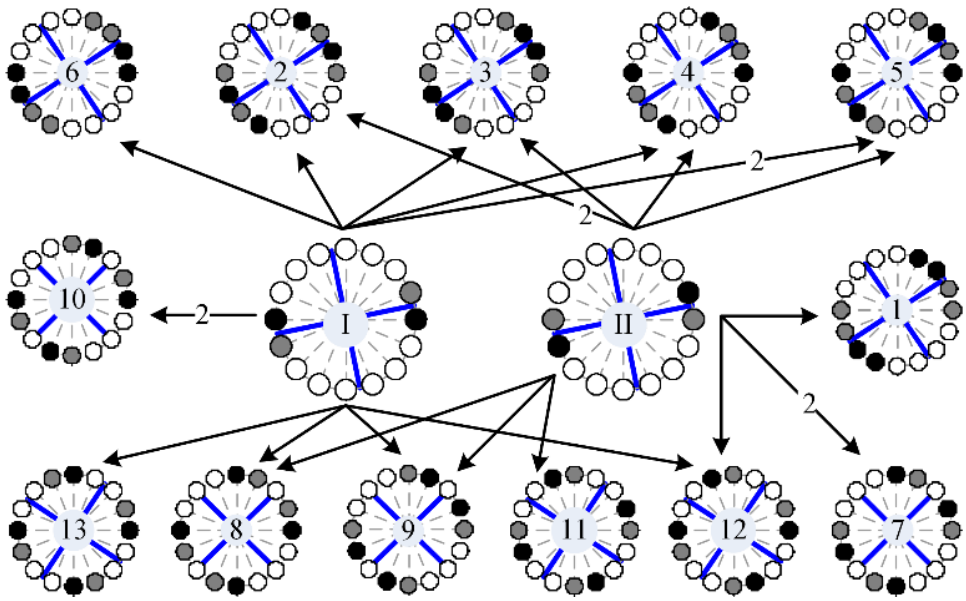


Figure 3.5 Illustration of state transitions from states with one failed pair to those with two failed pairs for a *k*-out-of-8 pairs:G Balanced system with units performing two complementary functions

Now we obtain all the possible balanced state patterns with a certain number of failed units. For each state pattern, we have a typical state. Then we obtain the diagram of system state transition paths as follows. Starting from $h=0$, for each state pattern with h failed units, we enumerate all the follow-up states of its typical state. Here a follow-up state of a state is obtained by turning one of the operating pairs into failure and then forcing down a closest operating pair. For each of the follow-up states, if it is balanced, find the first matching state pattern with $(h+1)$ failed units by enumerating all the state patterns with $(h+1)$ failed units in a fixed order. If a match is found we stop enumeration and add 1 to the corresponding elements in $\mathbf{Q}_{h,h+1}$. Note that steps 2 to 4 introduced above can bring repetitions to the set of the state patterns, so it is necessary to stop enumeration as we find

a match so that the repeated state patterns receive no input transitions. Figure 3.5 shows the transitions from balanced states patterns with one failed pair to those with two failed pairs. The transition matrix is $\mathbf{Q}_{12} = \begin{pmatrix} 0 & 1 & 1 & 1 & 2 & 1 & 0 & 1 & 1 & 2 & 0 & 1 & 1 \\ 1 & 2 & 1 & 1 & 1 & 0 & 2 & 1 & 1 & 0 & 1 & 1 & 0 \end{pmatrix}$.

The first and second rows represent the state patterns I and II in Figure 3.5 with one pair failed and one pair forced down. The columns 1 to 13 represent the state patterns 1 to 13 with two pairs failed and two pairs forced down. The number of transitions between two state patterns is one unless a greater number is associated with the corresponding transition link. For instance, $\mathbf{Q}_{12}(1,5) = 2$ means there are two possible transitions from state pattern I with one failed pair to state pattern 5 with two failed pairs as shown in Figure 3.5.

3.2.5.2 *System Reliability Estimation*

System reliability can be estimated as

$$R_{sys}(t) = \sum_{h=0}^{(n-k)/2} P_h^n(t) \tag{3.6}$$

where

$$\begin{aligned} & P_h^n(t) \\ & = \eta_h^n 0.5^h \int_{\tau_h=0}^{t} \cdots \int_{\tau_1=0}^{\tau_2} \begin{Bmatrix} f(\tau_1)\bar{F}(\tau_1)\cdots \\ f(\tau_h)\bar{F}(\tau_h)d\tau_1 \cdots d\tau_h \end{Bmatrix} \left[\bar{F}(t)\right]^{n-2h} \\ & = \frac{\eta_h^n}{4^h h!}\left[1-\bar{F}^2(t)\right]^h \left[\bar{F}(t)\right]^{n-2h} \end{aligned} \tag{3.7}$$

where η_h^n is the number of realizable system state transition paths that lead to a balanced state with h failed pairs and h forced-down pairs, which is obtained as in the previous section.

3.3 Reliability Estimation of Systems Considering Standby

3.3.1 *System Description and Fundamental Method*

In Section 3.2, we investigate two types of k-out-of-n pairs:G Balanced systems under the assumption that an unbalanced system is considered as failed. In this section, we investigate the two systems further by rebalancing unbalanced system: when unbalanced system states occur, one or more operating pairs are forced down to balance the system according to some balance requirements.

The operating pairs that are forced down for balancing the system are called standby pairs. A standby pair can be resumed into operation when an operating pair fails and resuming the standby pair can bring the system to a balanced state with an additional operating pair. For example, in Figure 1.2(a), the system is unbalanced considering the operating pairs (in white) are not symmetric w.r.t. any axes. Then an operating pair is forced down into standby (in gray) as shown in Figure 1.2(b). The system is now balanced by noting that the operating pairs are symmetric w.r.t. a pair of perpendicular axes. When an additional pair fails, the standby pair is resumed into operation to keep the system in balance and bring one more operating pair, see Figure 1.2(c).

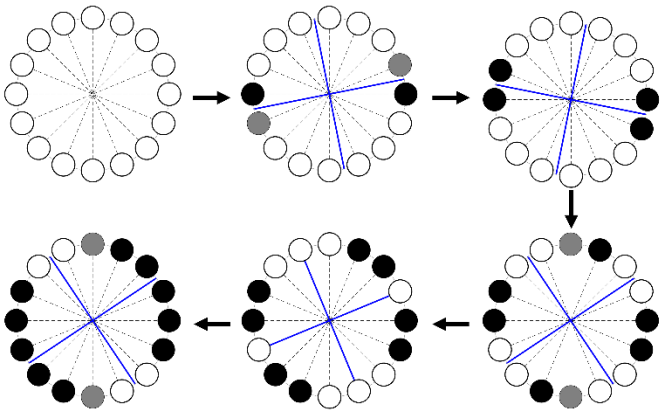
Similarly, to estimate the system reliability, we first enumerate all the successful events; then estimate the probability of each event; and the system reliability is the sum of the probabilities of all successful events.

In this section, we assume that the lifetimes of the individual units follow an exponential distribution with parameter λ.

3.3.2 *Probability Estimation for Successful Events*

A successful event of a k-out-of-n pairs:G Balanced system with standby is an event with g operating pairs, s standby pairs and h failed pairs where $g+s+h=n$ and $g \geq k$. Such an event occurs after a sequence of failures. The forcing-down and resumption of standby pairs, if necessary, always happen at the times of failures. Each pair may experience a sequence of forcing-down and resumption which results in a time period during which the corresponding pair is in standby. Consider the system shown in Figure 3.4(a). A possible system state transition path of the system is shown in Figure 3.6(a) where the unbalanced system is rebalanced by forcing down operating pairs into standby and resuming standby pairs into operation. In Figure 3.6(a), an operating pair is shown in white, a failed pair is in black, and a standby pair is in gray. The perpendicular cross lines are the axes of symmetry for each system state. The algorithm for looking for the standby pairs and axes of symmetry is discussed later. The sequence of failures and standby of each pair of units is shown in Figure 3.6(b) where the failure times are denoted as τ_i where $i=1$ to 5, and all the eight pairs are listed vertically. For instance, according to Figure 3.6(b) pair

2^* is forced down into standby at τ_1 when pair 1^* fails; is resumed into operation at τ_2 since its resumption can bring the system back to balance without forcing down additional pairs; and fails at τ_5.



(a)

	$[\tau_1, \tau_2)$	$[\tau_2, \tau_3)$	$[\tau_3, \tau_4)$	$[\tau_4, \tau_5)$	$[\tau_5, t)$
1*	Failed				
2*	Standby				Failed
3*				Failed	
4*			Failed		
5*			Standby		Standby
6*					
7*					
8*		Failed			

(b)

Figure 3.6 Illustration example for calculating the probabilities of successful events

The probability of a successful event, assuming that all units have identical constant failure rate $\lambda / 2$, is calculated as

$$P_h^w(t;\boldsymbol{\delta}_h^w) =$$

$$\int_{\tau_h=0}^{t}\cdots\int_{\tau_2=0}^{\tau_3}\int_{\tau_1=0}^{\tau_2}\left\{\begin{array}{l}(\lambda)^h\exp\left(-\lambda\sum_{i=1}^{h}\left(\tau_i-\delta_{h,x_i^*}^w\right)\right)\\ \times\exp\left(-\lambda\sum_{y^*\in Y_h^w}\left(T_{h,y^*}^w-\delta_{h,y^*}^w\right)\right)\\ \times\exp\left(-\lambda\sum_{z^*\in G_h^w}\left(t-\delta_{h,z^*}^w\right)\right)\end{array}\right\}d\tau_1 d\tau_2\cdots d\tau_h \qquad (3.8)$$

where $P_h^w\left(t;\boldsymbol{\delta}_h^w\right)$ denotes the probability of an event that has h failed pairs where superscript w is used to distinguish different events that all have the same value for h at time t; $\boldsymbol{\delta}_h^w=\left[\delta_{h,1^*}^w,\delta_{h,2^*}^w,\cdots,\delta_{h,n^*}^w\right]$ is a vector that contains the standby periods for individual pairs; λ is the hazard rate of individual pairs and is twice the hazard rate of individual units; τ_i $(i=1,\cdots,h)$ is the time of the i^{th} failure, x_i^* is the identity number of the i^{th} failed pair; Y_h^w is the set of identity numbers of standby pairs at time t; y^* is the identity number of standby pair; T_{h,y^*}^w is the time when pair y^* is forced down the last time (and not resumed by time t) the value of which is one of the τ_i's; G_h^w is the set of identity numbers of operating pairs at time t; z^* is the identity number of operating pair. Note that when the last time a pair y^* is forced down at T_{h,y^*}^w and is still in standby state at time t, its standby period, δ_{h,y^*}^w, should exclude $\left(t-T_{h,y^*}^w\right)$ since the last forcing-down and resumption cycle is not completed.

The event shown in Figure 3.6 is a successful event if we consider a 2-out-of-8 pairs:G Balanced system with any two adjacent units performing two complementary functions.

By time t, there are five failed pairs, one standby pair and two operating pairs, so $h=5$, $s=1$, and $g=2$. The identity numbers for the five failed pairs in order are $x_1^*=1^*$, $x_2^*=8^*$, $x_3^*=4^*$, $x_4^*=3^*$, $x_5^*=2^*$. The only standby pair has identity number 5^*, i.e. $Y_h^w=\{5^*\}$ and $T_{h,5^*}^w=\tau_5$. The set of operating pairs is $G_h^w=\{6^*,7^*\}$. In addition, $\delta_{h,2^*}^w=\tau_2-\tau_1$ since pair 2^* is forced down at τ_1 and resumed at τ_2; and $\delta_{h,5^*}^w=\tau_4-\tau_3$ since pair 5^* is forced down at τ_3 and resumed at τ_4 before it is forced down for the last time at τ_5. Note that $\delta_{h,5^*}^w$ does not include $\left(t-T_{h,5^*}^w\right)$ since even though pair 5^* is in standby during the period $[\tau_5,t)$ it is not resumed by t.

Since T_{h,y^*}^w's are τ_i's and $\delta_{h,x_i^*}^w$'s are the sum of τ_i's and $-\tau_i$'s, the exponent part of Eq. (3.8) is overall the sum of τ_i's and $-\tau_i$'s. Denote the overall coefficient of τ_i as $\alpha_{h,i}^w$ which can be obtained based on Eq. (3.8). Consider the event in Figure 3.6, the $\alpha_{h,i}^w$'s for τ_i's are $-2\lambda,\ 0,\ -2\lambda,\ 0,\ -2\lambda$ respectively for $i=1$ to 5.

The general expression for $P_h^w(t;\boldsymbol{\delta}_h^w)$ can be simplified as

$$P_h^w(t;\boldsymbol{\delta}_h^w)=\left(\lambda\right)^h\exp\left(-\lambda\cdot g_h^w\cdot t\right)$$
$$\int_{\tau_h=0}^{t}\cdots\int_{\tau_2=0}^{\tau_3}\int_{\tau_1=0}^{\tau_2}\exp\left(\sum_{i=1}^{h}\alpha_{h,i}^w\tau_i\right)d\tau_1 d\tau_2\cdots d\tau_h \qquad (3.9)$$

where g_h^w is the number of operating pairs when there are *h* failed pairs.

We denote the integral in Eq. (3.9) as

$$I_h^w(h,t) \coloneqq \int_{\tau_h=0}^{t} \cdots \int_{\tau_2=0}^{\tau_3} \int_{\tau_1=0}^{\tau_2} \exp\left(\sum_{i=1}^{h} \alpha_{h,i}^w \tau_i\right) d\tau_1 d\tau_2 \cdots d\tau_h \tag{3.10}$$

The iterations to obtain the closed form expression of a general integral $I_h^w(h,t)$ are as follows.

When $h=1$

$$I_h^w(1,t) = \left(-1+e^{\alpha_{h,1}^w t}\right) \Big/ \alpha_{h,1}^w \tag{3.11}$$

When $h>1$, we obtain $I_h^w(2,t)$ based on $I_h^w(1,t)$, then obtain $I_h^w(3,t)$ based on $I_h^w(2,t)$, until we obtain $I_h^w(h,t)$ based on $I_h^w(h-1,t)$. $I_h^w(x,t)$ has x terms where $x=1$ to h. The first term of $I_h^w(x,t)$ is

$$\frac{(-1)^x\left(1-e^{\alpha_{h,x}^w t}\right)}{\left(\alpha_{h,1}^w+\alpha_{h,2}^w+\cdots+\alpha_{h,x-1}^w\right)\cdots\left(\alpha_{h,x-2}^w+\alpha_{h,x-1}^w\right)\alpha_{h,x-1}^w\alpha_{h,x}^w} \tag{3.12}$$

The other $(x-1)$ terms in ascending order are

$$(-1)^{x+i}\frac{1-\exp\left(\sum_{j=0}^{i}\alpha_{h,x-j}^w t\right)}{\Omega_h^w(x-1,i)\times\sum_{j=0}^{i}\alpha_{h,x-j}^w} \tag{3.13}$$

where $i=1$ to $(x-1)$ and $\Omega_h^w(x-1,i)$ is the denominator of the i^{th} term in $I_h^w(x-1,t)$. For example, based on Eq. (3.12) when $h=2$ the first term in $I_h^w(2,t)$ is

$$\frac{(-1)^2\left(1-e^{\alpha_{h,2}^w t}\right)}{\alpha_{h,1}^w \alpha_{h,2}^w} \tag{3.14}$$

Then we obtain the second term for $I_h^w(2,t)$ based on Eq. (3.13) with $i=1$ and $x=2$ as

$$(-1)^{2+1}\frac{1-e^{\left(\alpha_{h,2}^w+\alpha_{h,1}^w\right)t}}{\Omega_h^w(1,1)\left(\alpha_{h,2}^w+\alpha_{h,1}^w\right)} \tag{3.15}$$

where $\Omega_h^w(1,1)=\alpha_{h,1}^w$ is the denominator of the 1st term, which is also the only term, in $I_h^w(1,t)$ as shown in Eq. (3.11).

Note that when we have zero-valued terms, i.e. sum of $\alpha_{h,i}^w$'s, in any denominators of $I_h^w(h,t)$, we take the limit of the expression $I_h^w(h,t)$ as the $\alpha_{h,i}^w$'s that compose the zero-valued terms reach their extreme values. Then we substitute the other $\alpha_{h,i}^w$'s that are not in the zero-valued terms using their numerical values. For example, when $h=3$ based on $I_h^w(2,t)$ we have

$$\begin{aligned} I_h^w(3,t) = &\frac{e^{\alpha_{h,3}^w t}-1}{\alpha_{h,2}^w\alpha_{h,3}^w\left(\alpha_{h,2}^w+\alpha_{h,1}^w\right)} - \frac{e^{\left(\alpha_{h,3}^w+\alpha_{h,2}^w\right)t}-1}{\alpha_{h,1}^w\alpha_{h,2}^w\left(\alpha_{h,3}^w+\alpha_{h,2}^w\right)} \\ &+\frac{e^{\left(\alpha_{h,3}^w+\alpha_{h,2}^w+\alpha_{h,1}^w\right)t}-1}{\alpha_{h,1}^w\left(\alpha_{h,2}^w+\alpha_{h,1}^w\right)\left(\alpha_{h,3}^w+\alpha_{h,2}^w+\alpha_{h,1}^w\right)} \end{aligned} \tag{3.16}$$

If $\alpha_{h,1}^w$, $\alpha_{h,2}^w$, $\alpha_{h,3}^w$, and $\left(\alpha_{h,3}^w+\alpha_{h,2}^w\right)$ are nonzero but $\left(\alpha_{h,2}^w+\alpha_{h,1}^w\right)=0$, then $I_h^w(3,t)$ in Eq. (3.16) cannot be obtained directly by substituting the $\alpha_{h,i}^w$'s with their values. We first take

the limit of $I_h^w(3,t)$ as $\alpha_{h,1}^w$ and $\alpha_{h,2}^w$ reach their values together or, in other words, when $\left(\alpha_{h,2}^w + \alpha_{h,1}^w\right)$ reaches zero, which results in an equation with only $\alpha_{h,3}^w$; then we substitute $\alpha_{h,3}^w$ to obtain the value of $I_h^w(3,t)$.

3.3.3 *Fundamental Method of Successful Event Enumeration*

Starting from a system state with one failure, we enumerate its follow-up states, the further follow-up states of the follow-up states and so on until we exhaust all the possible system states with at least k operating pairs and all the possible transitions between these states. A follow-up state of a state can be obtained by turning one of the operating pairs in the current system into failure and rebalancing it if the additional failure results in an unbalanced system. A successful event is a transition path that leads to a successful system state, i.e. a state with at least k operating pairs. For instance, Figure 3.7 shows the state transition diagram for 2-out-of-6 pairs:G Balanced system with units performing single function considering standby, where units in white are operating, units in black are failed, and units in gray are in standby. As shown in Figure 3.7, the successful events of a 2-out-of-6 pairs:G Balanced system are all the transition paths that lead to states 1 to 11 and the states that have the same state patterns with states 1 to 11.

The procedure of enumerating the successful events will be discussed later for specific systems. For each system we first introduce a heuristic to find the axis of symmetry for an unbalanced system; then discuss the enumeration of successful events.

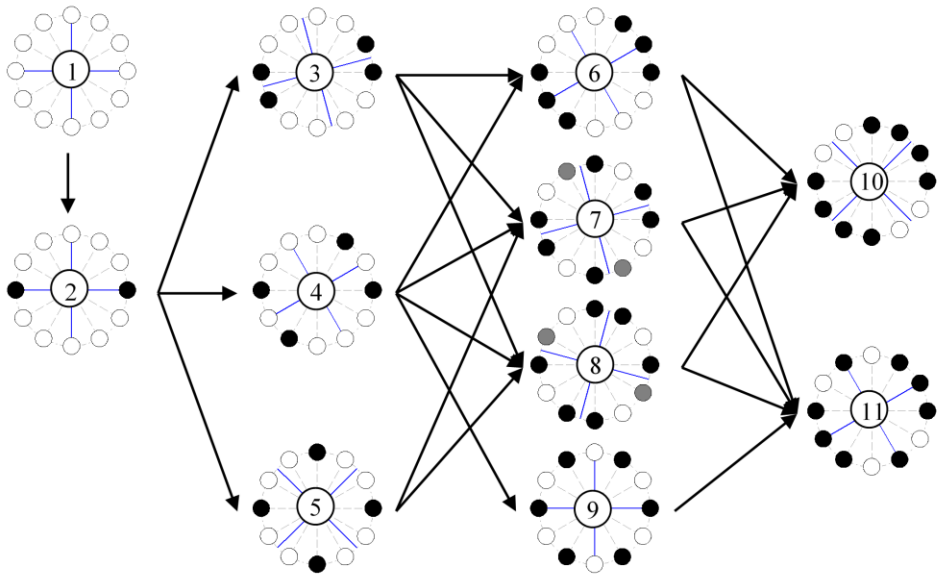


Figure 3.7 State transition diagram for 2-out-of-6 pairs:G Balanced system with units performing single function

3.3.4 *Successful Event Enumeration for Systems with Units Performing Single Function*

3.3.4.1 *Axis of Symmetry Algorithm*

We use Moment Difference as introduced in Section 3.2 to quantify the symmetry of the system; or determine an initial candidate axis, from which we begin to enumerate a series of candidate axes until an axis of symmetry is found.

A system is already symmetric: (i) if there is only one failed pair, then the axes of symmetry are along or perpendicular to the failed pair; (ii) if there are two failed pairs, then an axis of symmetry is the middle axis of the two pairs; (iii) if the failed pairs are consecutively

arranged, then an axis of symmetry is the middle axis of the failed pairs. If the state of the system does not fall into these special cases, then the following procedure is implemented.

A candidate axis of symmetry is either along a pair of units, or in the middle of two adjacent pairs. Due to the orthogonal relationship of the candidate axes, only n unique pairs of perpendicular candidate axes need to be considered. Beginning from any arbitrary candidate axis pair and ending with the n^{th} afterwards in an anticlockwise direction, we calculate MDs w.r.t. each candidate axis pair.

We then calculate the difference between the two MDs w.r.t. the two axes in each candidate axis pair, which we call MD difference. When a candidate axis pair results in the minimum absolute value of MD difference, it is set as the initial candidate axis pair. In the two MDs for the initial candidate axis pair, if the MD that has the smaller absolute value is not zero, then we choose the corresponding candidate axis as the initial candidate axis; otherwise, we choose the axis that has the greater absolute value of MD. As we find the initial candidate axis, we set its corresponding MD as the minimum MD.

When there are ties in the absolute values of MD differences for multiple axis pairs, it is observed that the minimum value of MD difference is always zero. In this case, we choose the pair of axes that result in the smallest MD in terms of absolute value as initial candidate axis pair and either one of them as the initial axis. Further ties are broken arbitrarily.

When the minimum MD is zero and *n* is no more than 30, the system is already symmetric w.r.t. the initial candidate axis. When the minimum MD is zero but *n* is greater than 30, we use an additional condition to determine its symmetry as discussed in Section 4.2.1.

When the system is not symmetric, we enumerate the $\lfloor n/2 \rfloor$ candidate axes on each side of the initial candidate axis as well as the initial candidate axis itself in order to find the pairs of operating units to force down to regain system balance. We start from the initial candidate axis and then those closest to the initial candidate axis. In addition, we start from the left of the initial candidate axis if minimum MD is greater than zero and from right otherwise. We stop when we reach an axis that needs only one pair of operating unit to be forced down, or when we have enumerated *n* axes. Note that when we enumerate the candidate axes, we consider the standby pairs forced down in previous stages as operating pairs. If it is no longer necessary for them to be in standby, they can resume operation.

Among all the candidate axes enumerated, we choose the one that results in the least number of standby pairs and hence the most operating pairs. If there are more than one such axes, we use other criteria to choose the optimal one. For example, we choose the axes resulting in the smallest standard deviation of the numbers of angles between any two consecutive operating units. Other criteria can be applied according to specific system requirements in practice.

3.3.4.2 *Successful Event Enumeration*

The procedure of enumerating the successful events (except the event that system has no failure at all) is as follows.

Step 1. If $n-k \geq 1$, choose one of the n pairs and assume it fails to obtain a successful system state S_1. Denote this event as E_1. The realization of this event is n. The number of failed pairs is 1, the number of operating pairs is $(n-1)$, and no standby pairs are involved.

Step 2. If $n-k \geq 2$, every transition path that leads to the follow-up states of S_1 is a successful event with realization n. Denote the successful system states obtained in this step as $\{S_2^w\}$ and the successful events that lead to corresponding successful system states as $\{E_2^w\}$.

Step 3. If $n-k \geq 3$, enumerate all the follow-up states of $\{S_2^w\}$. Each follow-up state with at least k operating pairs is a successful system state and the transition path that leads to this state is a successful event. Each successful event has n realizations. Denote the successful system states and events obtained in this step as $\{S_3^w\}$ and $\{E_3^w\}$ respectively. Standby starts to appear in some successful events in this step. For each E_h^w we record the times at which each pair is forced down into standby and resumed into operation in a matrix, $\mathbf{B}_h^w$, with n rows and h columns, where h is the number of failures and in this step $h=3$. Each row of $\mathbf{B}_h^w$ is a vector composed of 0, –1, and 1. $\mathbf{B}_h^w(l,i)=-1$ means the l^{th} pair (pair

l^* in the system) is forced down into standby at τ_i; $\mathbf{B}_h^w(l,i)=1$ means the l^{th} pair is resumed into operation at τ_i; and $\mathbf{B}_h^w(l,i)=0$ means otherwise.

Step 4. If $n-k \geq h$ where $h \geq 4$, enumerate all the follow-up states of the set of successful system states with $(h-1)$ failed pairs in $\{S_{h-1}^w\}$. Record all the successful system states and successful events obtained in this step in the sets $\{S_h^w\}$ and $\{E_h^w\}$. Each successful event has n realizations. The first $(h-1)$ columns of $\mathbf{B}_h^w$ equal to $\mathbf{B}_{h-1}^v$ when E_h^w is derived from E_{h-1}^v. The h^{th} column of $\mathbf{B}_h^w$ is obtained based on the forcing-down or resumption of the pairs when we reach successful system state S_h^w from S_{h-1}^v.

In each step, we record the number of failed pairs, h, and the set of identity numbers of operating pairs, G_h^w, for each successful event E_h^w.

To reduce computational effort, the successful events can be aggregated into groups according to the three features: the numbers of failed pairs h and operating pairs g_h^w, and the values of $\alpha_{h,i}^w$'s, which can be obtained from $\mathbf{B}_h^w$ using Eq. (3.17)

$$\boldsymbol{\alpha}_h^w = \lambda\left(\sum_{column} \mathbf{B}_h^w - \mathbf{e}_h\right) \tag{3.17}$$

where $\boldsymbol{\alpha}_h^w$ is the vector of $\alpha_{h,i}^w$ from $i=1$ to h for successful event E_h^w; $\sum_{column} \mathbf{B}_h^w$ is a sum of each column of $\mathbf{B}_h^w$; and $\mathbf{e}_h$ is a row vector of 1's with h columns. The grouping

can be carried out in each enumerating step after the complete sets of $\{S_h^w\}$ and $\{E_h^w\}$ are obtained. Successful events in each group have the same probability.

3.3.5 *Successful Event Enumeration for Systems with Units Performing Complementary Functions*

3.3.5.1 *Axis of Symmetry Algorithm*

Again, to meet the balance requirement for the system that any two adjacent operating pairs should perform complementary functions, we force down the operating pair closest to, not necessarily adjacent to, the failed pair on either side with probability 0.5 immediately after a failure occurs.

In addition, note that the forced-down pairs mentioned in the algorithm of this section are equivalent to standby pairs.

First, we introduce some simple rules for special cases in which the system is already balanced and the axes of symmetry are easily found.

Rule 1. When there are only one failed pair and one forced-down pair, the system is balanced. An axis of symmetry is along the diameter in the middle of the two pairs.

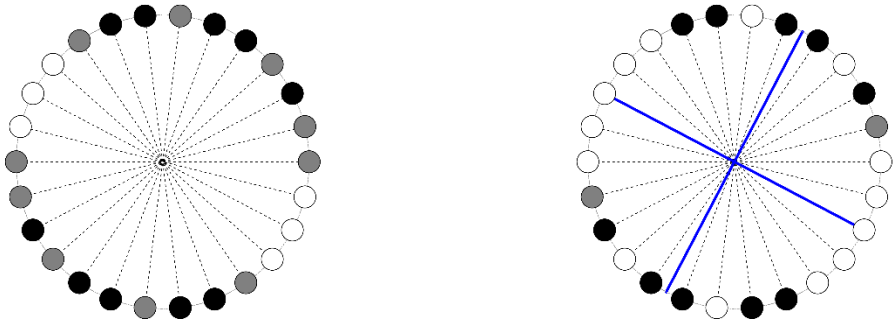
Rule 2. When there are two failed pairs and two corresponding pairs that are forced down, and the four pairs are not consecutively arranged, the system is balanced. An axis of symmetry is along the diameter in the middle of the four pairs.

Rule 3. When the failed and forced-down pairs are consecutively arranged, the axis of symmetry can be found by the following rules.

(i) If all the pairs are either failed or forced-down and n is odd, resume one of the forced-down pairs if any. Then an axis of symmetry is along this resumed pair. If there is more than one forced-down pair, they have equal probability to be resumed. If all the pairs are failed, then no resumption occurs and the system is symmetric.

(ii) If all the pairs are either failed or forced-down and n is even: (a) If there are at least two forced-down pairs with complementary functions, then resume two of the forced-down pairs with complementary functions. An axis of symmetry is in the middle of the two pairs. When two pairs are to be resumed, if there are more than two forced-down pairs with complementary functions, then a criterion such as the standard deviation of the angles between operating units can be used to select the appropriate pairs. (b) Otherwise, no resumption occurs and the system is symmetric with all pairs failed or forced down.

(iii) If there are still operating pairs in the system, we find all the forced-down pairs and all possible combinations of the forced-down pairs. For each combination, we resume the forced-down pairs in it and examine if the state after resumption is balanced. A balanced state requires that the clusters of failed and forced-down pairs always have an even number of pairs arranged consecutively and the remaining operating units are symmetric w.r.t. at least a pair of perpendicular axes. We examine the symmetry by calculating MD considering the weight of a forced-down unit as 0. Note that all the failed and forced-down pairs are consecutively arranged before resumption, which form a cluster. After resumption, the cluster is divided into several clusters. Consider Figure 3.8(a), the failed and forced-down pairs form a cluster. Three smaller clusters are formed after four forced-down pairs are resumed into operation, as shown in Figure 3.8(b). In order to obtain a system state with the most operating pairs, when enumerating the combinations of forced-down pairs we start with the combinations with the greatest number of forced-down pairs. Consider Figure 3.8(a), we start with the combinations with all the five forced-down pairs, then move to combinations with four forced-down pairs, and so on. When there are no combinations that result in a balanced state, then no resumption occurs. The system is symmetric with an axis of symmetry in the middle of the failed and forced-down pairs.



(a) (b)

Figure 3.8 Example of consecutively arranged failed pairs and forced-down pairs in a system

When the system state does not fall into the special cases discussed above (see Figure 3.9 for example), we enumerate all the unique candidate axes of symmetry. A candidate axis of symmetry is either along an operating pair or in the middle of two pairs. Due to the orthogonal relationship of the candidate axes, only n unique pairs of perpendicular candidate axes need to be considered. In this system, a candidate axis of symmetry does not exist along or perpendicular to a failed pair. Starting from the axis along an arbitrary pair and rotating anticlockwise by step of $\pi / 2n$, we consider n pairs of perpendicular candidate axes except those along or perpendicular to failed pairs.

First, for each pair of perpendicular candidate axes, we make the system symmetric w.r.t. the axes by forcing down additional operating pairs without considering the complementary functions of the pairs. If any forced-down pairs are along the candidate axes, then they are resumed into operation. Now the system state may not be balanced, e.g. some clusters of failed and forced-down pairs may have an odd number of pairs. Consider

the system shown in Figure 3.9, we obtain six pairs of perpendicular candidate axes and the corresponding system states after forcing down additional operating pairs as shown in Figure 3.10(a) to Figure 3.10(f). The states shown in Figure 3.10(d) and Figure 3.10(f) are not balanced because they have clusters with odd numbers of failed and forced-down pairs.

Second, for each symmetric system corresponding to each candidate axis pair, we resume redundant forced-down pairs in order to obtain a balanced system state. If any two forced-down pairs are symmetric w.r.t. the candidate axes, then both the forced-down pairs are considered redundant. We find all the redundant forced-down pairs and all the combinations of these pairs. Note that any two forced-down pairs that are symmetric w.r.t. the candidate axis pair should be included in the same combination together. For each of these combinations, we examine the balance of the system state obtained by resuming the forced-down pairs in the combinations. If the obtained system state is balanced, we call the combination a balancing combination. This might result in several balancing combinations exhausting all the combinations in a symmetric system. Among all the balancing combinations, we choose the combination that results in the optimal system state based on some criteria and resume the forced-down pairs accordingly. The criteria may include the number of operating pairs (the larger the better); the standard deviation of the angles between operating pairs (the smaller the standard deviation, the more uniform the spatial distribution of the operating pairs); and the number of times that all the pairs are forced down or resumed (the smaller the better since each forcing down or resumption is likely to increase the risk of the system's failure).

The system states after the second step are as shown in Figure 3.10(g) to Figure 3.10(l). It is possible that no balancing combinations are obtained. For system states shown in Figure 3.10(b) and Figure 3.10(f), no balancing combinations of redundant forced-down pairs are found. In this case, the system will not change after the second step as shown in Figure 3.10(h) and Figure 3.10(l).

Finally, among all the pairs of perpendicular candidate axes, we eliminate the axis pairs that result in unbalanced system states. For example, the candidate axis pair corresponding to system states in Figure 3.10(l) should be eliminated since the corresponding system state has clusters with odd numbers of failed and forced-down pairs and hence is not balanced. Among the remaining candidate axes, we choose the one that results in the optimal system state based on the aforementioned criteria. Consider the system in Figure 3.9, the state in Figure 3.10(i) is the optimal state since on one hand it has the most operating pairs as the state in Figure 3.10(j); on the other hand, to obtain the state in Figure 3.10(i) from the state in Figure 3.9, only two pairs have to be forced down or resumed; but four pairs have to be forced down or resumed to obtain the state in Figure 3.10(j). Note that both the states in Figure 3.10(i) and Figure 3.10(j) have the same standard deviation of the angles between operating units. So the axis pair corresponds to Figure 3.10(i) is chosen as the final axes of symmetry.

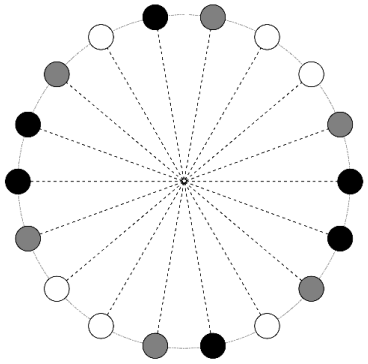


Figure 3.9 Example of system state that does not fall into special cases

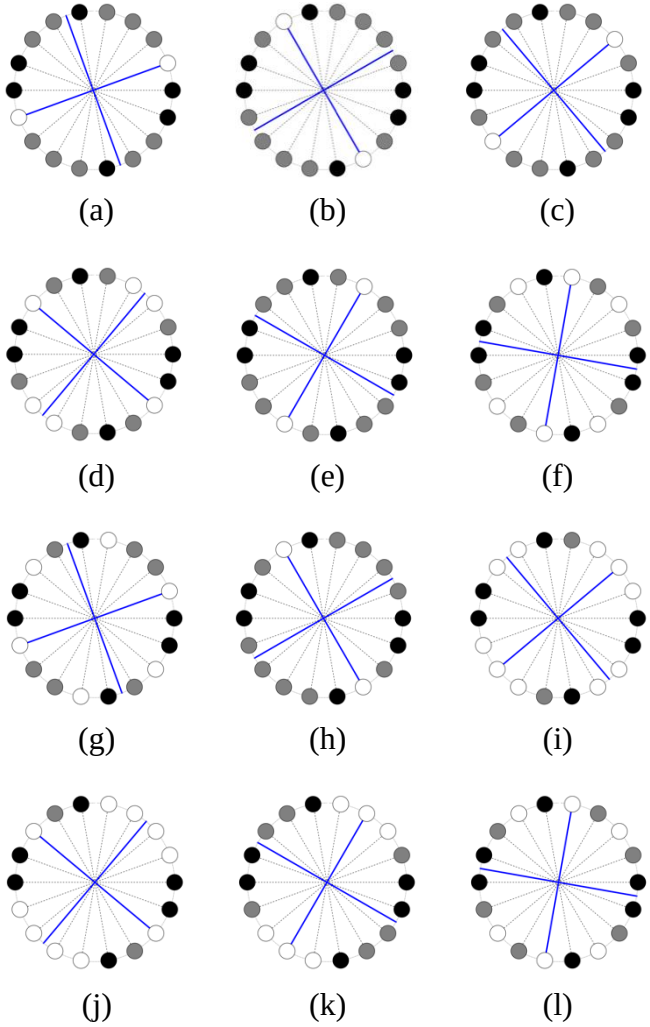


Figure 3.10 System state after being balanced w.r.t. candidate axes

3.3.5.2 *Successful Event Enumeration*

Since we assume identical lifetime distributions, the computation can be simplified by noting that the two groups of system state transition paths, which start from respectively the two possible options of standby pairs for the first failed pair, have the same probability. Consider the state transition diagram in Figure 3.11, the group of paths with states 2, 4, 5, 8, and 9 has the same probability as the group of paths with states 3, 6, 7, 10, and 11. We only need to enumerate the system state transition paths in one group and estimate the corresponding probabilities.

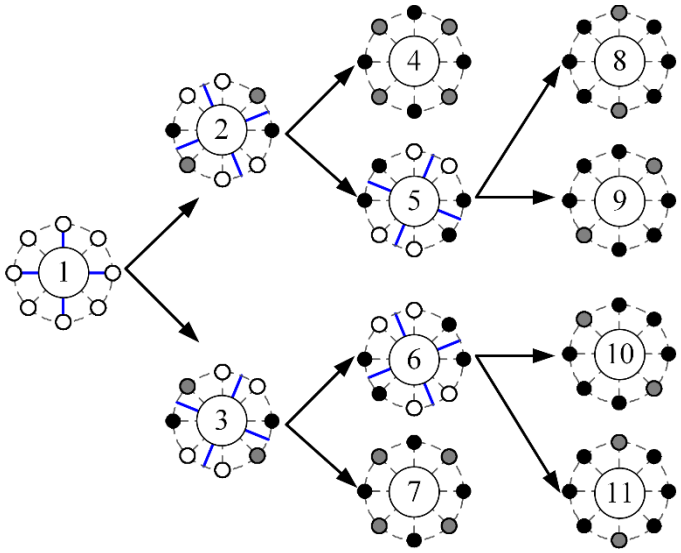


Figure 3.11 State transition diagram for *k*-out-of-4 pairs:G Balanced system with units performing complementary functions

The procedure of enumerating the successful events (except the event that system has no failure at all) is as follows.

Step 1. If $k < n$, choose one of the n pairs, turn it into failure, and turn a consecutive pair into standby to obtain a successful system state S_1. Denote this event as E_1. The realization of this event is $0.5n$ since the standby pair is chosen with probability 0.5. The number of failed pair is 1, the number of operating pairs is $(n-2)$, and 1 pair is forced down into standby.

Step 2. If $n-k \geq 2$, enumerate all the follow-up states of S_1. Here a follow-up state of a state is obtained by turning one of the operating pairs into failure, turning a closest operating pair into standby, and balancing the system by resuming or forcing down standby pairs if necessary. A follow-up state with at least k operating pairs is a successful system state and the transition path that leads to it is a successful event. Denote the successful system states obtained in this step in $\{S_2^w\}$ and the successful events that lead to corresponding successful system states in $\{E_2^w\}$. For each E_2^w we record the corresponding matrix $\mathbf{B}_2^w$, which is as defined before. Record the numbers of failed and operating pairs for each successful event. The realization of a successful event equals to the realization of the event that leads to the successful event multiplied by z. $z = 0.5$ if there are at least two operating pairs left in the system after the latest failure occurs; $z = 1$ otherwise.

Step 3. If $n-k \geq 2h-2$ where $h > 2$, enumerate all the follow-up states of the set of successful system states with $(h-1)$ failed pairs in $\{S_{h-1}^w\}$. Record the set of successful events obtained in this step, $\{E_h^w\}$, and its corresponding $\{S_h^w\}$ and $\mathbf{B}_h^w$.

In each step, we record the number of failed pairs, h, and the set of identity numbers of operating pairs, G_h^w, for each successful event E_h^w.

Similarly, to simplify the computation of the probability of the successful events, the successful events can be aggregated into groups according to the three features: the numbers of failed pairs h and operating pairs g_h^w, and the values of $\alpha_{h,i}^w$'s.

3.4 Numerical Examples

3.4.1 *Numerical Example 1*

Consider a 6-out-of-n pairs:G Balanced system with units performing a single function where the unbalanced system is considered as a failure. The lifetimes of individual units follow Weibull distribution with scale parameter 40 and shape parameter 2, i.e. Weibull (40, 2). The system reliabilities with n ranging from 6 to 12 and $n=16$ are shown in Figure 3.12. When the number of redundant pairs increases from 0 to 10, the system reliability increases and then decreases. The system reliability value reaches a maximum when the system has 9 pairs.

We also obtain the MTTF and its confidence interval as shown in Figure 3.13(a). The MTTF increases as n increases and decreases after MTTF reaches its maximum value.

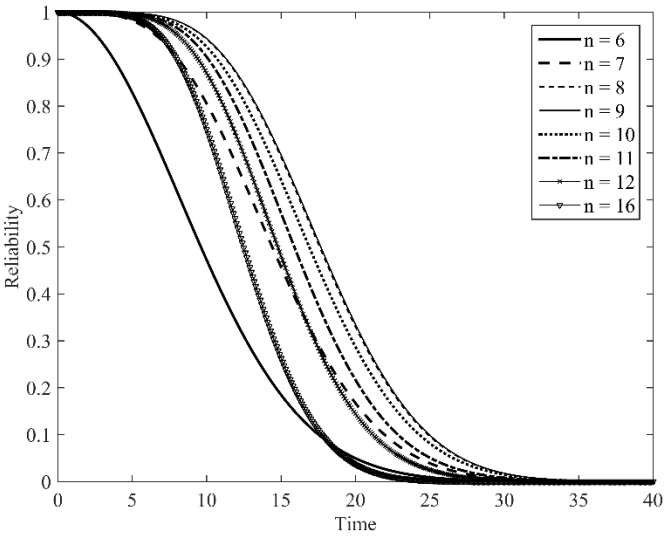


Figure 3.12 System reliability of 6-out-of-*n* pairs:G Balanced systems with units performing single function considering unbalance as failure

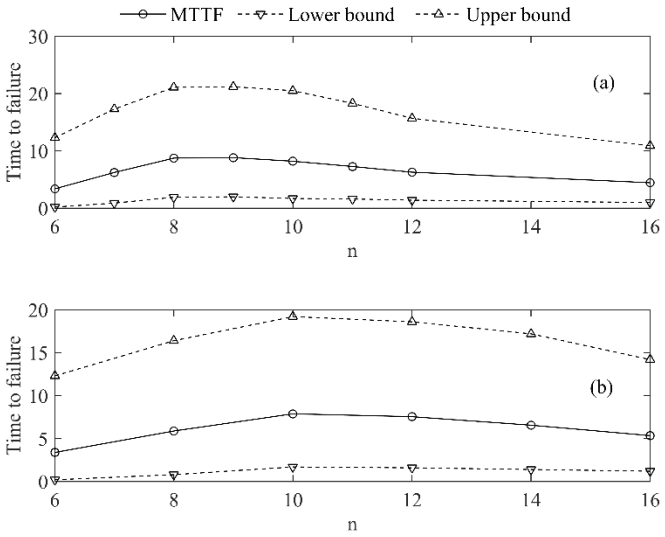


Figure 3.13 Mean times to failure with confidence interval

We also estimate the MTTF of 6-out-of-n pairs:G Balanced system with units performing two complementary functions where the unbalanced system is considered as a failure. The

lifetimes of individual units follow Weibull (40, 2). We plot the MTTF and its confidence interval in Figure 3.13(b) which shows that MTTF first increases and then decreases when n increases. In addition, the MTTF reaches its maximum value when $n = 10$.

3.4.2 *Numerical Example 2*

In order to investigate the effect of standby pairs, we compare the reliability of the systems in the two scenarios: (i) unbalanced system considered as a failure; and (ii) unbalanced system rebalanced by forcing down operating pairs into standby and/or resuming standby pairs into operation. Let the individual units have exponentially *i.i.d.* lifetimes with failure rate 0.025.

We estimate the reliability of the k-out-of-6 pairs:G Balanced system with all units performing the same function in the two scenarios, as shown in Figure 3.14 where dashed curves are the reliabilities in the first scenario and solid curves are the reliabilities in the second scenario. As shown in Figure 3.14, the system has higher reliability in the second scenario when $k \leq 2$; and the same reliability otherwise. The reason is that when an unbalanced system state occurs, e.g. system state shown in Figure 1.2(a), rebalancing system by forcing down operating pairs into standby will result in a balanced system, as shown in Figure 1.2(b). When $k \leq 2$, the system is considered as failed in the first scenario since no standby is considered; but it is considered as operating in the second scenario.

We then estimate the reliability of the k-out-of-8 pairs:G Balanced system with units performing two complementary functions in the two scenarios, as shown in Figure 3.15

where dashed curves are the reliabilities in the first scenario and solid curves are the reliabilities in the second scenario. As shown in Figure 3.15, the system has the same reliability in both scenarios only when $k = 8$ and has greater reliability in the second scenario otherwise. The reason is also that when $k \leq 6$ some system states will only be considered as operating after it is rebalanced by forcing down some operating pairs into standby and resuming standby pairs back to operation in the second scenario, which brings more successful events and hence higher system reliability.

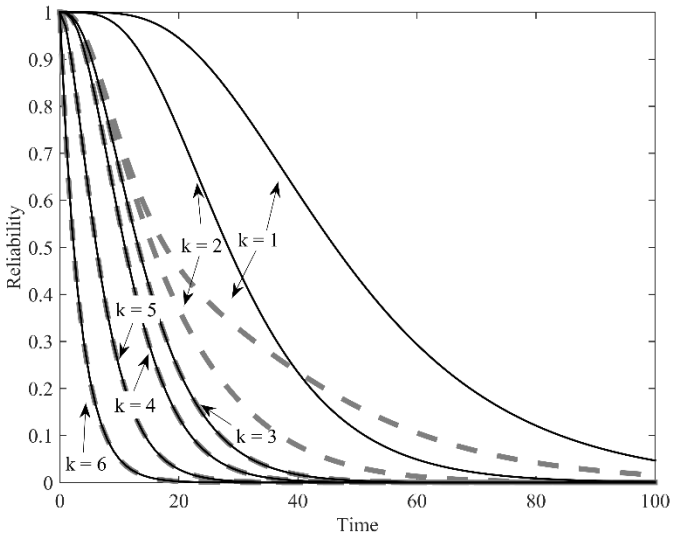


Figure 3.14 Reliability plots for *k*-out-of-6 pairs:G Balanced systems with units performing single function in two scenarios

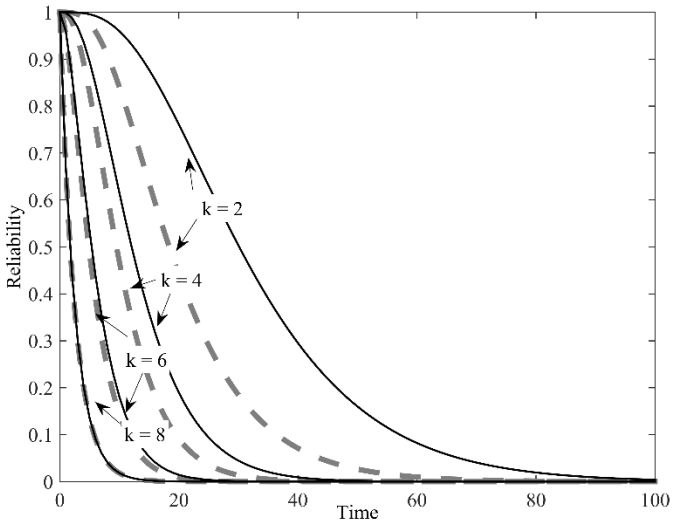


Figure 3.15 Reliability plots for *k*-out-of-8 pairs:G Balanced systems with units performing two complementary functions in two scenarios

3.5 Conclusions

This is the first research that investigates and generalizes the reliability estimation of k-out-of-n pairs:G Balanced systems. We introduce two types of k-out-of-n pairs:G Balanced systems with spatially distributed units: systems with all units performing the same function and systems with any two adjacent units performing two complementary functions. The reliability estimation is complicated due to (i) the enumeration of successful events must be carried out considering dynamic system state transitions and (ii) the probability of successful events must be calculated by considering the sequence of failures. We estimate the reliability of these systems under two scenarios. In the first scenario, unbalanced systems are considered as failed. In the second scenario, unbalanced systems are rebalanced by forcing down operating pairs into standby and/or resuming standby pairs to operation.

We propose a measure of symmetry, Moment Difference, to determine the balance of a k-out-of-n pairs:G Balanced system. We also propose algorithms for rebalancing unbalanced systems. Specifically, the algorithms determine the operating pairs to force down into standby or standby pairs to resume operation to rebalance unbalanced systems. The algorithms also determine the axes of symmetry for the rebalanced systems. The algorithms can be generalized to other systems with spatially distributed units.

The numerical examples show that the reliability of a k-out-of-n pairs:G Balanced system first increases and then decreases as more redundancy is added to the system in the scenario where unbalanced systems are considered as failed. In other words, there exists an optimal n for a given k in this scenario. The reason is as follows. On one hand, unbalanced system states are more likely to occur and hence the ratio of the number of successful events to the number of all possible events decreases as n increases, which tends to decrease system reliability. On the other hand, a system with more redundancy is able to survive a larger number of failed pairs, which tends to increase system reliability.

In addition, the examples show that a k-out-of-n pairs:G Balanced system tends to have higher reliability when the unbalanced system is rebalanced by forcing down operating pairs into standby and resuming standby pairs back to operation. By rebalancing the unbalanced system, we expect more successful events and fewer failure events to occur and hence higher system reliability.

The reliability estimation algorithms introduced in this chapter require extensive computational effort when systems are large. In Chapter 4, we investigate reliability approximation approaches for k-out-of-n pairs:G Balanced systems with large n which reduces the computation time and provides adequate accuracy. The current research is a benchmark for potential approximation approaches.

In addition, real-time reliability update based on degradation monitoring of individual units is also an interesting and challenging problem since system reliability in practice can be affected by uncertainty from manufacturing and operational environment of the individual units. We investigate the degradation analysis of such systems in Chapter 5.

CHAPTER 4

RELIABILITY APPROXIMATION OF *k*-OUT-OF-*n* PAIRS:G BALANCED SYSTEMS

4.1 Problem Definition and Assumptions

In Chapter 3, we estimate the reliability of a variety of k-out-of-n pairs:G Balanced systems. The reliability estimation for such systems is difficult to obtain due to the complexity of the problem: the operation of the systems depends on not only the number of operating pairs but also their spatial configuration. It is difficult, if not impossible, to estimate the reliability metrics accurately for systems with a large number of units since: it is time-consuming to determine the complete set of system states and successful events by enumeration and it is computationally intensive to obtain the probabilities of the successful events. Therefore, in this chapter we develop Monte Carlo simulation-based reliability approximation approach for the k-out-of-n pairs:G Balanced systems.

In addition, we investigate the reliability approximation in two scenarios: In the first scenario, we consider unbalanced systems as failed systems. In the second scenario, we rebalance any unbalanced system by considering standby.

The assumptions stated in the Introduction hold throughout this chapter. We also assume that all units in a system perform the same function. We assume the exponentially distributed lifetimes for individual units in order to make the problem more tractable. The

approximation approach we developed in this chapter increases computation efficiency by reducing the number of successful events to be enumerated. The probability calculation for any successful event with many failure occurrences is still complex in the second scenario when the lifetimes of individual units are not exponentially distributed.

4.2 Challenges in Reliability Estimation of Large Systems

4.2.1 *System Balance Determination*

Determination of system balance is essential for enumerating the successful events of any *k*-out-of-*n* pairs:G Balanced system. In Chapter 3, system balance is determined by using the concept of Moment Difference (MD). We can numerically validate that MD is effective when *n* is less than 30. However, it is not theoretically proved, and the numerical validation is time-consuming, if not impossible, when *n* is large. Therefore, we add another condition for balance determination in this chapter.

A unit is forced down whenever the opposite one in the same pair fails, and a pair of units are always forced down to standby or resume operation simultaneously. Therefore, the balance of the system is equivalent to the symmetry of operating pairs w.r.t. at least one pair of perpendicular axes. There are *n* pairs of perpendicular candidate axes of symmetry, which are either along a pair of units or in the middle of two pairs. The basic idea is to enumerate all unique pairs of perpendicular candidate axes and calculate the MD of the system w.r.t. to the axes.

For any candidate axis, denoted as a, we consider the units in the range of $\pm\pi/2$ on the two sides of the axis. We assign weights to the units. A unit has a weight value of one if it is operating and a weight value of zero otherwise. The angle from a unit to the axis is positive if it is clockwise and negative if it is anticlockwise. The weights of the units with positive (negative) angles compose a vector $\mathbf{w}_a^+$ ($\mathbf{w}_a^-$); and the corresponding angle vector is $\boldsymbol{\theta}_a^+$ ($\boldsymbol{\theta}_a^-$). The elements in the weight vectors and angle vectors are ordered according to the absolute values of the angles. The MD is calculated as

$$M_a = \mathbf{w}_a^+ \cdot \sin\boldsymbol{\theta}_a^+ + \mathbf{w}_a^- \cdot \sin\boldsymbol{\theta}_a^- \tag{4.1}$$

According to Section 3.2.2, the condition for a system to be balanced (symmetric) is that the system has at least one pair of perpendicular candidate axes that are both associated with zero MD. When the condition is not met, the system is unbalanced. The necessity of the condition is obvious whereas its sufficiency is not proved. In this chapter, we add another condition as follows. When we find a pair of perpendicular axes that are both associated with zero MD, we compare the vectors $\mathbf{w}_a^+$ and $\mathbf{w}_a^-$ for either axis in the perpendicular axis pair: When $\sum_{\substack{w_{a,i}^+ \in \mathbf{w}_a^+ \\ w_{a,i}^- \in \mathbf{w}_a^-}} \left| w_{a,i}^+ - w_{a,i}^- \right| = 0$, then the system is symmetric and balanced, otherwise the system is not balanced.

4.2.2 *k-out-of-n Pairs:G Balanced Systems Considering Unbalanced State as Failure*

In the first scenario where unbalanced systems are considered as failures, the probability calculation is straightforward based on Eq. (3.2) and Eq. (3.3) once the number of

successful events, η_h^n, is obtained by enumerating all possible successful events. However, enumerating all possible states for large systems is difficult and perhaps impossible. Therefore, in this chapter, we approximate the number of successful states η_h^n by Monte Carlo simulation.

4.2.3 *k-out-of-n Pairs:G Balanced Systems Considering Standby*

In the second scenario, unbalanced systems are rebalanced by forcing down operating pairs into standby and/or resuming standby pairs into operation. The iterative procedure to obtain the closed form expression of the general integral $I_h^w(h,t)$ is as described in Section 3.3.2. The procedure requires symbolic computation to obtain the limit of the equation, which becomes time-consuming when the number of failed pairs, h, becomes large.

In this scenario, the difficulty lies in both the enumeration of successful events and the probability calculation. It requires much more computation time to obtain the complete set of successful events than in the previous scenario since we have to rebalance the unbalanced system states by a heuristic, which takes additional computation time. The symbolic computation involved in calculating the probability of successful events also requires a considerable amount of additional computation time.

4.3 Reliability Approximation of Systems Considering Unbalanced State as Failure

In this section, we present a Monte Carlo simulation-based approximation for the reliability of k -out-of- n pairs:G Balanced system by assuming that unbalanced systems are considered as failures.

4.3.1 *Fundamental Method*

Suppose we do not impose balance requirements on the system except that we force down a unit permanently when the unit in the same pair fails, the number of successful events where h out of n pairs fail have $\left[n\cdot(n-1)\cdots(n-i+1)\cdots(n-h+1)\right]$ realizations since the first failure has n options, the second failure has $(n-1)$ options, the i^{th} failure has $(n-i+1)$ options and so forth. However, imposing balance requirements on the system results in a much smaller number of successful events since unbalanced systems are considered as failures.

Consider the state transition diagram for a 2-out-of-6 pairs:G Balanced system shown in Figure 3.3. We observe that the first failure has 6 options and the second failure has 5 options. Hence there are $\eta_2^6 = 30$ successful events with 2 out of 6 failed pairs. Note here we use η_h^n to denote the number of realizable state transition paths that lead to a balanced system with n pairs of units in total and h failed pairs. As shown in Figure 3.3 the third failure may result in an unbalanced system. Based on Eq. (3.5), $\eta_3^6 = 48$ instead of

$6\times5\times4=120$. So overall only $48/120=40\%$ of the options for the third failure contribute to the set of successful events.

We denote the proportion of options for the h^{th} failures that result in a balanced system as ξ_h^n where n represents that there are n pairs of units in total in the system. It is immediate that $\xi_h^n=1$ when $h=1, 2, (n-2), (n-1)$ and n because the h^{th} failure does not result in an unbalanced system when h has one of the five values. In addition, we obtain η_h^n as

$$\eta_h^n=\prod_{i=1}^{h}(n-i+1)\xi_i^n \tag{4.2}$$

The exact value of ξ_h^n cannot be obtained without enumerating all the possible successful events, though once ξ_h^n is obtained, η_h^n is immediate, and system reliability can be obtained easily based on Eq. (3.2) and Eq. (3.3).

4.3.2 *Monte Carlo Simulation-Based Algorithm*

4.3.2.1 *Algorithm Description*

We present a Monte Carlo simulation to obtain ξ_h^n. We enumerate state transition paths randomly instead of enumerating the complete set of state transition paths. By observing the enumerated state transition paths, which is a small portion of the complete set, we can obtain approximate values for ξ_h^n. By executing the simulation multiple times, we approximate the value of each ξ_h^n using the average of its approximate values obtained through all the simulation runs.

In each simulation run, we enumerate a portion of the system states after the h^{th} failures for h from 1 to $\min(n-k, n-3)$ in an iterative way. The reason why we only consider $h \le n-3$ is that when $h > n-3$, $\xi_h^n = 1$. Specifically, at the beginning of the j^{th} simulation, we let one of the pairs, say pair 1^*, to fail. We then randomly select a portion, $\psi_{2,j}^p$, of the $(n-1)$ remaining operating pairs, i.e. $\lceil \psi_{2,j}^p (n-1) \rceil$ remaining operating pairs, to fail, so that $\lceil \psi_{2,j}^p (n-1) \rceil$ system states with 2 failed pairs are generated. Note that $\lceil \bullet \rceil$ is the round-up operator. Then we randomly select a portion, $\psi_{3,j}^s$, of the generated system states with 2 failed pairs. For each of the selected system state, we randomly select a portion, $\psi_{3,j}^p$, of the $(n-2)$ remaining operating pairs to fail in order to generate some system states with 3 failed pairs. Generally, we randomly select a portion, $\psi_{h,j}^s$, of the generated system states with $(h-1)$ failed pairs. For each of the system state, if it is balanced, we randomly select a portion, $\psi_{h,j}^p$, of the $(n-h+1)$ remaining operating pairs to fail in order to generate some system states with h failed pairs. We continue this process for all $h \le \min(n-k, n-3)$. In each simulation run, we obtain an approximate value for ξ_h^n, particularly

$$\tilde{\xi}_{h,j}^n = \frac{N_{h,j}^B}{N_{h,j}} \tag{4.3}$$

where $N_{h,j}^B$ and $N_{h,j}$ are the number of balanced states and the total number of the generated states with h failed pairs in the j^{th} simulation run.

The approximate value for ξ_h^n is obtained by taking the average of the $\tilde{\xi}_{h,j}^n$ obtained through all the simulation runs. The flow chart of this approach is as shown in Figure 4.1 where $\left\{S_{h,j}^w\right\}$ is a set of generated states with h failures in the j^{th} simulation run, and we use the superscript w to index different states in the same set. The values of the parameters can be selected by the method introduced in Section 4.5.

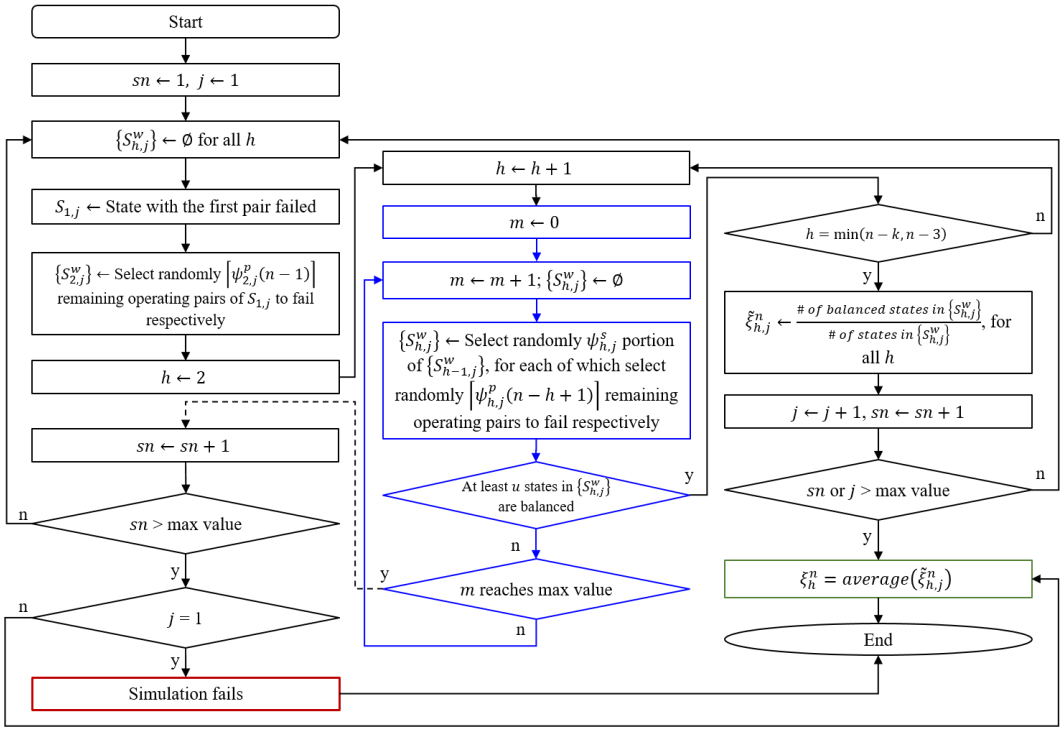


Figure 4.1 Flowchart of the Monte Carlo simulation for approximating ξ_h^n

Note that the simulation algorithm can fail when the sample size is too small: When not enough operating system states (balanced states in this case) are generated in the set $\left\{S_{h,j}^w\right\}$, its follow-up state set, $\left\{S_{h,j+1}^w\right\}$, may not contain any operating system states. Consequently, the simulation cannot continue. We use a loop indicated in blue in Figure 4.1 to generating

$\left\{S_{h,j}^{w}\right\}$ from $\left\{S_{h,j-1}^{w}\right\}$ for multiple times to prevent the lack of operating system in $\left\{S_{h,j}^{w}\right\}$. But the index of the loop, m, has a maximum value. Once m reaches the maximum value and there are still not enough operating system states in $\left\{S_{h,j}^{w}\right\}$. We will abandon the result obtained from the current simulation run and start a new simulation run. Once we exhaust the maximum number of simulation runs and all simulation runs are abandoned due to the small sample size and lack of operating system states, the simulation fails. Therefore, it is critical to choose the right sample size by choosing right values for $\psi_{h,j}^{s}$ and $\psi_{h,j}^{p}$, as explained in Section 4.5.

4.3.2.2 *Algorithm Complexity*

We compare the complexity of the proposed algorithm and the exact algorithm in Section 3.2.4.1. We only discuss the complexity of event enumeration by assuming k to be less than $n/2$. Note that smaller k results in more system states and state transitions. The number of fundamental instructions in the exact algorithm is maximally

$$\begin{aligned} & 1+\sum_{h=1}^{n-k-1}\frac{\binom{n}{h}}{n}\frac{\binom{n}{h+1}}{n} \\ & \le \frac{1}{n^2}\sum_{h=1}^{n-k-1}\frac{n-h}{h+1}\left(\prod_{j=0}^{h-1}\frac{n-j}{h-j}\right)^2 \\ & \le \arg\max_{h}\left[\frac{1}{n^2}\frac{n-h}{h+1}\left(\prod_{j=0}^{h-1}\frac{n-j}{h-j}\right)^2\right] \end{aligned} \tag{4.4}$$

If we denote the maximum value in Eq. (4.4) as A, we find that $\log(\log A) \approx \log n$ by numerical computation. Thus the computational complexity of the exact algorithm is approximately $O(e^n)$. The number of fundamental instructions in the proposed algorithm is maximally

$$\begin{aligned} & K \times sn_{max} \times m_{max} \times \sum_{h=1}^{n-k} \left(N^{med} \times (n-h+1) \right) \\ & \leq K \times sn_{max} \times m_{max} \times \sum_{h=1}^{n-k} \left(n^C \times n \right) \end{aligned} \tag{4.5}$$

where K is a constant less than 1 and C is a constant that is much less than $(n-k)$, e.g. $C=2$ when $N_1^{med}=1$, $N_2^{med}=(n-1)$ and $N_h^{med}=(n-1)(n-2)$ for $h \geq 3$. Since sn_{max} is related to n while m_{max} is not, the algorithm complexity is $O(n^{C+3})$.

4.4 Reliability Approximation of Systems Considering Standby

In this section, we present a Monte Carlo simulation-based approximation for the reliability of k-out-of-n pairs:G Balanced system by assuming that unbalanced systems to be balanced by forcing down operating pairs into standby and/or resuming standby pairs into operation. Also, we assume that the lifetimes of the individual units follow an exponential distribution with parameter λ.

4.4.1 *Fundamental Method*

4.4.1.1 *Successful Event Enumeration*

We first investigate the enumeration of successful events. Unlike the system in the first scenario where an unbalanced system is considered as a failure, here any two successful events may have different probabilities even if they have the same number of failed pairs due to the effect of standby. Therefore, we must obtain not only the number of successful events, as in the first scenario, but also necessary information on the successful events themselves in order to obtain the probability of a successful event. Specifically, for each successful event in the j^{th} simulation run, $E_{h,j}^{w}$, we must obtain the values of the corresponding coefficient vector for τ_i's, $\boldsymbol{\alpha}_{h,j}^{w}$, the number of operating pairs, $g_{h,j}^{w}$, and the number of failed pairs, h, to obtain the value of Eq. (3.9). In order to simplify the calculation further, we categorize the events by h, $g_{h,j}^{w}$, and $\boldsymbol{\alpha}_{h,j}^{w}$. Thus the probabilities of the events in each category are the same.

Using Monte Carlo simulation, we generate state transition paths by randomly selecting operating pairs to fail one by one. By observing and synthesizing the results of multiple runs of simulation, we obtain the total number of events with h failed pairs, η_h^n, categorize the events based on corresponding $g_{h,j}^{w}$ and $\boldsymbol{\alpha}_{h,j}^{w}$, and obtain the proportion of events in each category out of all the events with h failed pairs, $\rho_{h,c}^{n}$, where $h=1$ to $(n-k)$. The number of events with h failed pairs in category c is $\left(\eta_h^n \cdot \rho_{h,c}^{n}\right)$.

4.4.1.2 *Probability Calculation*

The probability calculation for successful events is presented in Section 3.3.2, where the major challenge is to obtain the value for the integral in Eq. (3.10) We develop an iteration rule to obtain its closed form expression. Numerical values can then be obtained by applying the values of $\boldsymbol{\alpha}_h^w = \left(\alpha_{h,1}^w, \alpha_{h,2}^w, \cdots, \alpha_{h,h}^w\right)$, h, and g_h^w. However, we must carry out symbolic computation to obtain the limit of the closed form expression when the denominators in the closed form expression have zero values, which is caused by zero-valued combinations of $\alpha_{h,i}^w$ ($i = 1, \cdots, h$). Thus, the method of calculating the probabilities of successful events presented in Section 3.3.2 can be computationally expensive.

In order to obtain an approximation of Eq. (3.10), we perform the following: First, we replace the zero-valued $\alpha_{h,i}^w$ with a small number ε. The value of ε can be set around $\lambda / 10^3$ based on experience. When there is zero-valued sum of $\alpha_{h,i}^w$ or $-\alpha_{h,i}^w$ in the dominators, we replace $\alpha_{h,i}^w$ with $\alpha_{h,i}^w \cdot \left(1 + \varepsilon \cdot \sigma_{h,i}^w\right)$ where $\sigma_{h,i}^w$ follows a uniform distribution, $U(0,1)$.

4.4.2 *Monte Carlo Simulation-Based Algorithm*

4.4.2.1 *Algorithm Description*

The simulation procedure is similar to the procedure introduced in Section 4.3. Figure 4.2 shows the flowchart of the simulation procedure. The values of the parameters can be selected by the method introduced in Section 4.5.

The basic idea is as follows. Starting from a state with one failed pair, we randomly generate its follow-up states and the further follow-up states until we exhaust all the possible numbers of failed pairs. A follow-up state of a system state can be obtained by turning one of the operating pairs in the current system into failure and balancing the system if the additional failure results in an unbalanced system.

Specifically, in the j^{th} simulation run, starting with a system state with one failed pair, denoted as $S_{1,j}$, we randomly select a portion of its operating pairs to generate an incomplete set of follow-up states, $\left\{S_{2,j}^{w}\right\}$, and the corresponding set of events with 2 failed pairs, $\left\{E_{2,j}^{w}\right\}$. Then we randomly select a portion of the generated states, $\left\{S_{2,j}^{w}\right\}$, for each of which we randomly select a portion of operating pairs to fail to obtain its follow-up states and corresponding events. Thus we obtain $\left\{S_{3,j}^{w}\right\}$ and $\left\{E_{3,j}^{w}\right\}$. We continue this iterative procedure until we exhaust all the possible values for the number of failed pairs, h, and obtain corresponding sets $\left\{S_{h,j}^{w}\right\}$ and $\left\{E_{h,j}^{w}\right\}$ where $1 \le h \le n-k$.

The events can be categorized into different categories according to the values of h, $g_{h,j}^{w}$ and $\boldsymbol{\alpha}_{h,j}^{w}$. For example, in Figure 3.7 the transition paths from state 1 to states 6, 7, 8, and 9 are all successful events with three failed pairs, namely $h=3$ for all these events. According to the number of operating pairs g_{3}^{w}, and the coefficient vector, $\boldsymbol{\alpha}_{3}^{w}$, before failure times, the state transition paths reaching states 6 and 9 are considered in one category, whereas the paths reaching states 7 and 8 are considered in a different category.

Specifically, states 6 and 9 have three operating pairs, and no standby is required to rebalance the states. States 7 and 8 have two operating pairs, and one operating pair is forced down into standby when the third failure occurs.

We denote the proportion of events with h failed pairs in category c as $\rho_{h,c}^{n}$. Consider the state transition diagram in Figure 3.7 again and suppose that the successful events reaching states 6 and 9 are in category 1 and those reaching states 7 and 8 are in category 2. There are 120 possible transition paths from state 1 to states 6 to 9. Among these paths, 48 of them reach states 6 or 9 while the other 72 reach states 7 and 8. Therefore, $\rho_{3,1}^{6} = 48/120 = 0.4$ and $\rho_{3,2}^{6} = 72/120 = 0.6$.

In addition, we determine the category of event $E_{h,j}^{w}$ according to the category of event $E_{h-1,j}^{v}$ when event $E_{h,j}^{w}$ is derived from event $E_{h-1,j}^{v}$.

For each enumerated event $E_{h,j}^{w}$, we record (i) the number of operating pairs in its corresponding state $S_{h,j}^{w}$, i.e. $g_{h,j}^{w}$; (ii) the vector of the coefficient in Eq. (3.10), $\boldsymbol{\alpha}_{h,j}^{w}$; and (c) the category of the event according to $g_{h,j}^{w}$ and $\boldsymbol{\alpha}_{h,j}^{w}$, denoted as $c_{h,j}^{w}$.

Similarly, we obtain η_{h}^{n}, the total number of events with h failed pairs using Eq. (4.2) by replacing ξ_{h}^{n} by κ_{h}^{n}. The meaning of κ_{h}^{n} is the average ratio between the number of follow-up states that can be generated from a state $S_{h-1,j}^{w}$ and $\left(n-\left(h-1\right)\right)$ which is the

greatest number of follow-up states that can be generated from a $S_{h-1,j}^{w}$. Note that $\kappa_h^n = 1$ when $h = 1, 2, 3, (n-1)$ and n. Given $\left(n-\left(h-1\right)\right) \geq k$ a state $S_{h-1,j}^{w}$ can generate at most $\left(n-\left(h-1\right)\right)$ follow-up states when it is balanced without standby, and less than $\left(n-\left(h-1\right)\right)$ follow-up states when it is rebalanced by forcing down operating pairs into standby. In the latter case, if state $S_{h-1,j}^{w}$ is a failed state, i.e. $S_{h-1,j}^{w}$ has less than k operating pairs after being rebalanced, then it cannot generate any follow-up states. In each simulation run, we obtain an approximate value for κ_h^n, particularly

$$\tilde{\kappa}_{h,j}^{n} = \frac{\sum_{w=1}^{\left\|\left\{S_{h-1,j}^{w}\right\}\right\|} g_{h-1,j}^{w} I\left(g_{h-1,j}^{w} \geq k\right)}{\sum_{w=1}^{\left\|\left\{S_{h-1,j}^{w}\right\}\right\|} \left(n-(h-1)\right)} \tag{4.6}$$

where $\left\|\left\{S_{h-1,j}^{w}\right\}\right\|$ is the size of the set $\left\{S_{h-1,j}^{w}\right\}$; $I\left(g_{h-1,j}^{w} \geq k\right) = 1$ if $g_{h-1,j}^{w} \geq k$, and 0 otherwise. The approximate value for κ_h^n is obtained by taking the average of the $\tilde{\kappa}_{h,j}^{n}$ obtained through all the simulation runs.

In order to obtain the proportion of events with h failed pairs in category c, $\rho_{h,c}^{n}$, we synthesize the categorization of events $\left\{E_{h,j}^{w}\right\}$ from different simulation runs. In each simulation run, the events $\left\{E_{h,j}^{w}\right\}$ are categorized independently without considering the other simulation runs. The categorizations in all simulation runs are not synthesized. For example, the first category in simulation run j_1 could be the second category in simulation run j_2 or might not exist in simulation run j_2.

In each simulation run, we categorize $\left\{E_{h,j}^{w}\right\}$ into different groups. Suppose each event $E_{h,j}^{w}$ is assigned to a category $c_{h,j}^{w}$. Denote the number of unique values of $c_{h,j}^{w}$ (the number of unique categories) as $\left\langle\left\{c_{h,j}^{w}\right\}\right\rangle$. We summarize the category information as follows. Let notations with a bar represent the category features for a specific simulation run. For each unique category $c \in \left\{1,\cdots,\left\langle\left\{c_{h,j}^{w}\right\}\right\rangle\right\}$, we first count the total number of events in this category, denoted as $\bar{o}_{h,c,j}$; then we record the common features of events in this category, i.e. the vector $\bar{\boldsymbol{\alpha}}_{h,c,j}$ and the number of operating pairs $\bar{g}_{h,c,j}$.

To synthesize the event categorizations from different simulation runs, we first set the categorization obtained in the first simulation run as the synthesized categorization. We set the corresponding category features as the synthesized category features, i.e. $\hat{\boldsymbol{\alpha}}_{h,c}^{n} = \bar{\boldsymbol{\alpha}}_{h,c,1}$, $\hat{g}_{h,c}^{n} = \bar{g}_{h,c,1}$, and $\hat{o}_{h,c}^{n} = \bar{o}_{h,c,1}$ where notations with a hat represent the synthesized category features. Then for each h ($h = 1,\cdots,n-k$), we match the categories obtained in the second simulation to the synthesized categories by comparing $\bar{\boldsymbol{\alpha}}_{h,c,2}$ and $\bar{g}_{h,c,2}$ with $\hat{\boldsymbol{\alpha}}_{h,c'}^{n}$ and $\hat{g}_{h,c'}^{n}$. If category c in the second simulation can be matched to the synthesized category c', then we add the corresponding $\bar{o}_{h,c,2}$ to the $\hat{o}_{h,c'}^{n}$. If a category in the second simulation cannot be matched to any of the synthesized categories, we add a new category and its corresponding category features to the synthesized categorization. Then we carry out the matching procedure for all the other simulation runs iteratively to obtain the comprehensive synthesized categorization. Note that after we complete the matching procedure, we can

also reorganize the $\bar{o}_{h,c,j}$ obtained in each simulation run. This way, the intermediate results for $\rho_{h,c}^{n}$, denoted as $\tilde{\rho}_{h,c,j}^{n}$ can also be obtained.

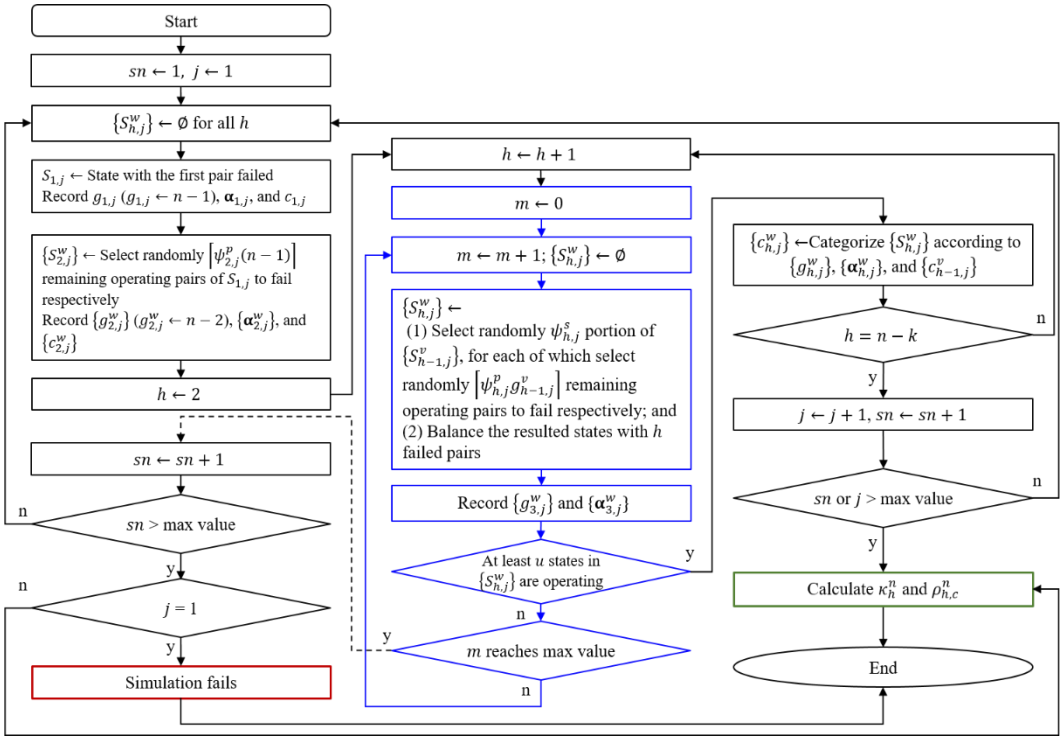


Figure 4.2 Flowchart of the Monte Carlo simulation for approximating κ_{h}^{n} and ρ_{h}^{n}

When the synthesized categorization is obtained, we approximate the value of $\rho_{h,c}^{n}$ as

$$\rho_{h,c}^{n} = \frac{\hat{o}_{h,c}^{n}}{\sum_{c'=1}^{\varsigma_{h}^{n}} \hat{o}_{h,c'}^{n}} \tag{4.7}$$

where ς_{h}^{n} is the number of categories of events with h failed pairs out of the n pairs. System reliability can be obtained as

$$R_{sys}(t) = \left[R(t)\right]^{n} + \sum_{h=1}^{n-k} \eta_{h}^{n} \sum_{c=1}^{\varsigma_{h}^{n}} I\left(\hat{g}_{h,c}^{n} \geq k\right) P_{h,c}^{n}(t)\, \rho_{h,c}^{n} \tag{4.8}$$

where $\hat{g}_{h,c}^{n}$ is the number of operating pairs remaining by the end of the events with h failed pairs in category c; $P_{h,c}^{n}(t)$ is the probability of an event with h failed pairs in category c.

4.4.2.2 *Algorithm Complexity*

We compare the complexity of the proposed algorithm and the exact algorithm in Section 3.3.4.2. We only discuss the complexity of event enumeration. The complexity of the proposed method is $O\left(n^{C+3}\right)$ as in Eq. (4.5). The computational complexity of the exact method can be obtained as follows. The number of fundamental instructions in the exact algorithm is as in Eq. (4.9). Thus the algorithm complexity is $O\left(n^{n-(k+1)}\right)$.

$$1+\sum_{h=1}^{n-(k+1)}\left(\prod_{j=1}^{h}(n-h)\right) \tag{4.9}$$

4.4.2.3 *Comparison of Two Algorithms*

We propose two algorithms for the reliability approximation of k-out-of-n pairs:G Balanced systems in two scenarios: In the first scenario unbalanced systems are considered as failed and in the second scenario unbalanced systems are rebalanced by using standby. We now compare the two algorithms.

Similarities

- First, both algorithms approximate system reliability by approximating the parameters which describe the complete set of successful events. The parameters include ξ_h^n in the first algorithm and κ_h^n and $\rho_{h,c}^n$ in the second one.
- Second, the method of randomly selecting system states and generating follow-up states is the same.
- Third, the method of choosing the parameters is the same as introduced in Section 4.5.
- Fourth, the method of preventing simulation failure is the same.

Differences

- First, the parameters that describe the complete set of successful events are different in the two algorithms. Therefore, the information we obtain from the simulation is different.
- Second, we need to rebalance the unbalanced system in the second algorithm.

4.5 Parameter Estimation Algorithm

In the proposed algorithms, we have five parameters to set: $\psi_{h,j}^s$, $\psi_{h,j}^p$, u, and the maximum values for j, sn, and m, i.e. j_{max}, sn_{max}, and m_{max}. We now discuss the procedure for determining the values of the parameters.

Parameter $\psi_{h,j}^s$ is the proportion of states to select from $\left\{S_{h-1,j}^w\right\}$ to generate follow-up states $\left\{S_{h,j}^w\right\}$. For each state in $\left\{S_{h-1,j}^w\right\}$ we select $\psi_{h,j}^p$ portion of its operating pairs to fail

to generate its follow-up states. We generate an adequate number of states to obtain at least u balanced states in $\left\{S_{h,j}^{w}\right\}$ to avoid additional loops shown in blue in Figure 4.1 and Figure 4.2.

We develop a rule to assign the values for $\psi_{h,j}^{s}$ and $\psi_{h,j}^{p}$. First, we calculate reasonable values for the number of states we want to generate in $\left\{S_{h,j}^{w}\right\}$, which we denote as N_h^{med}. For instance, we always let $N_1^{med}=1$, and we let $N_2^{med}=(n-1)$, $N_h^{med}=(n-1)(n-2)$ for $h\geq 3$. Second, we determine a reasonable sampling ratio for $\psi_{h,j}^{p}$, which we denote as ψ_{h0}^{p}, e.g. we let $\psi_{h0}^{p}=0.1$. Then we determine the values by following the flowchart as shown in Figure 4.3 where $N_{h-1,j}^{c}$ is the number of generated successful states with $(h-1)$ failed pairs in the j^{th} simulation run; $N_h^{min}=0.5N_h^{med}$; $N_h^{max}=1.5N_h^{med}$; and

$$B_{h,j}^{s}=\frac{N_h^{med}}{N_{h-1,j}^{c}(n-h+1)\psi_{h,j}^{p}} \tag{4.10}$$

and

$$B_{h,j}^{p}=\frac{N_h^{med}}{N_{h-1,j}^{c}(n-h+1)\psi_{h,j}^{s}} \tag{4.11}$$

The basic idea of the algorithm is that 1) when the number of successful states in $\left\{S_{h-1,j}^{w}\right\}$ is small, i.e. $N_{h-1,j}^{c}<N_h^{min}$, we select all the states in $\left\{S_{h-1,j}^{w}\right\}$, i.e. set $\psi_{h,j}^{s}$ to 1, to increase the number of states in $\left\{S_{h,j}^{w}\right\}$ up to N_h^{med}; 2) when $N_{h-1,j}^{c}>N_h^{max}$, we use the preset value $\psi_{h,j}^{p}=\psi_{h0}^{p}$ and make $\psi_{h,j}^{s}$ small enough to reduce the number of states in $\left\{S_{h,j}^{w}\right\}$ down to

N_h^{med}; when $N_h^{min} \le N_{h-1,j}^c \le N_h^{max}$, we assign proper values to $\psi_{h,j}^s$ and $\psi_{h,j}^p$ to maintain the number of states in $\left\{S_{h,j}^w\right\}$ around N_h^{med}.

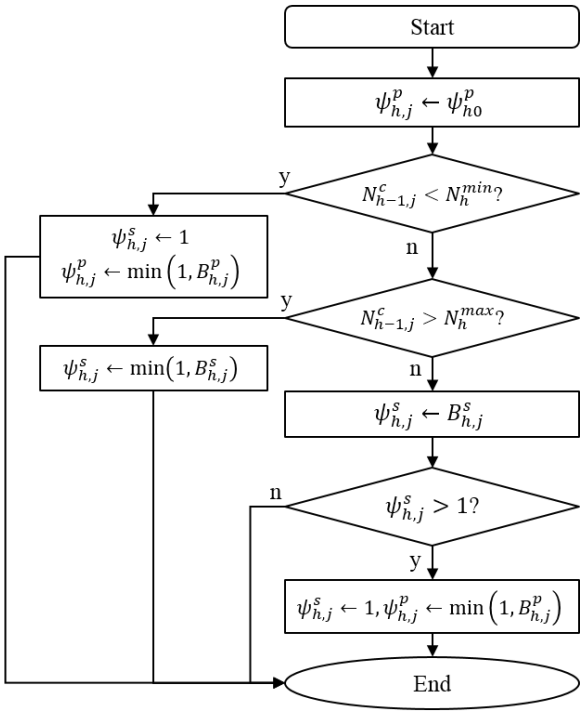


Figure 4.3 Rules of assigning the values for $\psi_{h,j}^s$ and $\psi_{h,j}^p$

The value of u can be obtained by trial. In fact, once we assign proper values for $\psi_{h,j}^s$ and $\psi_{h,j}^p$, it is likely to generate enough balanced states at each state, then we can simply let $u = 1$. For the same reason the maximum value for m, i.e. m_{max} can be selected arbitrarily in most of the cases.

The maximum value for j, i.e. j_{max}, depends on the value of n. A greater n results in a greater j_{max}. Here we let $j_{max} = n$ to $2n$. The maximum value for sn, i.e. sn_{max}, is an arbitrary number that is much greater than j_{max}.

4.6 Numerical Examples

In this section, we present two numerical examples to show the effectiveness and efficiency of the approximation approach proposed in this paper. The second scenario where unbalanced systems are rebalanced by standby pairs is a more general case. Therefore, we give the numerical examples based on the second scenario. In the numerical examples, we let $N_1^{med} = 1$, and $N_2^{med} = (n-1)$, and $N_h^{med} = (n-1)(n-2)$ for $h \geq 3$ to determine the number of states to generate, i.e. the sample size. The algorithm for determining sample size and the notation, N_h^{med}, are explained in the Section 4.5. Note that this setting for N_h^{med} means that we generate all the possible system states with less than or equal to three failed pairs. This is because that the total numbers of system states with one, two, and three failed pairs are 1, $(n-1)$, and $(n-1)(n-2)$, respectively, when we always choose pair 1^* to be the first pair to fail as indicated in both Figure 4.1 and Figure 4.2. Note all the pairs in the system are identical and can fail first with the same probability. Therefore, choosing pair 1^* to fail first does not affect the simulation results.

4.6.1 *Numerical Example 1: Approximation Accuracy*

Consider a 6-out-of-12 pairs:G Balanced system under the assumption that unbalanced system is rebalanced by standby pairs. The lifetime of individual units follows an exponential distribution with mean 40. We implement ten simulation runs to obtain the results. The approximate values for κ_h^{12} where $h=4$ to 6 are plotted in the first subplot in Figure 4.4 where the black horizontal lines indicate the 0.95 confidence intervals of the simulated results. As shown, the approximate values are very close to the exact values, which validates the efficiency of the proposed approximation approach. In addition, we show the values of $\tilde{\kappa}_{h,j}^{12}$ in the second subplot in Figure 4.4 where different simulations are represented by different colors.

As shown in Figure 4.4, different simulations results for different values of $\tilde{\kappa}_{h,j}^{12}$. The randomness is overall derived from two sources. First, the system states in each simulation run are randomly generated, which is only a portion of the complete set of system states. Therefore, the values of $\tilde{\kappa}_{h,j}^{12}$ calculated based on these states are random. Second, the number of system states generated in any simulation run is random. The number of system states generated is dynamically determined by the algorithm described in the Appendix. The number of generated system states with h failed pairs depends on the number of generated operating system states with $(h-1)$ failed pairs. Since the states are randomly generated, the number of operating system states varies. Therefore, the number of states to be generated also shows randomness.

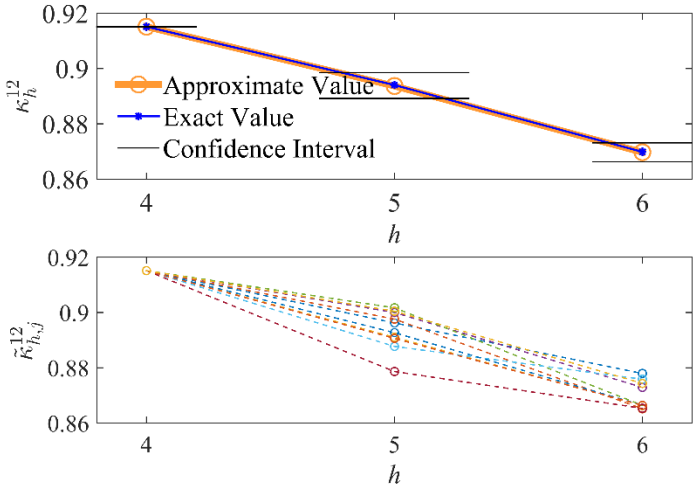


Figure 4.4 Simulation results: the values of κ_h^{12} and $\tilde{\kappa}_{h,j}^{12}$

From Figure 4.4, we observe that the approximate value of $\tilde{\kappa}_{4,j}^{12}$ does not show any randomness and the confidence interval of the approximate value of κ_4^{12} has overlapping upper and lower bounds. Based on Eq. (4.6), the value of $\tilde{\kappa}_{4,j}^{12}$ is determined by the system states with three failed pairs. In this numerical example, we choose to generate all possible system states with three failed pairs as explained at the beginning of Section 4.5. Therefore, we observe no randomness in the value of $\tilde{\kappa}_{4,j}^{12}$.

From Figure 4.4, we also observe that the value of $\tilde{\kappa}_{5,j}^{12}$ shows more discrepancy than that of $\tilde{\kappa}_{6,j}^{12}$. Generally, based on the algorithm described in the Appendix, we maintain the number of generated system states around a target value, N_h^{med}. On the other hand, the total number of system states with h failed pair increases with h. We expect the discrepancy in the value of $\tilde{\kappa}_{h,j}^{12}$ to increase with h. However, for a system with an even number of

pairs, unbalance is more likely to occur when h is odd than when it is even, which creates additional randomness. This explains the greater discrepancy of $\tilde{\kappa}^{12}_{5,j}$ than $\tilde{\kappa}^{12}_{6,j}$. We show the simulation results of $\tilde{\kappa}^{11}_{h,j}$ and $\tilde{\kappa}^{12}_{h,j}$ when $k=1$ in Figure 4.5. We observe increasing discrepancy as h increases both when $n=11$ (odd) and when $n=12$ (even). In addition, we observe greater discrepancy at odd-valued h only when $n=12$ (even).

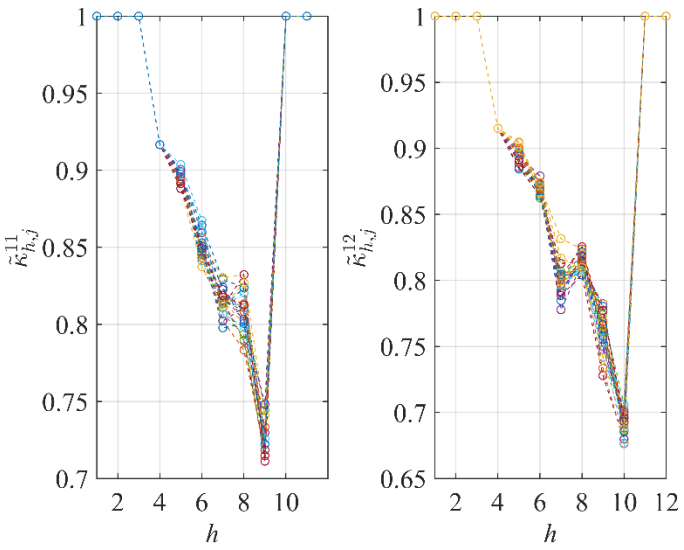


Figure 4.5 Discrepancy in $\tilde{\kappa}^{n}_{h,j}$ when $n=11$ (odd) and 12 (even)

The approximate value for $\rho^{12}_{h,c}$ where $h=3$ to 6 is plotted in Figure 4.6. In Figure 4.6, the approximate values are represented by circles, and the exact values are represented by asterisks. Similarly, we indicate the 0.95 confidence intervals of the approximate values. Since there does not exist randomness when the number of failures $h=3$ as explained previously, the corresponding upper and lower bounds of the confidence interval in Figure

4.6 overlap. We also present the intermediate simulation results for $h=3$ to 6 in Figure 4.7 where different colors represent different simulation runs.

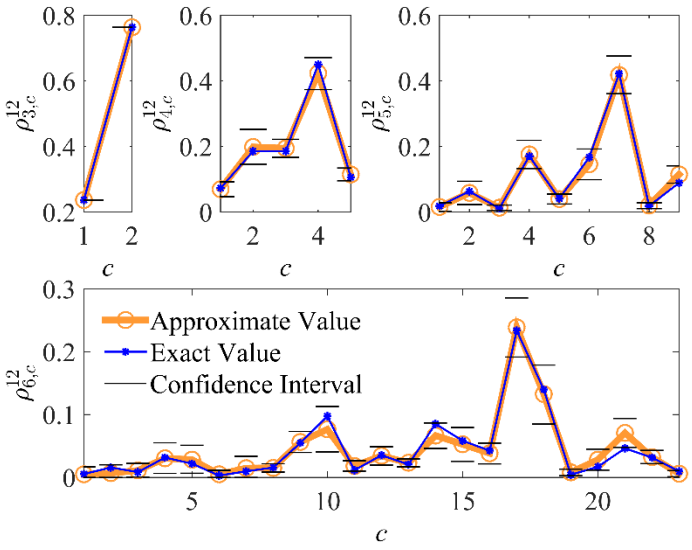


Figure 4.6 Simulation results: values of $\rho_{h,c}^{12}$

We then estimate the system reliability, as shown in Figure 4.8. The difference between the exact value and approximate value is negligible. The mean absolute error of approximation is 4.4107×10^{-4}, and the maximum absolute error is 0.0030, which demonstrates the effectiveness of the approximation approach. The exact values are obtained by using the method introduced in Chapter 3 in a computational time of 41 seconds compared with the 4 seconds used for the Monte Carlo simulation approximation. This shows our approximation approach is also efficient.

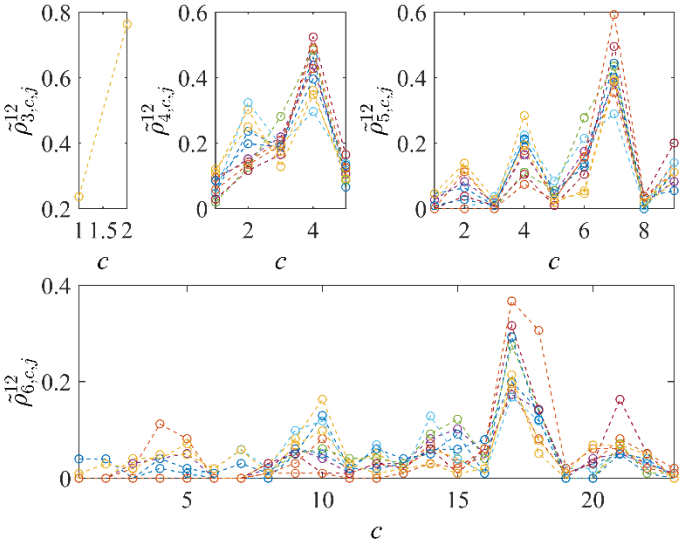


Figure 4.7 Simulation results: values of $\tilde{\rho}^{12}_{h,c,j}$

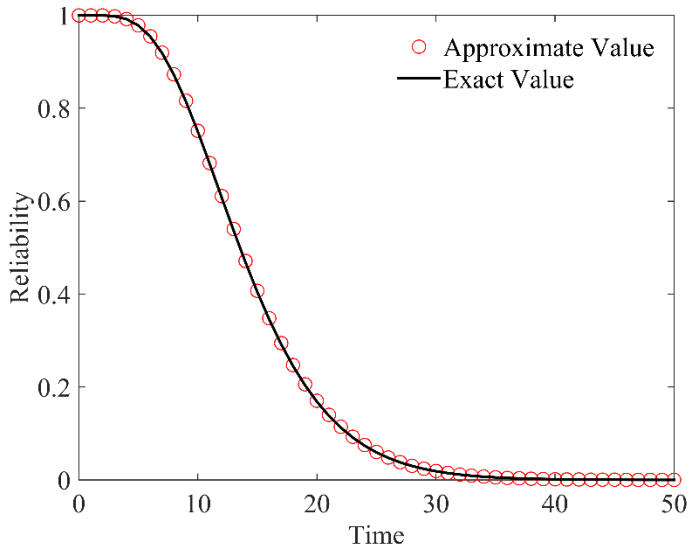


Figure 4.8 Simulation results: reliability approximation

We have shown the accuracy of the proposed method when n is relatively small. Due to the large computation time required by the exact estimation method, it is very difficult to obtain the exact value of reliability for large systems. Thus, for large systems, we calculate

the sum of the probabilities of all the possible events, namely, we assume $k=0$. Note that the exact reliability value for any 0-out-of-n pairs:G Balanced system at any time should be 1. We calculate the absolute error of the approximate reliability value at each time for $n=15$ to 30. We found that the mean of the absolute errors is 9.56×10^{-3} and the maximum absolute error is 0.027, which is negligible.

We also investigate the effect of sample size (the number of system states to be generated) and the number of simulation runs on the accuracy and efficiency of the approximation algorithm. We let the target sample size, N_h^{med}, be 1 and $(n-1)$, respectively, when failure number $h=1$ and 2, and let it be constant for the other failure numbers. We calculate the approximate system reliability and its absolute error when the simulation number, sn, is 5 to 25 with a step 5 and the target sample size N_h^{med} ($h>2$) be 30, 70 and 110. We plot the absolute error of the approximate reliability and the computation time used by the algorithm in Figure 4.9 and Figure 4.10.

In Figure 4.9 we present the boxplots of the absolute error of approximate reliability. Each subplot corresponds to a different target sample size. From Figure 4.9 we observe that the absolute error tends to decrease when simulation number and sample size increase.

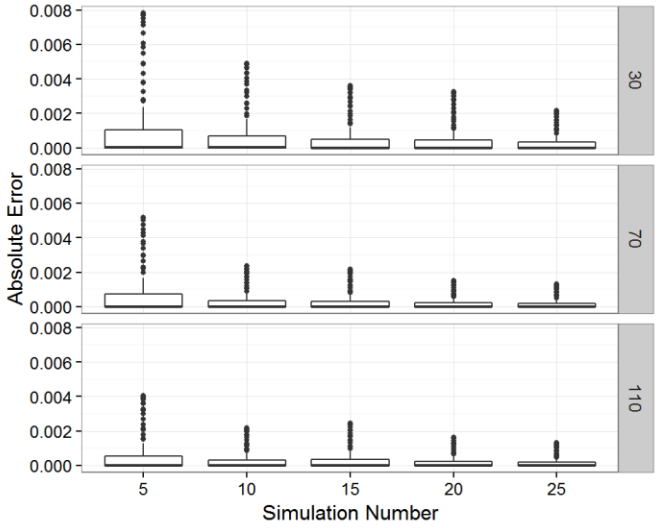


Figure 4.9 The effect of the number of simulations and sample size on the algorithm accuracy

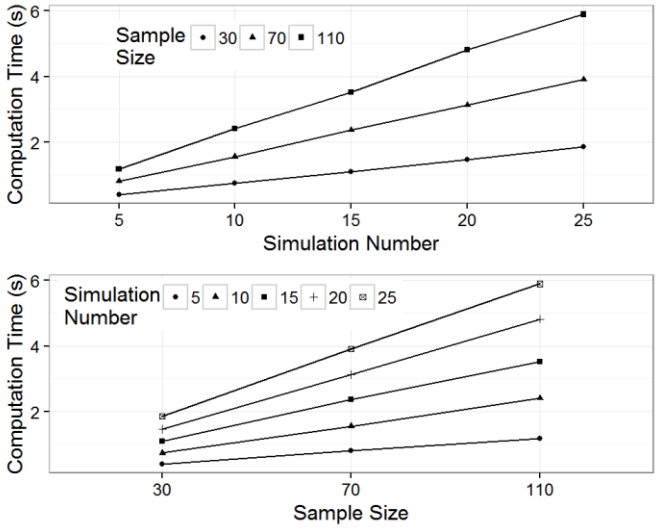


Figure 4.10 The effect of the number of simulations and sample size on the algorithm efficiency

In Figure 4.10 we plot the computation time against simulation number and sample size in two subplots. It is clear that computation time increases linearly with simulation number and sample size.

4.6.2 *Numerical Example 2: Approximation Efficiency*

In addition, we show the computation efficiency of the proposed reliability approximation method by comparing its computation time with that of the exact reliability method introduced in Chapter 3. We estimate the reliability for several $(n-5)$-out-of-n pairs:G Balanced systems for $n=6$ to 30 by assuming unbalanced systems can be rebalanced by standby. Again, we assume that the lifetimes of individual units follow an exponential distribution with mean 40. We plot the computation times by the two methods in Figure 4.11 we plot the computation time for $(n-5)$-out-of-n pairs:G Balanced systems when $n \in \{40, 50, \cdots, 100\}$. We also indicate the horizontal lines of four hours and eight hours. This shows that the computation time of the approximate algorithm increases exponentially.

In fact, when n increases and k decreases, the number of successful events to enumerate increases exponentially and the maximum dimension of the integral in Eq. (3.9) increases. The latter is due to the greater number of failures. Consequently, the performance of the algorithm is affected in two aspects: First, computation time increases exponentially. Second, the accuracy of the algorithm degrades since approximation error accumulates when we sum the probabilities of successful events. When the dimension of the integral in Eq. (3.9) increases, more zero-valued terms will be involved in its calculation. So the

probability is approximated with greater error. At the same time, we have more events to include into system reliability. Consequently the approximation error for system reliability increases.

We find that the application of the approximation method is limited when $n > 30$ based on the results of additional numerical experiments. Overall, when $30 < n \leq 50$, the value of k should be within [$n-15$, n], when $50 < n \leq 60$, the value of k should be within [$n-10$, n], and when $n > 60$, the value of k should be within [$n-5$, n]. When the value of k is out of the recommended range, extensive computation time and low accuracy is expected.

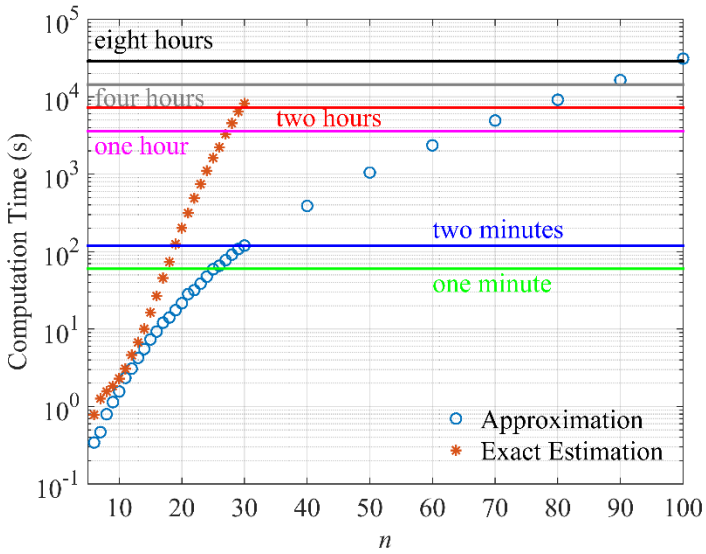


Figure 4.11 Computation time of approximation method and exact estimation method

4.7 Conclusions

In this chapter, we investigate the reliability approximation approach for k-out-of-n pairs:G Balanced systems in two scenarios: 1) unbalanced systems are considered as

failures; and 2) unbalanced systems are rebalanced by forcing down operating pairs into standby or resuming standby pairs into operation. The major difficulty in obtaining the exact reliability of such systems is the extensive computation time caused by (i) enumerating the complete set of successful events, which are sequences of ordered failures; and (ii) calculating the probabilities of successful events (only in the second scenario).

We proposed Monte Carlo simulation-based approximation approaches for the two scenarios. The basic idea of the approaches is generating events sequentially by randomly selecting a portion of operating pairs to fail. The approximation of the key parameters for estimating system reliability is obtained by observing the system states and events, including failure events and successful events, generated in all simulation runs.

The numerical examples validate the effectiveness and efficiency of the approximation approaches by showing that the approximate system reliability values are obtained with high accuracy in a much shorter computation time compared with the exact method.

CHAPTER 5

RELIABILITY ESTIMATION BASED ON DEGRADATION MODELING OF SPATIALLY DISTRIBUTED UNITS

5.1 Problem Definition and Assumptions

Many systems are composed of spatially distributed units which are subject to different operating conditions. The reliability of such systems depends not only on the reliability of individual units but also on their configurations. In this chapter, we develop a degradation model for systems where units are spatially distributed and balanced. More specifically, we consider *k*-out-of-*n* pairs:G Balanced systems. The effect of operating conditions on the units is considered and the corresponding reliability estimate is obtained. The degradation path of every unit is modeled based on collected observations of the degradation indicators and its physics or statistics degradation rate. We investigate the effect of the system configuration on the overall system reliability. We also estimate the *pdf* of time to a specified failure.

We first estimate the reliability metrics of a 1-out-of-6 pairs:G Balanced system, as shown in Figure 1.1(a), at the initial operation stage when no failures occur. We then provide a procedure for estimating the reliability metrics of any k -out-of- n pairs:G Balanced systems at any stage when h $\left(0 \le h \le n-k\right)$ failures are observed.

The assumptions stated in the Introduction hold throughout this chapter except that the lifetimes of individual units in this chapter are not assumed to be *i.i.d.*. The distribution of any unit's lifetime is determined by its degradation process.

5.2 System Description

In this chapter, we first estimate reliability metrics for a 1-out-of-6 pairs:G Balanced system shown in Figure 1.1(a) with all units performing the same function. The system is considered balanced when the operating units are symmetric w.r.t. at least a pair of perpendicular axes. Unbalanced systems are rebalanced by forcing down additional operating pairs into a standby state. Standby pairs resume operation when their resumption can bring the system back to balance or bring additional operating pairs to the system.

The state transition diagram of the system, which describes all of the possible states that may occur and all of the state transition paths that may lead to these states, is then obtained as shown in Figure 5.1. Note that each system state shown in Figure 5.1 is a typical state of all the states that have the same relative locations of failed pairs. For instance, system state 3 is a typical state of all the states that have two failed adjacent pairs. A system state itself contains no information on the sequence of failures. The sequence of failures is found by observing the transitions from one state to another. A successful event is a state transition path that leads to an operating system state. Here, an operating system state should have at least k pairs of operating units that are symmetric w.r.t. at least a pair of perpendicular axes.

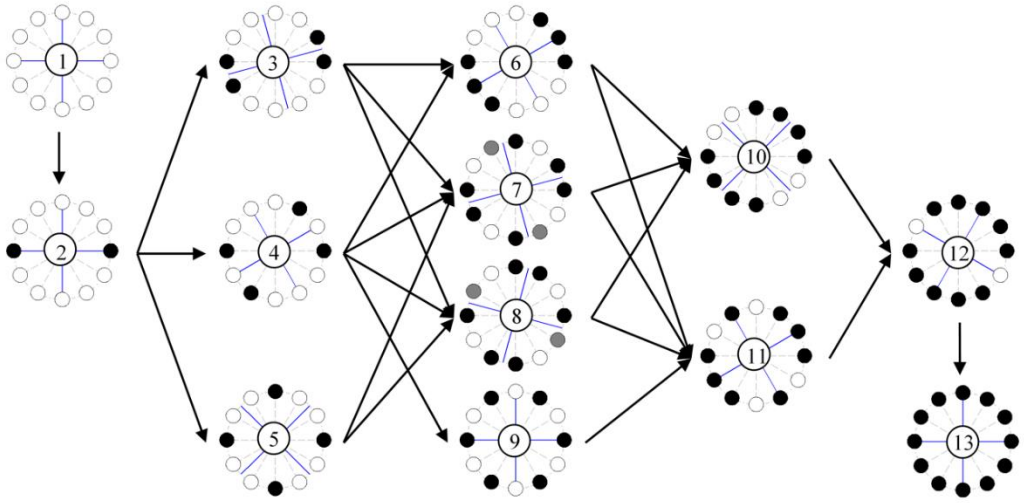


Figure 5.1 State transition diagram for 1-out-of-6 pairs:G Balanced system

5.3 Degradation Model for Individual Units

In a system with spatially distributed units, the degradation paths of individual units can be significantly different due to the operating conditions at the spatial locations. In this section, we introduce a degradation model for individual units considering different operating conditions. We also investigate the effect of the standby state on the degradation of individual units.

5.3.1 *Baseline Degradation Rate Model*

We consider the following degradation model [86]:

$$dD_i(\tau) = \mu_i(\tau)d\tau + \sigma_i\sqrt{\mu_i(\tau)}dB(\tau) \tag{5.1}$$

where $D_i(\tau)$ is the amount of degradation of unit i at time τ; $dD_i(\tau)$ is the derivative of $D_i(\tau)$ at τ; $\mu_i(\tau) = \mu_0(\tau;\boldsymbol{\Omega})A(\mathbf{S}_i(\tau);\boldsymbol{\Theta})$ is the degradation rate of unit i at τ;

$A(\mathbf{S}_i(\tau);\boldsymbol{\Theta})$ is the acceleration factor due to the operating condition $\mathbf{S}_i(\tau)=\{S_{il}(\tau);l=1,2,\ldots,o\}$ which is a vector of stress factors affecting unit i at τ; o is the number of stress factors considered in the model; $\mu_0(\tau;\boldsymbol{\Omega})$ is the baseline degradation rate of individual units of the same type when they are subject to baseline operating condition $\mathbf{S}_0=\{S_{0,l};l=1,2,\cdots,o\}$; and $\boldsymbol{\Omega}$ and $\boldsymbol{\Theta}$ are the parameter vectors of corresponding functions. Note that $A(\mathbf{S}_0;\boldsymbol{\Theta})=1$ and $\mu_i(\tau)=\mu_0(\tau)$ when $\mathbf{S}_i(\tau)=\mathbf{S}_0$.

Based on Eq. (5.1), the degradation increment $\Delta D_i(t)$ in time interval $(t-\Delta t,t]$ follows a normal distribution. Specifically

$$\Delta D_i(t)\sim N\left(M_i(t)-M_i(t-\Delta t),\sqrt{V_i(t)-V_i(t-\Delta t)}\right) \tag{5.2}$$

where

$$M_i(t):=\int_0^t \mu_i(\tau)\,d\tau \tag{5.3}$$

is the expected degradation increment of unit i at time t from its initial degradation value; and

$$V_i(t):=\int_0^t \sigma_i^2\mu_i(\tau)\,d\tau=\sigma_i^2 M_i(t) \tag{5.4}$$

is the variance of degradation increment of unit i at time t from its initial degradation value.

Assume unit i fails when its degradation value reaches a threshold h_i, its reliability is obtained as [142]

$$
\begin{aligned}
r_i(t) &= \Phi\left(\frac{h_i - x_i^0 - M_i(t)}{\sqrt{V_i(t)}}\right) - \exp\left(\frac{2M_i(t)\left(h_i - x_i^0\right)}{V_i(t)}\right)\Phi\left(-\frac{h_i - x_i^0 + M_i(t)}{\sqrt{V_i(t)}}\right) \\
&= \Phi\left(\frac{h_i - x_i^0 - M_i(t)}{\sigma_i\sqrt{M_i(t)}}\right) - \exp\left(\frac{2\left(h_i - x_i^0\right)}{\sigma_i^2}\right)\Phi\left(-\frac{h_i - x_i^0 + M_i(t)}{\sigma_i\sqrt{M_i(t)}}\right)
\end{aligned} \tag{5.5}
$$

where x_i^0 is the initial degradation value of unit i.

The corresponding *pdf* of lifetime of unit i is obtained as:

$$
\begin{aligned}
q_i(t) = -\phi\left(\frac{h_i - x_i^0 - M_i(t)}{\sigma_i\sqrt{M_i(t)}}\right) &\times \left(-\frac{\mu_i(t)}{2\sigma_i\sqrt{M_i(t)}} - \frac{\left(h_i - x_i^0\right)\mu_i(t)}{2\sigma_i M_i(t)\sqrt{M_i(t)}}\right) \\
+\exp\left(\frac{2\left(h_i - x_i^0\right)}{\sigma_i^2}\right)&\phi\left(-\frac{h_i - x_i^0 + M_i(t)}{\sigma_i\sqrt{M_i(t)}}\right) \\
\times &\left(-\frac{\mu_i(t)}{2\sigma_i\sqrt{M_i(t)}} + \frac{\left(h_i - x_i^0\right)\mu_i(t)}{2\sigma_i M_i(t)\sqrt{M_i(t)}}\right)
\end{aligned} \tag{5.6}
$$

Note that when we obtain $M_i(t)$ and $V_i(t)$, the values of stress factors $S_{il}(\tau)$ in $\mu_i(\tau)$ are not available for $\tau > T$ the observation time. In that case, we predict the stress factors after T using their means through time T. Suppose we obtain the measurements of the stress factor l of unit i at discrete time t_j, $S_{il}(t_j)$, by observation time T where $i = 1, 2, \cdots, 2n$ (because we have n pairs of units and hence $2n$ units in total) and $j = 1, 2, \cdots, m$ if there are m measurements by time T. The means of the stress factors $\bar{S}_{il}(T) = \sum_{j=1}^{m} S_{il}(t_j) \big/ m$ at time T are used as a prediction of $S_{il}(\tau)$ when $\tau > T$. Note that $\bar{\mathbf{S}}_i(T) = \left\{\bar{S}_{il}(T); l = 1, 2, \cdots, o\right\}$.

5.3.2 *Degradation Model Considering Observation Update and Standby*

Suppose we observe the degradation $D_i(T)$ of unit i by time T, and the operating condition, $\bar{\mathbf{S}}_i(T)$, is known. The expected degradation increment of unit i at t ($t \geq T$), i.e. $M_i(t)$, is updated accordingly. Specifically, Eq. (5.3) is modified as

$$M_i(t) = \left[D_i(T) - x_i^0 \right] + A\left(\bar{\mathbf{S}}_i(T)\right) \int_T^t \mu_0(\tau) d\tau \tag{5.7}$$

When standby is considered, the expected degradation increment of unit i at t, $M_i(t)$, is modified further to reflect the effect of possible standby periods during which standby pairs are not subject to degradation or failures. Note that for 1-out-of-6 pairs:G Balanced system, a pair can only be forced down into standby when the third failure occurs, and it resumes its operation when the fourth failure occurs, as shown in Figure 5.1. Two scenarios are considered, and the corresponding $M_i(t)$ is modified, respectively:

(i) After observation time T, unit i is forced down into standby at time τ_3 and not resumed by time t. In this scenario, the expected degradation increment does not increase after τ_3, hence

$$M_i(t;\tau_3) = \left[D_i(T) - x_i^0 \right] + A\left(\bar{\mathbf{S}}_i(T)\right) \int_T^{\tau_3} \mu_0(\tau) d\tau \tag{5.8}$$

(ii) After T, unit i is forced down into standby at time τ_3 and resumes operation at τ_4 where $\tau_3 < \tau_4 \leq t$. In this scenario, the expected degradation increment does not increase during $(\tau_3, \tau_4]$ and

$$M_i(t;\tau_3,\tau_4)=\left[D_i(T)-x_i^0\right]+A\left(\bar{\mathbf{S}}_i(T)\right)\int_T^{t-(\tau_4-\tau_3)}\mu_0(\tau)d\tau \tag{5.9}$$

The reliability, $r_i(t)$ in Eq. (5.5), and *pdf*, $q_i(t)$ in Eq. (5.6), of unit i in each scenario are then obtained based on the corresponding $M_i(t)$ in Eq. (5.8) and Eq. (5.9).

5.4 Reliability Metrics for 1-out-of-6 Pairs:G Balanced Systems with Units Performing One Function Considering Standby

In this section, we estimate the system reliability and the *pdf* of the time to the h^{th} failure of the 1-out-of-6 pairs:G Balanced system introduced in section 5.2. The reliability metrics are estimated at observation time T at which we assume no failure occurs.

The estimation is carried out according to the state transition diagram in Figure 5.1 from which all the possible events can be derived. In Figure 5.1, we observe an unbalanced system may result from the third failure. If this is indeed the case, an operating pair is forced down into standby to bring the system to balance. The standby pair resumes operation when the fourth failure occurs. If the third failure does not cause unbalanced system, then no pairs are forced into standby state.

In this section, we use the following notations: Denote the *pdf* of the lifetime of pair i^* as f_{i^*} and its reliability as R_{i^*}. The *pdf*s, q_i and q_{i+n}, and reliability functions, r_i and r_{i+n}, of individual units i and $(i+n)$ of pair i^* are obtained by Eq. (5.5) and Eq. (5.6), respectively. Then

$$R_{i^*}(t) = r_i(t) \cdot r_{i+n}(t) \tag{5.10}$$

and

$$f_{i^*}(t) = q_i(t) \cdot r_{i+n}(t) + r_i(t) \cdot q_{i+n}(t) \tag{5.11}$$

Note that the $M_i(t)$ used in Eq. (5.5) and Eq. (5.6) should be modified when standby occurs in some events as specified in Eq. (5.8) and Eq. (5.9). In addition, we define:

- U universal set of pair identity numbers $U = \{1^*, 2^*, 3^*, 4^*, 5^*, 6^*\}$.
- τ_h random variable of time to the h^{th} failure starting from time zero.
- $C_{v,S}^6$ set of combinations of v out of the 6 pairs the failure of which results in a symmetric (balanced) system.
- $C_{v,A}^6$ set of combinations of v out of the 6 pairs the failure of which results in an asymmetric (unbalanced) system.
- $C_{v,A,B}^6$ set of operating pairs that are forced down into standby to bring unbalanced systems back to balanced states. The i^{th} element in $C_{v,A}^6$ corresponds to the i^{th} element in $C_{v,A,B}^6$.

5.4.1 *System Reliability Estimation*

At observation time T, we sum the probabilities of all successful events to obtain system reliability $R_{sys}(t)$. Conditional system reliability, $R_{sys}(t|T)$, is obtained by normalizing system reliability by its value at time T. In particular,

$$R_{sys}(t) = \sum_{g=k}^{6} \sum_{w} P_g^{6,w}(t) \tag{5.12}$$

and

$$R_{sys}(t \mid T) = R_{sys}(t) / R_{sys}(T) \tag{5.13}$$

where $P_g^{6,w}(t)$ is the probability of a set of events that g out of 6 pairs operate in balanced states by time t $(t \geq T)$. The same equation is used to obtain the probabilities of a set of events that share the same or similar relative locations and relative sequence of failures though the actual identities of failed pairs in each event are different and hence the probabilities of the events are different. The superscript w is used to distinguish different sets of events that result in the same number of operating pairs out of the 6 pairs.

The probability that all pairs are operating by time t is

$$P_6^6(t) = \prod_{i^*=1}^{6} R_{i^*}(t) \tag{5.14}$$

The events that five pairs are operating by time t include all the state transition paths from state 1 to state 2. The probability is obtained by

$$P_5^6(t) = \sum_{i^*=1}^{6} \left[R_{i^*}(T) - R_{i^*}(t) \right] \prod_{j^* \in U - \{i^*\}} R_{j^*}(t) \tag{5.15}$$

The events that four pairs are operating by time t include all the state transition paths from state 1 to states 3, 4 and 5. The probability is obtained by

$$P_4^6(t) = \sum_{c \in C_{2,S}^6} \prod_{i^* \in c} \left[R_{i^*}(T) - R_{i^*}(t) \right] \prod_{j^* \in \bar{c}} R_{j^*}(t) \tag{5.16}$$

where $\prod_{i^* \in c}\left[R_{i^*}(T)-R_{i^*}(t)\right]$ is the probability that pairs in combination c fail in any order between time instants (T,t); set $C_{2,S}^{6}$ includes all the combinations of choosing two pair identity numbers out of the six pair identity numbers in U; and $\overline{c}=U-c$ is the complement set of c.

The only event for the system to have three operating pairs is that three pairs fail in a symmetric arrangement as in states 6 and 9. The events that three pairs are operating by time t hence include all the state transition paths from state 1 to states 6 and 9. The probability is obtained by

$$P_3^6(t)=\sum_{c\in C_{3,S}^6}\prod_{i^*\in c}\left[R_{i^*}(T)-R_{i^*}(t)\right]\prod_{j^*\in\overline{c}}R_{j^*}(t) \tag{5.17}$$

where set $C_{3,S}^{6}$, elements of which are combinations of three pair identity numbers, is found in Table 5.1.

As shown in Figure 5.1, states 7 and 8 with three failed pairs and states 10 and 11 with four failed pairs both have two operating pairs. The events that two pairs are operating by time t include all the state transition paths (i) from state 1 to states 7 and 8; (ii) from state 1 to states 10 and 11 via states 6 and 9; and (iii) from state 1 to states 10 and 11 via states 7 and 8. The probabilities of these events are obtained by Eq. (5.18), Eq. (5.19) and Eq. (5.20), respectively. The equations are derived based on the failures occurring sequentially at time τ_h ($h=1$ to the total number of failures, e.g. totally three failures are considered in Eq. (5.18)) and the survival of standby pairs and operating pairs by time t according to the corresponding state transitions paths. Specifically, Eq. (5.18) quantifies the probability of

all the events that two pairs in set $c-\left\{i_2^*\right\}$ fail between time instants (T,τ_3) in any order where $\tau_3 \in (T,t)$, then pair i_2^* fails at τ_3, and standby pair j^* and all the other pairs (pairs in set $\bar{c}-\left\{j^*\right\}$) survive time t. The first two failures result in a balanced system, but the third failure at τ_3 makes the system unbalanced. This is why we differentiate between the first two failures and the third one. Once a combination c from set $C_{3,A}^6$ is selected, the corresponding standby pair j^* in set $C_{3,A,B}^6$ is determined. But the three pairs in c can fail in any order. So the third failure occurs in any pair in c which results in the second level of summation in Eq. (5.18). Specifically

$$P_2^{6,1}(t)=\sum_{\substack{c\in C_{3,A}^6\\ j^*\in C_{3,A,B}^6}}\sum_{i_2^*\in c}\left\{\begin{array}{l}\int_{\tau_3=T}^{t}\prod_{i_1^*\in c-\left\{i_2^*\right\}}\left[R_{i_1^*}(T)-R_{i_1^*}(\tau_3)\right]f_{i_2^*}(\tau_3)R_{j^*}(t;\tau_3)\,d\tau_3\\ \times\prod_{h^*\in\bar{c}-\left\{j^*\right\}}R_{h^*}(t)\end{array}\right\} \tag{5.18}$$

where set $C_{3,A}^6$ and set $C_{3,A,B}^6$ are found in Table 5.1; and $R_{j^*}(t;\tau_3)$ is the reliability of the standby pair j^* given it is forced down at time τ_3 and not resumed by time t. To obtain $R_{j^*}(t;\tau_3)$, we use Eq. (5.8) to obtain the corresponding expected degradation increments at t, i.e. $M_j(t;\tau_3)$ and $M_{j+n}(t;\tau_3)$, of units j and $(j+n)$ in pair j^*. Similarly Eq. (5.19) quantifies the probability of all the events that three pairs in set c fail in any order between (T,τ_4) where $\tau_4 \in (T,t)$, which results in a balanced system, then pair j^* fails at τ_4, and all the other pairs survive time t.

$$P_2^{6,2}(t)=\sum_{c\in C_{3,S}^6}\sum_{j^*\in\bar{c}}\left\{\int_{\tau_4=T}^{t}\prod_{i^*\in c}\left[R_{i^*}(T)-R_{i^*}(\tau_4)\right]f_{j^*}(\tau_4)\,d\tau_4\prod_{h^*\in\bar{c}-\left\{j^*\right\}}R_{h^*}(t)\right\} \tag{5.19}$$

Eq. (5.20) quantifies the probability of all the events that two pairs in set $c-\left\{i_2^*\right\}$ fail in any order between (T,τ_3), then pair i_2^* fails at τ_3, then pair h^* fails at τ_4 where $\tau_3 \in (T,\tau_4)$ and $\tau_4 \in (T,t)$, and then both pair j^* and pair m^* survive time t. The third failure results in an unbalanced system, so pair j^* is forced down into standby at τ_3 to rebalance the system. When the fourth failure occurs at τ_4, pair j^* is no longer needed in standby and hence resumes operation. Note that τ_3 is smaller than τ_4 so the upper limit of τ_3 is τ_4. Specifically

$$P_2^{6,3}(t)=\sum_{\substack{c\in C_{3,A}^6 \\ j^*\in C_{3,A,B}^6}}\sum_{\substack{h^*\in \bar{c}-\{j^*\} \\ m^*\in \bar{c}-\left\{j^*,h^*\right\}}}\sum_{i_2^*\in c}\left\{\int_{\tau_4=T}^{t}\int_{\tau_3=T}^{\tau_4}\left\{\begin{array}{l}\prod_{i_1^*\in c-\left\{i_2^*\right\}}\left[R_{i_1^*}(T)-R_{i_1^*}(\tau_3)\right] \\ f_{i_2^*}(\tau_3)\,f_{h^*}(\tau_4) \\ R_{j^*}(t;\tau_3,\tau_4)\,d\tau_3 d\tau_4\end{array}\right\}R_{m^*}(t)\right\} \quad (5.20)$$

where $R_{j^*}(t;\tau_3,\tau_4)$ is the reliability of pair j^* given it is forced down into standby during τ_3 and τ_4. To obtain the value of $R_{j^*}(t;\tau_3,\tau_4)$, we use Eq. (5.9) to obtain the corresponding expected degradation increments at t, i.e. $M_j(t;\tau_3,\tau_4)$ and $M_{j+n}(t;\tau_3,\tau_4)$, of units j and $(j+n)$ in pair j^*.

The events that one pair is operating by time t include all the state transition paths from state 1 to state 12 (i) via states 6 and 9; and (ii) via states 7 and 8. The probabilities are obtained by Eq. (5.21) and Eq. (5.22), respectively. Specifically, the events modeled by Eq. (5.21) are the same as Eq. (5.19) except that one of the two pairs in $\bar{c}-\left\{j^*\right\}$ fails between

(τ_4, t) and the other one survives time t. Thus we have the third level of summation in Eq. (5.21).

$$P_1^{6,1}(t) = \sum_{c \in C_{3,S}^6} \sum_{j^* \in \bar{c}} \sum_{\substack{h^* \in \bar{c} - \{j^*\} \\ m^* \in \bar{c} - \{j^*, h^*\}}} \left\{ \int_{\tau_4 = T}^{t} \left\{ \begin{array}{l} \prod_{i^* \in c} \left[R_{i^*}(T) - R_{i^*}(\tau_4) \right] f_{j^*}(\tau_4) \\ \left[R_{h^*}(\tau_4) - R_{h^*}(t) \right] d\tau_4 \end{array} \right\} R_{m^*}(t) \right\} \quad (5.21)$$

In addition, the events modeled by Eq. (5.22) are the same as Eq. (5.20) except that either pair j^*, the standby pair, or pair m^* fails between (τ_4, t).

$$P_1^{6,2}(t) = \sum_{\substack{c \in C_{3,A}^6 \\ j^* \in C_{3,A,B}^6}} \sum_{\substack{h^* \in \bar{c} - \{j^*\} \\ m^* \in \bar{c} - \{j^*, h^*\}}} \sum_{i_2^* \in c} \left\{ \begin{array}{l} \int_{\tau_4 = T}^{t} \int_{\tau_3 = T}^{\tau_4} \left\{ \begin{array}{l} \prod_{i_1^* \in c - \{i_2^*\}} \left[R_{i_1^*}(T) - R_{i_1^*}(\tau_3) \right] \\ \left[R_{j^*}(\tau_4; \tau_3, \tau_4) - R_{j^*}(t; \tau_3, \tau_4) \right] \\ f_{i_2^*}(\tau_3) f_{h^*}(\tau_4) R_{m^*}(t) d\tau_3 d\tau_4 \end{array} \right\} \\ + \int_{\tau_4 = T}^{t} \int_{\tau_3 = T}^{\tau_4} \left\{ \begin{array}{l} \prod_{i_1^* \in c - \{i_2^*\}} \left[R_{i_1^*}(T) - R_{i_1^*}(\tau_3) \right] \\ \left[R_{m^*}(\tau_4) - R_{m^*}(t) \right] R_{j^*}(t; \tau_3, \tau_4) \\ f_{i_2^*}(\tau_3) f_{h^*}(\tau_4) d\tau_3 d\tau_4 \end{array} \right\} \end{array} \right\} \quad (5.22)$$

Table 5.1 Set $C_{3,S}^6$, $C_{3,A}^6$ and $C_{3,A,B}^6$

$C_{3,S}^6$	$C_{3,A}^6$	$C_{3,A,B}^6$
$1^* 2^* 3^*$	$2^* 4^* 5^*$	1^*
$2^* 3^* 4^*$	$1^* 4^* 5^*$	2^*
$3^* 4^* 5^*$	$1^* 2^* 5^*$	4^*
$4^* 5^* 6^*$	$1^* 2^* 4^*$	5^*
$1^* 5^* 6^*$	$3^* 5^* 6^*$	2^*
$1^* 2^* 6^*$	$2^* 5^* 6^*$	3^*
$1^* 3^* 5^*$	$2^* 3^* 6^*$	5^*
$2^* 4^* 6^*$	$2^* 3^* 5^*$	6^*
	$3^* 4^* 6^*$	1^*
	$1^* 4^* 6^*$	3^*
	$1^* 3^* 6^*$	4^*
	$1^* 3^* 4^*$	6^*

5.4.2 *The pdf of Time to the h^{th} Failure*

For redundant systems such as k-out-of-n pairs:G Balanced systems investigated in this chapter, it is important to estimate the time to the h^{th} failure to support maintenance actions. In this section, we discuss the estimation for the *pdf* of the time to the h^{th} failure. Again, we consider the 1-out-of-6 pairs:G Balanced system.

Let $g_h(\tau_h)$ denote the *pdf* of time to the h^{th} failure, τ_h, starting from time zero. The sets $C_{3,S}^6$, $C_{3,A}^6$, and $C_{3,A,B}^6$ used in the following equations are found in Table 5.1. In addition, $C_{2,S}^6$ is composed of all the combinations of choosing two pair identity numbers out of the six pair identity numbers in U.

We estimate *pdf* of time to the h^{th} failure as follows. The *pdf* of time to the first failure is

$$g_1(\tau_1) = \sum_{i^*=1}^{6} f_{i^*}(\tau_1) \prod_{j^* \in U - \{i^*\}} R_{j^*}(\tau_1) \tag{5.23}$$

By considering the scenarios where states 3, 4, or 5 are reached at τ_2, as shown in Figure 5.1, the *pdf* of time to the second failure is obtained as in Eq. (5.24). Specifically, Eq. (5.24) is the probability density of all the events that pair i_1^* fails between (T, τ_2), then pair i_2^* fails at τ_2, and all the other pairs survives beyond τ_2.

$$g_2(\tau_2) = \sum_{c \in C_{2,S}^6} \sum_{\substack{i_1^* \in c \\ i_2^* \in c - \{i_1^*\}}} \left[R_{i_1^*}(T) - R_{i_1^*}(\tau_2) \right] f_{i_2^*}(\tau_2) \prod_{j^* \in \bar{c}} R_{j^*}(\tau_2) \tag{5.24}$$

Two scenarios can happen when the third failure occurs: (i) three pairs fail in symmetric arrangements at τ_2, i.e. either state 6 or state 9 is reached at τ_2; or (ii) three pairs fail in asymmetric arrangements and one operating pair is forced down into standby when the third failure occurs at τ_3, i.e. either state 7 or state 8 is reached at τ_3. The *pdf* of time to the third failure $g_3(\tau_3)$ is hence composed of two parts: $g_{3,S}(\tau_3)$ and $g_{3,A}(\tau_3)$ which

correspond to the two scenarios, as shown in Eq. (5.26) and Eq. (5.27). Specifically, the two equations quantify the probability density of all the events that two pairs fail between (T,τ_3), then pair i_2^* fails at τ_3, which results in a balanced system, as in Eq. (5.26), or an unbalanced system, as in Eq. (5.27), and all the other pairs survive τ_3.

$$g_3(\tau_3) = g_{3,S}(\tau_3) + g_{3,A}(\tau_3) \tag{5.25}$$

where

$$g_{3,S}(\tau_3) = \sum_{c\in C_{3,S}^6} \sum_{i_2^*\in c} \left\{ \begin{array}{l} \prod_{i_1^*\in c-\{i_2^*\}} \left[R_{i_1^*}(T) - R_{i_1^*}(\tau_3) \right] \\ f_{i_2^*}(\tau_3) \prod_{j^*\in\bar{c}} R_{j^*}(\tau_3) \end{array} \right\} \tag{5.26}$$

and

$$g_{3,A}(\tau_3) = \sum_{c\in C_{3,A}^6} \sum_{i_2^*\in c} \left\{ \begin{array}{l} \prod_{i_1^*\in c-\{i_2^*\}} \left[R_{i_1^*}(T) - R_{i_1^*}(\tau_3) \right] \\ f_{i_2^*}(\tau_3) \prod_{j^*\in\bar{c}} R_{j^*}(\tau_3) \end{array} \right\} \tag{5.27}$$

Similarly, $g_4(\tau_4)$, $g_5(\tau_5)$, and $g_6(\tau_6)$ are composed of two parts, which correspond to the two possible branches of transition paths after the third failure. The *pdf*s can be obtained as follows.

By considering the scenario that either state 10 or state 11 is reached at τ_4 via either state 6 or state 9, we obtain the first part of $g_4(\tau_4)$ as

$$g_{4,S}(\tau_4)=\sum_{c\in C_{3,S}^6}\sum_{j^*\in\bar{c}}\left\{\begin{array}{l}\prod\limits_{i^*\in c}\left[R_{i^*}(T)-R_{i^*}(\tau_4)\right]\\ f_{j^*}(\tau_4)\prod\limits_{h^*\in\bar{c}-\{j^*\}}R_{h^*}(\tau_4)\end{array}\right\} \tag{5.28}$$

Eq. (5.28) quantifies the probability density of all the events that the three pairs in set c fail in any order between (T,τ_4), which results in a balanced system, then pair j^* fails at τ_4, and all the other pairs survive time τ_4. By considering the scenario that either state 10 or state 11 is reached at τ_4 via either state 7 or state 8, we obtain the second part of $g_4(\tau_4)$ as

$$g_{4,A}(\tau_4)=\sum_{\substack{c\in C_{3,A}^6\\ j^*\in C_{3,A,B}^6}}\sum_{\substack{h^*\in\bar{c}-\{j^*\}\\ m^*\in\bar{c}-\{j^*,h^*\}}}\sum_{i_2^*\in c}\int_{\tau_3=T}^{\tau_4}\left\{\begin{array}{l}\prod\limits_{i_1^*\in c-\{i_2^*\}}\left[R_{i_1^*}(T)-R_{i_1^*}(\tau_3)\right]\\ f_{i_2^*}(\tau_3)R_{j^*}(\tau_3)d\tau_3\end{array}\right\}f_{h^*}(\tau_4)R_{m^*}(\tau_4) \tag{5.29}$$

Eq. (5.29) quantifies the probability density of all the events that the two pairs in set $c-\{i_2^*\}$ fail in any order between (T,τ_3) where $\tau_3\in(T,\tau_4)$, then pair i_2^* fails at τ_3, which results in an unbalanced system, pair j^* is forced down into standby at τ_3 to rebalance the system, which means pair j^* survives τ_3, then pair h^* fails at τ_4, and pair m^* survive time τ_4. Similarly, we obtain $g_{5,S}(\tau_5)$, $g_{5,A}(\tau_5)$, $g_{6,S}(\tau_6)$, and $g_{6,A}(\tau_6)$. Then $g_5(\tau_5)=g_{5,S}(\tau_5)+g_{5,A}(\tau_5)$ and $g_6(\tau_6)=g_{6,S}(\tau_6)+g_{6,A}(\tau_6)$. The derivations of Eq. (5.30) to Eq. (5.33) are obvious by referring to the above explanation.

$$g_{5,S}(\tau_5)=\sum_{c\in C_{3,S}^6}\sum_{j^*\in\bar{c}}\sum_{\substack{h^*\in\bar{c}-\{j^*\}\\ m^*\in\bar{c}-\{j^*,h^*\}}}\left\{\begin{array}{l}\int_{\tau_4=T}^{\tau_5}\prod_{i^*\in c}\left[R_{i^*}(T)-R_{i^*}(\tau_4)\right]f_{j^*}(\tau_4)\,d\tau_4\\ f_{h^*}(\tau_5)R_{m^*}(\tau_5)\end{array}\right\} \tag{5.30}$$

$$\begin{aligned}&g_{5,A}(\tau_5)\\ &=\sum_{\substack{c\in C_{3,A}^6\\ j^*\in C_{3,A,B}^6}}\sum_{\substack{h^*\in\bar{c}-\{j^*\}\\ m^*\in\bar{c}-\{j^*,h^*\}}}\sum_{i_2^*\in c}\int_{\tau_4=T}^{\tau_5}\int_{\tau_3=T}^{\tau_4}\left\{\begin{array}{l}\prod_{i_1^*\in c-\{i_2^*\}}\left[R_{i_1^*}(T)-R_{i_1^*}(\tau_3)\right]R_{m^*}(\tau_5)\\ f_{i_2^*}(\tau_3)\,f_{h^*}(\tau_4)\,f_{j^*}(\tau_5;\tau_3,\tau_4)\,d\tau_3 d\tau_4\end{array}\right\}\\ &+\sum_{\substack{c\in C_{3,A}^6\\ j^*\in C_{3,A,B}^6}}\sum_{\substack{h^*\in\bar{c}-\{j^*\}\\ m^*\in\bar{c}-\{j^*,h^*\}}}\sum_{i_2^*\in c}\int_{\tau_4=T}^{\tau_5}\int_{\tau_3=T}^{\tau_4}\left\{\begin{array}{l}\prod_{i_1^*\in c-\{i_2^*\}}\left[R_{i_1^*}(T)-R_{i_1^*}(\tau_3)\right]R_{j^*}(\tau_5;\tau_3,\tau_4)\\ f_{i_2^*}(\tau_3)\,f_{h^*}(\tau_4)\,f_{m^*}(\tau_5)\,d\tau_3 d\tau_4\end{array}\right\}\end{aligned} \tag{5.31}$$

where $f_{j^*}(\tau_5;\tau_3,\tau_4)$ is the *pdf* of pair j^* at τ_5 given it is forced down into standby between τ_3 and τ_4. To obtain the value for $f_j(\tau_5;\tau_3,\tau_4)$, we use Eq. (5.9) to obtain corresponding expected degradation increments of units j and $(j+n)$ at τ_5, i.e. $M_j(\tau_5;\tau_3,\tau_4)$ and $M_{j+n}(\tau_5;\tau_3,\tau_4)$.

$$g_{6,S}(\tau_6)=\sum_{c\in C_{3,S}^6}\sum_{j^*\in\bar{c}}\sum_{\substack{h^*\in\bar{c}-\{j^*\}\\ m^*\in\bar{c}-\{j^*,h^*\}}}\int_{\tau_4=T}^{\tau_6}\left\{\begin{array}{l}\prod_{i^*\in c}\left[R_{i^*}(T)-R_{i^*}(\tau_4)\right]f_{j^*}(\tau_4)\\ \left[R_{h^*}(\tau_4)-R_{h^*}(\tau_6)\right]d\tau_4\end{array}\right\}f_{m^*}(\tau_6) \tag{5.32}$$

$$
\begin{aligned}
& g_{6,A}(\tau_6) \\
&= \sum_{\substack{c \in C_{3,A}^6 \\ j^* \in C_{3,A,B}^6}} \sum_{h^* \in \bar{c} - \{j^*\}} \sum_{i_2^* \in c} \int_{\tau_4=T}^{\tau_6} \int_{\tau_3=T}^{\tau_4} \left\{ \begin{array}{l} \prod_{i_1^* \in c - \{i_2^*\}} \left[R_{i_1^*}(T) - R_{i_1^*}(\tau_3) \right] \\ f_{i_2^*}(\tau_3)\, f_{h^*}(\tau_4) \left[R_{m^*}(\tau_4) - R_{m^*}(\tau_6) \right] \\ f_{j^*}(\tau_6; \tau_3, \tau_4)\, d\tau_3 d\tau_4 \end{array} \right\} \\
&+ \sum_{\substack{c \in C_{3,A}^6 \\ j^* \in C_{3,A,B}^6}} \sum_{h^* \in \bar{c} - \{j^*\}} \sum_{i_2^* \in c} \int_{\tau_4=T}^{\tau_6} \int_{\tau_3=T}^{\tau_4} \left\{ \begin{array}{l} \prod_{i_1^* \in c - \{i_2^*\}} \left[R_{i_1^*}(T) - R_{i_1^*}(\tau_3) \right] \\ f_{i_2^*}(\tau_3)\, f_{h^*}(\tau_4)\, f_{m^*}(\tau_6) \\ \left[R_{j^*}(\tau_4; \tau_3, \tau_4) - R_{j^*}(\tau_6; \tau_3, \tau_4) \right] d\tau_3 d\tau_4 \end{array} \right\}
\end{aligned} \tag{5.33}
$$

The conditional *pdf* can be obtained by normalizing the *pdf* using a factor ξ.

$$\xi = R_{sys}(T) \tag{5.34}$$

If we estimate the *pdf* at $T = 0$, then $\xi = 1$. Otherwise, the range of *pdf* should exclude $[0, T]$, and *pdf* should then be normalized by ξ which is the probability that no failure occurs by observation time T.

5.5 Model Generalization

In this section, we present a procedure for estimating reliability metrics of any k-out-of-n pairs:G Balanced systems. In the previous section, we estimate the system reliability and *pdf* of time to the h^{th} failure of the 1-out-of-6 pairs:G Balanced system with all units performing the same function at observation time T under the assumption that no failure occurs by T. In this section, we generalize the model to estimate the reliability metrics

when any possible system state is observed by time T. Consider the 1-out-of-6 pairs:G Balanced system as an example. When state 7 in Figure 5.1 is observed at time T, the system already has three pairs failed and one pair in standby. In this case, we estimate the system reliability, *pdf*s of times to the 4th, 5th, and 6th failures, and their conditional values by considering all the possible state transition paths derived from state 7, as shown in Figure 5.1.

Again, we assume that the degradation processes of individual units are affected by their corresponding operating conditions. We use the following notations and define:

- U universal set of pair identity numbers.
- u number of failed pairs by observation time T.
- τ_h random variable of time to the h^{th} failure.
- $\bar{\tau}_h$ actual time to the h^{th} failure which is known after the h^{th} failure is observed by time T.

5.5.1 *System Reliability Estimation*

At any observation time T, system reliability at t $(t \geq T)$ can be obtained by

$$R_{sys}(t) = \sum_{h=u}^{n-k} \sum_{w} P_h^{u,w}(t) \tag{5.35}$$

where $P_h^{u,w}(t)$ is the probability of a successful event that h pairs fail sequentially by time t $(t \geq T)$, which results in a balanced system with no less than k operating pairs by time t given that u pairs have already failed by observation time T. The superscript w is used

to distinguish different events which all have the same values of u and h. The conditional system reliability is obtained by Eq. (5.13).

5.5.1.1 *Probability of Each Successful Event*

Probability $P_h^{u,w}(t)$ is obtained as

$$P_h^{u,w}(t)=\int_{\tau_h=T}^{t}\int_{\tau_{h-1}=T}^{\tau_h}\cdots\int_{\tau_{u+1}=T}^{\tau_{u+2}}\left\{\begin{array}{l}\displaystyle\prod_{i=u+1}^{h} f_{x_i^*}\left(\tau_i;\mathbf{b}_{h,x_i^*}^{u,w}\right)\\ \displaystyle\prod_{y^*\in Y_h^{u,w}} R_{y^*}\left(t;\mathbf{b}_{h,y^*}^{u,w}\right)\\ \displaystyle\prod_{z^*\in G_h^{u,w}} R_{z^*}\left(t;\mathbf{b}_{h,z^*}^{u,w}\right)\end{array}\right\}\prod_{i=u+1}^{h} d\tau_i \tag{5.36}$$

where i is the index for the failures that occur by time t; x_i^* is the identity number of the i^{th} failed pair and can be found in the set of identity numbers of failed pairs at time t which we denote as $X_h^{u,w}$; $Y_h^{u,w}$ is the set of identity numbers of standby pairs at time t; $G_h^{u,w}$ is the set of identity numbers of operating pairs at time t; and $\mathbf{b}_{h,p^*}^{u,w}$ is a row vector of length h that records the actions, which include being forced down into standby, resuming operation, and either continuing operating or failing, that pair p^* is subjected to at each failure time τ_i (or $\bar{\tau}_i$) ($i=1,\cdots,h$) from time $t=0$. Note that the two units in pair p^*, i.e. units p and $(p+n)$ share the same vector $\mathbf{b}_{h,p^*}^{u,w}$ because both units are forced down and resumed simultaneously. Again, we denote $\bar{\tau}_i$ ($i=1$ to u) as the times when the first u failures actually occur before observation time T, which have deterministic values, and denote τ_i ($i=u+1$ to h) as the times when the later $(h-u)$ failures occur after T, which

are random variables. Specifically, the i^{th} element in $\mathbf{b}_{h,p^*}^{u,w}$, $\mathbf{b}_{h,p^*}^{u,w}(i)$, equals -1 if pair p^* is forced down into standby at $\bar{\tau}_i$ (or τ_i); $\mathbf{b}_{h,p^*}^{u,w}(i)=1$ if pair p^* resumes operation at $\bar{\tau}_i$ (or τ_i); and $\mathbf{b}_{h,p^*}^{u,w}(i)=0$ otherwise. Note that the elements of $\mathbf{b}_{h,p^*}^{u,w}$ include not only the actions at $\bar{\tau}_i$ before T; but also the actions at τ_i between time interval $(T,t]$.

In (5.36), $\prod_{i=u+1}^{h} f_{x_i^*}\left(\tau_i;\mathbf{b}_{h,x_i^*}^{u,w}\right)d\tau_i$ is the probability that the i^{th} failure occurs at time τ_i where $i=(u+1),\cdots,h$ and $\tau_i\in(T,\tau_{i+1})$ for $i<h$ and $\tau_h\in(T,t)$; $\prod_{y^*\in Y_h^{u,w}} R_{y^*}\left(t;\mathbf{b}_{h,y^*}^{u,w}\right)$ and $\prod_{z^*\in G_h^{u,w}} R_{z^*}\left(t;\mathbf{b}_{h,z^*}^{u,w}\right)$ are the conditional probabilities that standby pairs in $Y_h^{u,w}$ and operating pairs in $G_h^{u,w}$ survive time t given the i^{th} failure occurs at time τ_i where $i=(u+1),\cdots,h$.

When $h=u$, we modify (5.36) as

$$P_u^u(t)=\prod_{y^*\in Y_h^u} R_{y^*}\left(t;\mathbf{b}_{u,y^*}^u\right)\prod_{z^*\in G_h^u} R_{z^*}\left(t;\mathbf{b}_{u,z^*}^u\right) \tag{5.37}$$

where we do not have superscript w because there is only one event where no further failures occur after observation time T.

To obtain the reliability and lifetime *pdf* of individual pairs, i.e. $R_{p^*}\left(\hat{t};\mathbf{b}_{h,p^*}^{u,w}\right)$ and $f_{p^*}\left(\hat{t};\mathbf{b}_{h,p^*}^{u,w}\right)$, where $\hat{t}=\tau_i$ if pair p^* fails at τ_i and $\hat{t}=t$ if pair p^* is either operating or in standby at t, in the expression of $P_h^{u,w}(t)$ as shown in Eq. (5.36), we obtain

$M_p\left(\hat{t};\mathbf{b}_{h,p^*}^{u,w}\right)$ and $M_{p+n}\left(\hat{t};\mathbf{b}_{h,p^*}^{u,w}\right)$ for units p and $(p+n)$ in pair p^* first. In particular, for unit p

$$M_p\left(\hat{t};\mathbf{b}_{h,p^*}^{u,w}\right)=\left[D_p(T)-x_p^0\right]+A\left(\bar{\mathbf{S}}_p(T)\right)\int_{\underline{T}}^{\bar{T}}\mu_0(\tau)d\tau \tag{5.38}$$

where the lower bound of the integral is

$$\underline{T}=\left[1+\sum_{c=1}^{u}\mathbf{b}_{h,p^*}^{u,w}(c)\right]T-\sum_{c=1}^{u}\mathbf{b}_{h,p^*}^{u,w}(c)\bar{\tau}_c \tag{5.39}$$

and the upper bound of the integral is

$$\bar{T}=\left[1+\sum_{c=1}^{h}\mathbf{b}_{h,p^*}^{u,w}(c)\right]\hat{t}-\left(\sum_{c=1}^{u}\mathbf{b}_{h,p^*}^{u,w}(c)\bar{\tau}_c+\sum_{c=u+1}^{h}\mathbf{b}_{h,p^*}^{u,w}(c)\tau_c\right) \tag{5.40}$$

In Eqs. (5.39) and (5.40), the coefficients $\left[1+\sum_{c=1}^{u}\mathbf{b}_{h,p^*}^{u,w}(c)\right]$ and $\left[1+\sum_{c=1}^{h}\mathbf{b}_{h,p^*}^{u,w}(c)\right]$ before T and $\hat{t}$ equal either 1 or 0 because: first, vector $\mathbf{b}_{h,p^*}^{u,w}$ either has the same number of "–1" (representing forcing-down) and "1" (representing resumption) or has one more "–1" than "1"; second, vector $\mathbf{b}_{h,p^*}^{u,w}$ must have exactly one "1" between two "–1"; and third, the other elements besides "–1" and "1" in vector $\mathbf{b}_{h,p^*}^{u,w}$ are all zeroes. The coefficients equal 0 if and only if pair p^* is in standby at T or $\hat{t}$. If a coefficient is 1, $\underline{T}$ (or $\bar{T}$) is T (or $\hat{t}$) minus the time interval during which pair p^* is in standby, i.e. $\sum_{c=1}^{u}\mathbf{b}_{h,p^*}^{u,w}(c)\bar{\tau}_c$ (or $\left(\sum_{c=1}^{u}\mathbf{b}_{h,p^*}^{u,w}(c)\bar{\tau}_c+\sum_{c=u+1}^{h}\mathbf{b}_{h,p^*}^{u,w}(c)\tau_c\right)$). If a coefficient is 0, then $\underline{T}$ (or $\bar{T}$) is the last time when pair p^* is forced down, which is the greatest $\bar{\tau}_c$ (or τ_c) that corresponds to a

$\mathbf{b}_{h,p^*}^{u,w}(c) = -1$ minus the time interval during which pair p^* is in standby. For example, if pair p^* is forced down at $\tau_{h-1}(h-1>u)$ and never resumes operation afterwards, then $\bar{T}$ is τ_{h-1} minus $\left(\sum_{c=1}^{u}\mathbf{b}_{h,p^*}^{u,w}(c)\bar{\tau}_c + \sum_{c=u+1}^{h-2}\mathbf{b}_{h,p^*}^{u,w}(c)\tau_c\right)$.

Similarly $M_{p+n}\left(\hat{t};\mathbf{b}_{h,p^*}^{u,w}\right)$ is obtained by changing the subscript p in Eq. (5.38) to $(p+n)$ and using the same lower and upper bounds $\underline{T}$ and $\bar{T}$ as obtained in Eq. (5.39) and Eq. (5.40) because both units p and $(p+n)$ are forced down and resumed together as a pair p^* as indicated in $\mathbf{b}_{h,p^*}^{u,w}$.

5.5.1.2 *Event Set Enumeration for System Reliability Estimation*

In this section, we introduce the procedure to obtain the event set for estimating system reliability given the observation of the system at time T. Starting from the observed system state at T, we enumerate all its follow-up states until we exhaust all the possible states and all the possible transitions between these states. A follow-up state of a state can be obtained by turning one of the operating pairs in the current system into failure and rebalancing it if the additional failure results in an unbalanced system. A successful event for estimating system reliability is a transition path that leads to a successful state, i.e. a balanced state with at least k operating pairs. Consider the state transition diagram in Figure 5.1, the successful events of a 2-out-of-6 pairs:G Balanced system are all the transition paths that lead to states 1 to 11.

The procedure of enumerating the successful event set for system reliability estimation given we observe u failed pairs at time T is as follows.

Step 1. If $n-k \geq u$, record the current system state, which is a successful state, as S_u^u, and the event of staying in the current state, which is a successful event, as E_u^u. We record the actions, such as being forced down into standby, resuming operation, and either continuing operating or failing, that pair p^* is subject to at each observed failure time $\bar{\tau}_i$ ($i=1,\cdots,u$) by observation time T in a row vector $\mathbf{b}_{u,p^*}^u$ ($p^*=1^*,\cdots,n^*$) where $\mathbf{b}_{u,p^*}^u$ is composed of elements with values 0, –1, and 1 as explained above. In addition, the n vectors compose a matrix, $\mathbf{B}_u^u$, with n rows and u columns with each row of $\mathbf{B}_u^u$ being $\mathbf{b}_{u,p^*}^u$ for an individual pair p^*. In addition, record the identity numbers of failed pairs in X_u^u in the order of failures, and the identity numbers of remaining operating pairs in G_u^u.

Step 2. If $n-k \geq u+1$, enumerate all the follow-up states of the observed system state S_u^u. Each follow-up state with at least k operating pairs is a successful state, and the transition path that leads to this state is a successful event. Record the successful states and events obtained in this step in $\left\{S_{u+1}^{u,w}\right\}$ and $\left\{E_{u+1}^{u,w}\right\}$, respectively where w is used to distinguish different elements in the sets. For each $E_{u+1}^{u,w}$ we record the matrix, $\mathbf{B}_{u+1}^{u,w}$, with n rows and $(u+1)$ columns with each row being vector $\mathbf{b}_{u+1,p^*}^{u,w}$ for pair p^*. The first u columns of $\mathbf{B}_{u+1}^{u,w}$ is $\mathbf{B}_u^u$. The $(u+1)^{\text{st}}$ column of $\mathbf{B}_{u+1}^{u,w}$ is obtained based on the forcing-down or

resumption of the individual pairs at τ_{u+1} when we derive successful state $S_{u+1}^{u,w}$ from S_u^u. In addition, record the corresponding set of identity numbers of failed pairs, $X_{u+1}^{u,w}$, in the order of failures, and the set of identity numbers of remaining operating pairs, $G_{u+1}^{u,w}$, for each successful event.

Step 3. Repeat Step 2 to enumerate the other successful events with h failed pairs iteratively, where $n-k \geq h > u+1$. In each repetition of Step 2, enumerate all the follow-up states of the successful states with $(h-1)$ failed pairs in $\left\{S_{h-1}^{u,w}\right\}$. Record all the successful states and successful events obtained in this step in sets $\left\{S_h^{u,w}\right\}$ and $\left\{E_h^{u,w}\right\}$, respectively. The first $(h-1)$ columns of $\mathbf{B}_h^{u,w}$ equal to $\mathbf{B}_{h-1}^{u,v}$ when $S_h^{u,w}$ is a follow-up state of $S_{h-1}^{u,v}$. We obtain the h^{th} column of $\mathbf{B}_h^{u,w}$ based on the forcing-down or resumption of all the individual pairs at τ_h when we derive successful state $S_h^{u,w}$ from $S_{h-1}^{w,v}$. In addition, record the set of identity numbers of failed pairs, $X_h^{u,w}$, in the order of failures, and the set of identity numbers of remaining operating pairs, $G_h^{u,w}$, for each successful event.

In each step, the set of pairs in standby, i.e. $Y_h^{u,w}$, can be obtained by finding the complement set of failed pairs and operating pairs given the universal set U which includes all the identity numbers of individual pairs.

5.5.2 *Estimation of the Distribution of the Time to the h[th] Failure*

The estimation of *pdf* of time to the h^{th} failure after a system state with u failed pairs is observed, denoted as $g_h^u(\tau_h)$, should consider all the possible events that lead to the h^{th} failure where $h > u$, which are all the state transition paths from the observed state to states with h failed pairs via successful states with $(h-1)$ failed units. We can determine the event set immediately based on the set of successful events for system reliability estimation that leads to $(h-1)$ failed pairs, i.e. $\{E_{h-1}^{u,w}\}$, by considering all the follow-up states of $\{S_{h-1}^{u,w}\}$. In other words, each event considered in the estimation of $g_h^u(\tau_h)$ is a state transition path recorded in $E_{h-1}^{u,w}$ extended by an additional failure of the remaining operating pairs recorded in $G_{h-1}^{u,w}$. Note that the h^{th} failure here does not necessarily result in a successful system state. Thus, the event set for estimating the *pdf* of time to the h^{th} failure is not the same as $\{E_h^{u,w}\}$, the set of successful events that have h pairs failed. The *pdf* of the time to the h^{th} failure can then be estimated by summing up all the probability densities of the corresponding events.

$$g_h^u(\tau_h) = \sum_w \sum_{x_h^* \in G_{h-1}^{u,w}} \int_{\tau_{h-1}=T}^{\tau_h} \cdots \int_{\tau_{u+1}=T}^{\tau_{u+2}} \left\{ \begin{array}{l} f_{x_h^*}\left(\tau_h; \mathbf{b}_{h-1,x_h^*}^{u,w}\right) \\ \prod_{i=u+1}^{h-1} f_{x_i^*}\left(\tau_i; \mathbf{b}_{h-1,x_i^*}^{u,w}\right) \\ \prod_{y^* \in Y_{h-1}^{u,w}} R_{y^*}\left(\tau_h; \mathbf{b}_{h-1,y^*}^{u,w}\right) \\ \prod_{z^* \in G_{h-1}^{u,w} - \{x_h^*\}} R_{z^*}\left(\tau_h; \mathbf{b}_{h-1,z^*}^{u,w}\right) \end{array} \right\} \prod_{i=u+1}^{h-1} d\tau_i \qquad (5.41)$$

where $G_{h-1}^{u,w}$ is the set of operating pairs corresponding to the successful state $S_{h-1}^{u,w}$.

In Eq. (5.41), $\prod_{i=u+1}^{h-1} f_{x_i^*}\left(\tau_i;\mathbf{b}_{h-1,x_i^*}^{u,w}\right)d\tau_i$ is the probability that the i^{th} failure occurs at time τ_i where $i=(u+1),\cdots,(h-1)$ and $\tau_i\in(T,\tau_{i+1})$; $f_{x_h^*}\left(\tau_h;\mathbf{b}_{h-1,x_h^*}^{u,w}\right)$ is the conditional probability density that the h^{th} failure occurs at τ_h given the i^{th} failure occurs at time τ_i where $i=(u+1),\cdots,(h-1)$; $\prod_{y^*\in Y_{h-1}^{u,w}} R_{y^*}\left(\tau_h;\mathbf{b}_{h-1,y^*}^{u,w}\right)$ and $\prod_{z^*\in G_{h-1}^{u,w}-\left\{x_h^*\right\}} R_{z^*}\left(\tau_h;\mathbf{b}_{h-1,z^*}^{u,w}\right)$ are the conditional probabilities that standby pairs in $Y_{h-1}^{u,w}$ and operating pairs in $G_{h-1}^{u,w}-\left\{x_h^*\right\}$ survive time τ_h given the i^{th} failure occurs at time τ_i where $i=(u+1),\cdots,(h-1)$.

The conditional *pdf* is

$$g_h^u\left(\tau_h\,|\,T\right)=g_h^u\left(\tau_h\right)/R_{sys}\left(T\right) \tag{5.42}$$

5.5.3 *Application of k-out-of-n Pairs:G Balanced Systems in UAV Systems*

In the Introduction, we present three applications of k-out-of-n pairs:G Balanced systems. In this section, we demonstrate the application of the proposed model and reliability estimation procedure in practice using UAV systems as an example. Consider an octocopter, which is an UAV with four pairs of rotors distributed evenly on a circle, its function depends on the balance of the rotor pairs. To evaluate the reliability of the octocopter in real-time, we monitor the vibration, current, or voltage of each rotor as a degradation indicator. We also monitor temperature and humidity of the operating

conditions. We obtain the reliability of each rotor and hence the reliability of each pair by fitting our degradation model to the real-time data.

We introduce the reliability estimation procedure based on the balance requirement that the operating pairs should be symmetric w.r.t. at least a pair of perpendicular axes. An UAV may have additional balance requirements due to the spatial configuration of rotors with different rotational directions. But this does not affect the application of our method in this case. We first develop a heuristic for determining the balance of the system and the operating pairs to be forced down into standby or the standby pairs to resume operation when the system is unbalanced. Then we enumerate all the successful events using the proposed procedure and the heuristic developed specifically for the octocopter system. Then we estimate system reliability and probability density of time to an ordered failure based on the derived equations.

5.6 Maximum Likelihood Parameter Estimation

Suppose we have observations of degradation values $D_i\left(t_j\right)$ and operating condition $\mathbf{S}_i\left(t_j\right)=\left\{S_{il}\left(t_j\right)\right\}$ for unit i at discrete time t_j where $j=1,2,\cdots,m$ if there are m observations by time T. Let $t_0=0$ and $D_i\left(t_0\right)=x_i^0$, and $\mu_i\left(\tau\right)=\mu_0\left(\tau;\boldsymbol{\Omega}\right)A\left(\mathbf{S}_i\left(\tau\right);\boldsymbol{\Theta}\right)$ has the parameters $\boldsymbol{\Omega}$ and $\boldsymbol{\Theta}$. We assume that the operating conditions between time $\left(t_{j-1},t_j\right]$ can be approximated by its value at t_j. In addition, we assume that the forcing-down and resumption of units occur at t_j. Then the likelihood function is obtained as

$$L\left(D_i,\mathbf{S}_i,\rho_{i\bullet},\delta_{i\bullet}|\boldsymbol{\Theta},\boldsymbol{\Omega},\sigma_i\right)$$
$$=\prod_{i=1}^{2n}\prod_{j=1}^{m}\left\{\begin{array}{l}\dfrac{1}{\sigma_i\sqrt{\int_{t_{j-1}-\delta_{i,j-1}}^{t_j-\delta_{i,j-1}}\mu_0\left(\tau;\boldsymbol{\Omega}\right)A\left(\mathbf{S}_i\left(t_j\right);\boldsymbol{\Theta}\right)d\tau}}\times \\ \phi\left(\dfrac{\left(D_i\left(t_j\right)-D_i\left(t_{j-1}\right)\right)-\int_{t_{j-1}-\delta_{i,j-1}}^{t_j-\delta_{i,j-1}}\mu_0\left(\tau;\boldsymbol{\Omega}\right)A\left(\mathbf{S}_i\left(t_j\right);\boldsymbol{\Theta}\right)d\tau}{\sigma_i\sqrt{\int_{t_{j-1}-\delta_{i,j-1}}^{t_j-\delta_{i,j-1}}\mu_0\left(\tau;\boldsymbol{\Omega}\right)A\left(\mathbf{S}_i\left(t_j\right);\boldsymbol{\Theta}\right)d\tau}}\right)\end{array}\right\}^{\rho_{ij}\omega_i} \tag{5.43}$$

where $\rho_{ij}=1$ if unit i is operating during time interval $\left(t_{j-1},t_j\right]$, 0 otherwise; ω_i is the weight for the degradation observations of individual unit i; and $\delta_{i,j-1}$ is the length of time during which unit i is in standby and not subject to degradation by time t_{j-1}. Note that when $\rho_{ij}=0$, we can just assign any real value to the part in the braces. The degradation observations of an individual unit should have a lower contribution to the likelihood function after it has failed by the observation time. Meanwhile, the parameters of the degradation model should reflect the properties of units that survive up to the observation time, namely operating units and standby units, because system reliability is estimated based on the reliabilities of such units. We then assign weights to the degradation observations of individual units. In particular, a lower weight is assigned to the failed units and a higher weight to the operating units and the standby units. Here, we set $\omega_i=0.8$ if unit i survives observation time and 0.2 otherwise.

Maximizing Eq. (5.43) is equivalent to minimizing Eq. (5.44).

$$l\left(D_i,\mathbf{S}_i,\rho_{i\bullet},\delta_{i\bullet}|\boldsymbol{\Theta},\boldsymbol{\Omega},\sigma_i\right)$$
$$=\sum_{i=1}^{2n}\sum_{j=1}^{m}\rho_{ij}\omega_i\left\{\begin{array}{l}2\ln\sigma_i+\ln\left(\int_{t_{j-1}-\delta_{i,j-1}}^{t_j-\delta_{i,j-1}}\mu_0\left(\tau;\boldsymbol{\Omega}\right)A\left(\mathbf{S}_i\left(t_j\right);\boldsymbol{\Theta}\right)d\tau\right)\\+\dfrac{\left(\left(D_i\left(t_j\right)-D_i\left(t_{j-1}\right)\right)-\int_{t_{j-1}-\delta_{i,j-1}}^{t_j-\delta_{i,j-1}}\mu_0\left(\tau;\boldsymbol{\Omega}\right)A\left(\mathbf{S}_i\left(t_j\right);\boldsymbol{\Theta}\right)d\tau\right)^2}{\sigma_i^2\int_{t_{j-1}-\delta_{i,j-1}}^{t_j-\delta_{i,j-1}}\mu_0\left(\tau;\boldsymbol{\Omega}\right)A\left(\mathbf{S}_i\left(t_j\right);\boldsymbol{\Theta}\right)d\tau}\end{array}\right\} \quad (5.44)$$

Taking the derivative of $l\left(D_i,\mathbf{S}_i,\rho_{i\bullet},\delta_{i\bullet}|\boldsymbol{\Theta},\boldsymbol{\Omega},\sigma_i\right)$ w.r.t. σ_i and equating the resultant equations to zero, we obtain

$$\hat{\sigma}_i^2=\frac{1}{\sum_{j=1}^{m}\rho_{ij}\omega_i}\sum_{j=1}^{m}\left\{\rho_{ij}\omega_i\frac{\left(\left(D_i\left(t_j\right)-D_i\left(t_{j-1}\right)\right)-\int_{t_{j-1}-\delta_{i,j-1}}^{t_j-\delta_{i,j-1}}\mu_0\left(\tau;\boldsymbol{\Omega}\right)A\left(\mathbf{S}_i\left(t_j\right);\boldsymbol{\Theta}\right)d\tau\right)^2}{\int_{t_{j-1}-\delta_{i,j-1}}^{t_j-\delta_{i,j-1}}\mu_0\left(\tau;\boldsymbol{\Omega}\right)A\left(\mathbf{S}_i\left(t_j\right);\boldsymbol{\Theta}\right)d\tau}\right\} \quad (5.45)$$

Now we can substitute this for σ_i^2 in Eq. (5.44) so that we have only $\boldsymbol{\Theta}$ and $\boldsymbol{\Omega}$ to estimate.

5.7 Numerical Example

In this section, we present a numerical example for the 1-out-of-6 pairs:G Balanced system. Consider the degradation model in Eq. (5.1). Suppose we only consider the temperature effect on the units. Let the baseline degradation rate $\mu_0\left(t\right)=\alpha t^{\beta}$ and acceleration factor $A\left(Temp_i\left(t\right)\right)=C\exp\left[e/Temp_0-e/Temp_i\left(t\right)\right]$ where $Temp_i\left(t\right)$ is the operating temperature of unit i at time t, $Temp_0$ is the baseline operating temperature expressed in Kelvin and *e* is a scale parameter. Note that

$$\begin{aligned}\mu_i(t) &= \mu_0(t)\,A\big(Temp_i(t)\big) \\ &= \alpha t^{\beta}\cdot C\exp\left[\frac{e}{Temp_0}-\frac{e}{Temp_i(t)}\right] \\ &= \left[\alpha C\exp\left(\frac{e}{Temp_0}\right)\right]t^{\beta}\exp\left[-\frac{e}{Temp_i(t)}\right]\end{aligned} \tag{5.46}$$

where $\left[\alpha C\exp\left(e/Temp_0\right)\right]$ is constant, which we denote as α'. We re-parameterize the model as follows to simplify the parameter estimation.

$$\mu_i(t) = \alpha' t^{\beta}\exp\left[-\frac{e}{Temp_i(t)}\right] \tag{5.47}$$

Let $\alpha' = 2$, $\beta = 0.3$, $\sigma_i \equiv 1$, $x_i^0 \equiv 0$, $h_i \equiv 100$ and $e = 400$. In addition, we set the design working temperature $Temp_0$=293K. We generate temperature data, $Temp_i(t)$, as shown in Figure 5.2 by considering an average temperature of 293K. The temperature data are generated in a way so that any two adjacent units according to Figure 1.1(a) have similar temperature values.

The degradation paths are generated by a Gamma process, as shown in Figure 5.3. The increments of the Gamma process in $(t-\Delta t, t]$ follow a Gamma distribution with a scale parameter σ_i and a shape parameter $\left(M_i(t)-M_i(t-\Delta t)\right)$. Note that because $\sigma_i = 1$, the degradation increments generated by the Gamma distribution have same mean and variance as the normally distributed degradation increments as specified in Eq. (5.2). Hence the generated degradation paths can be modeled with Eq. (5.1). The degradation value of a unit

is constant when the unit is in standby, e.g. the degradation paths of units 4 and 10 are flat between times 66 and 73 during which they are in standby; when the unit fails, e.g. unit 1 has a flat degradation path after it fails at time 52; or when the other unit of a pair fails, e.g. unit 7 has a flat degradation path after unit 1 fails at 52.

The states of individual units are shown in Figure 5.4. The state of a unit is 1 if it is operating, –1 if it is in standby, and 0 if it is failed or the other unit in the same pair has failed. In addition, we observe the times when units fail, are forced down or resume operation. If unit i fails at τ_j, then we mark τ_j in the subplot for unit i but not in the subplot of unit $(i+n)$ or unit $(i-n)$; and if unit i is forced down into standby or resumes operation, we mark the corresponding time in its subplot.

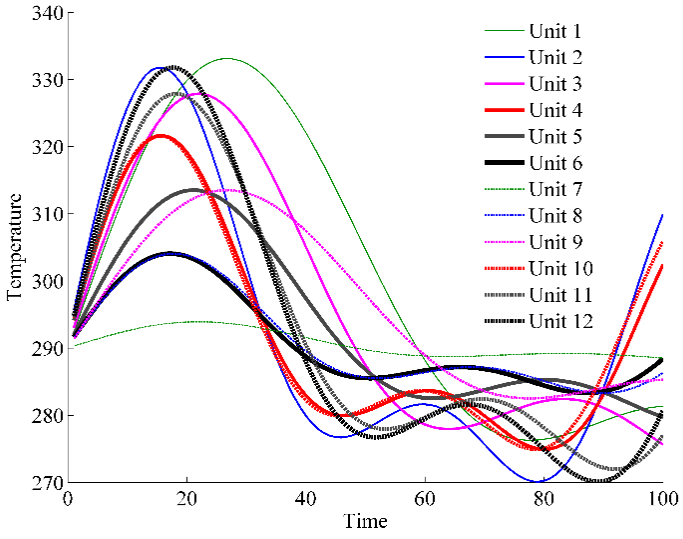


Figure 5.2 Temperature profile of the operating environment

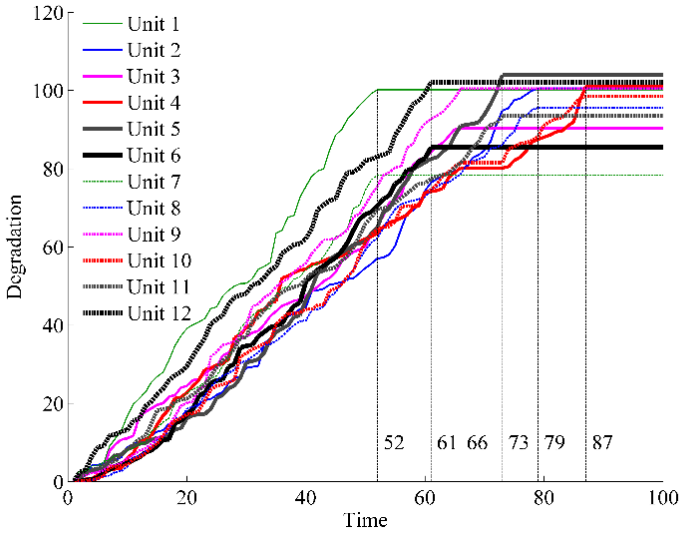


Figure 5.3 Degradation paths of individual units and the failure times

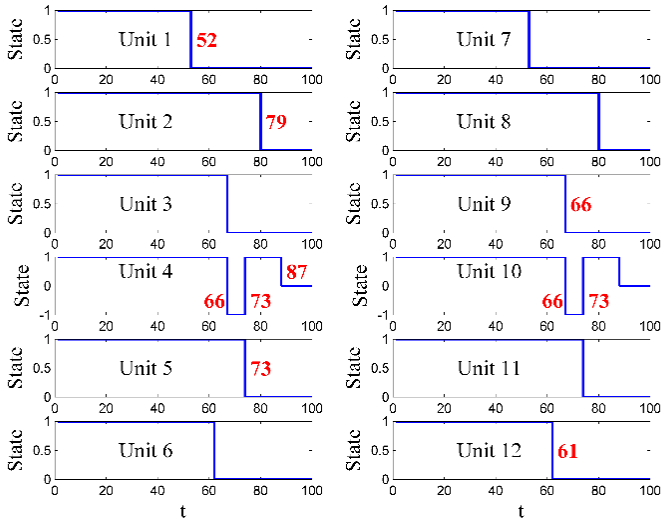


Figure 5.4 States of individual units over time

5.7.1 *Reliability Metric Estimation*

First, we predict the system reliability and *pdf* of the time to the h^{th} failure when the system starts at $t = 0$. Because we have no degradation observations or temperature measurements available, we let the parameters of the degradation model have their theoretical values, i.e. the values used for data generation, and let the operating temperature be $Temp_0$. The plots of system reliability and *pdf* are shown in Figure 5.5 and Figure 5.6, respectively. Note that 'τ_i bar' means $\bar{\tau}_i$, i.e. the actual time when the i^{th} failure occurs, in the figure legend. As shown in Figure 5.6, the estimation of *pdf* of time to the h^{th} failure based on theoretical values of degradation parameters does not accurately reflect the actual failure times. This is to be expected since the theoretical degradation parameter values only reflect the statistical properties of units from a large population under the design operating condition.

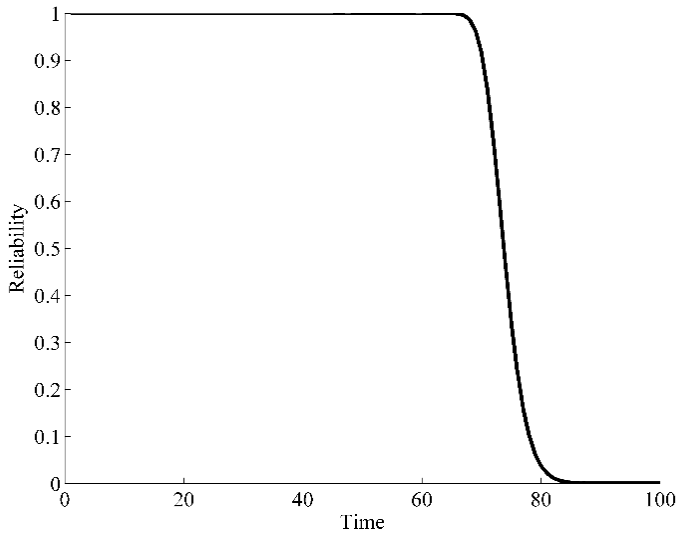


Figure 5.5 System reliability

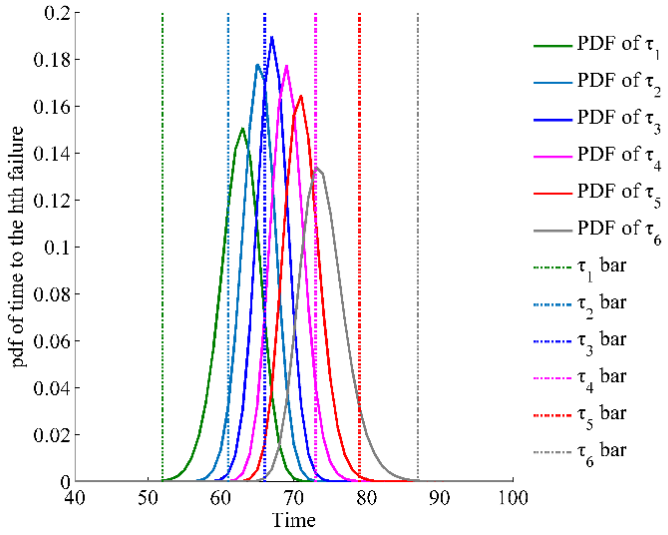


Figure 5.6 The *pdf* of time to the h^{th} failure

Then we estimate system reliability and conditional *pdf* of the time to the h^{th} failure at different observation times. System reliability plots are shown in Figure 5.7 where the dashed line with asterisks is the real-time system reliability estimate which is updated when we have more degradation observations and when an additional failure occurs.

We estimate the system residual life, including the mean and its 0.95 confidence interval, based on conditional system reliability, as shown in Figure 5.8. The actual value of the residual life is a naïve inference from the actual system failure time by assuming the residual life is linearly decreasing. The system residual life may not be a good indicator of a system's condition in the early stage due to the small number of failed pairs. Therefore, we estimate the time to the h^{th} failure instead.

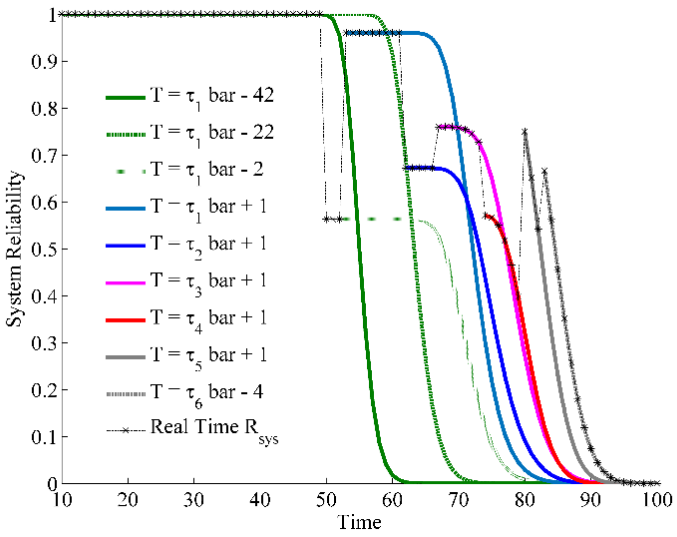


Figure 5.7 System reliability plots at different observation times

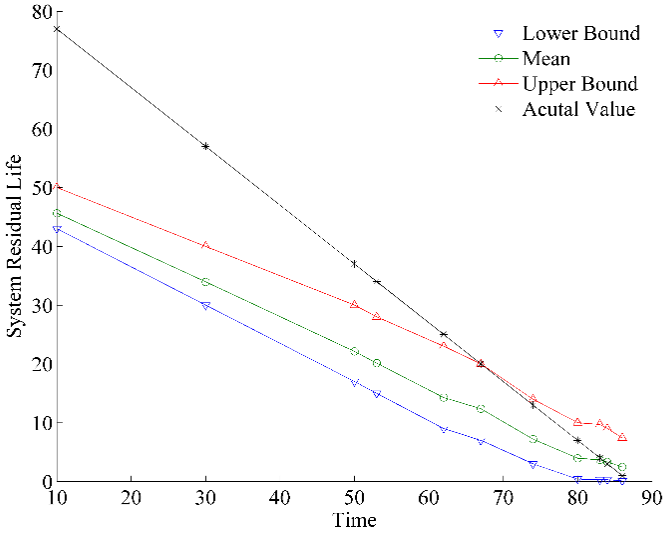


Figure 5.8 System residual life

Finally, we estimate the conditional *pdf* of the time to the h^{th} failure when $(h-1)$ failures occur. The conditional *pdf*s of the times to the h^{th} failure at different observation times are shown in Figure 5.9. Based on the *pdf*s, the mean times to the h^{th} failure ($h = 1$ to 6), and

their 0.95 confidence intervals are also estimated at different observation times, i.e. $(\bar{\tau}_1 - 22)$, $(\bar{\tau}_1 + 1)$, $(\bar{\tau}_2 + 1)$, $(\bar{\tau}_3 + 1)$, $(\bar{\tau}_4 + 1)$, and $(\bar{\tau}_6 - 4)$, as shown in Figure 5.10. Note that the time to the h^{th} failure is from time $t = 0$, instead of from the $(h-1)^{\text{st}}$ failure. The estimates of mean times to the h^{th} failure approach the actual values as time elapses. In addition, the 0.95 confidence intervals of the times to the h^{th} failures are narrow enough by noticing that they do not cover the mean times to the $(h+1)^{\text{st}}$ failures.

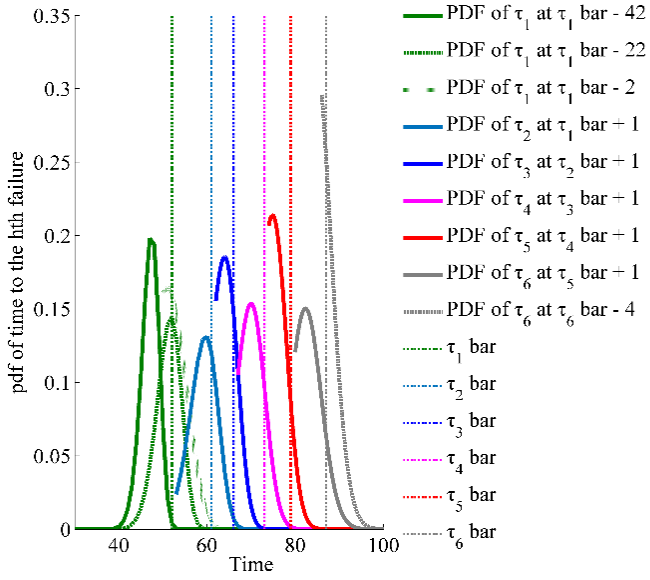


Figure 5.9 Conditional *pdf* of time to the h^{th} failure where h = 1 to 6

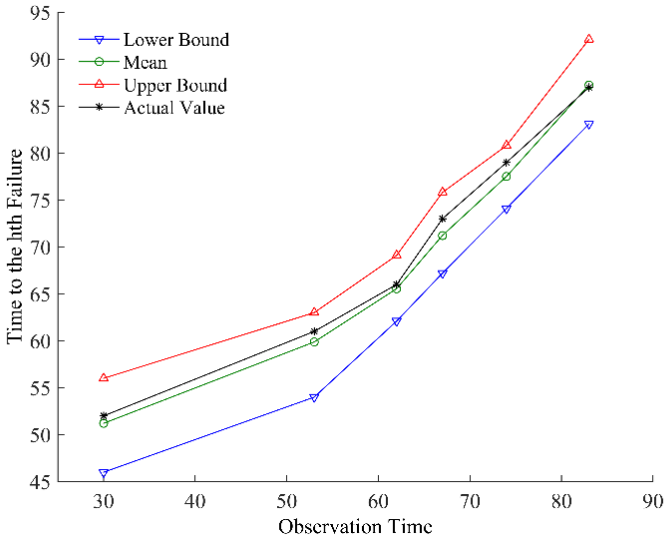


Figure 5.10 Time to the h^{th} failure where h = 1 to 6

5.7.2 *Reliability Computation*

The estimation of reliability metrics requires integration of complex equations. We discuss the computational time and accuracy of the integration in this section. The reliability estimate in this example is obtained using Matlab 2013a running on a desktop computer with an Intel(R) Core(TM) i7-3770S CPU, 16GB RAM, and 64-bit Windows 7 operating system. In general, a system with a large n and small k tends to have more successful events and more state transitions, which results in more complicated integration and hence higher computational time. The accuracy of the reliability estimation depends on the algorithm used for carrying out the integration. Matlab provides functions for calculating the numerical values of one to three-dimensional integrations with high accuracy.

Consider the computation time for system reliability when the system is new, as shown in Figure 5.5. This requires the most integration processes since, on one hand, it involves the

most successful events, and on the other hand, it considers the most time instants. The total computation time is approximately one hour. It involves 10800 single integrations and 21600 double integrations. The average computation time of a single integration is 0.0189 second, and 0.1722 second for a double integration.

We also investigate the increment of computation time with the dimension of integration by carrying out a simulation study as follows. In each simulation run we first carry out ten single integrations in the form of $\int_{\tau_1=1}^{t} f_{i^*}(\tau_1) R_{i^*}(\tau_1) d\tau_1$, ten double integrations in the form of $\int_{\tau_2=1}^{t}\int_{\tau_1=1}^{\tau_2}\prod_{j=1}^{2} f_{i^*}(\tau_j) R_{i^*}(\tau_j) d\tau_j$, and ten triple integrations in the form of $\int_{\tau_3=1}^{t}\int_{\tau_2=1}^{\tau_3}\int_{\tau_1=1}^{\tau_2}\prod_{j=1}^{3} f_{i^*}(\tau_j) R_{i^*}(\tau_j) d\tau_j$, where $i^* \in \{1^*, 2^*, \cdots, 6^*\}$ and t are generated randomly. Then we obtain the average computation times for the single, double and triple integrations. We run the simulation for ten times. We plot the average computation times in Figure 5.11 where we observe that the average integration computation time (in seconds) is exponentially increasing with the dimension of integration. For many practical applications n is relatively small and k is relatively large, and the dimensions of integrations can be reduced by considering the relationship between *pdf* and reliability function, thus the computation burden is insignificant. For example, the double integration $\int_{\tau_2=T}^{t}\int_{\tau_1=T}^{\tau_2} f_{i^*}(\tau_1) f_{j^*}(\tau_2) d\tau_1 d\tau_2$ can be reduced into a single integration $\int_{\tau_2=T}^{t}\left[R_{i^*}(T) - R_{i^*}(\tau_2)\right] f_{j^*}(\tau_2) d\tau_2$.

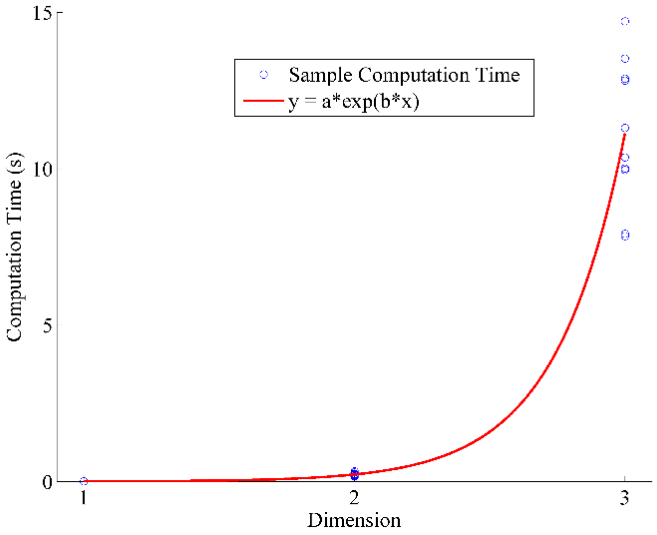


Figure 5.11 Computation time for integrations of different dimensions

5.8 Conclusions

Systems with spatially distributed units have emerged in many applications. The reliability estimation of such systems considering the degradation paths of individual units is challenging. In this chapter, we investigate 1-out-of-6 pairs:G Balanced system with all units performing the same function, which is a complex system with spatially distributed units. The system reliability and conditional *pdf* of time to the h^{th} failure are estimated by considering the degradation paths of individual units which are affected by different operating conditions. In addition, we update the estimation when system state changes and more degradation observations become available. A numerical example shows that the estimates for reliability metrics approach actual values as time elapses.

We also generalize the degradation model for individual units considering operating conditions and the procedure for estimating system reliability metrics. The generalized

model and procedure can be used for any k-out-of-n pairs:G Balanced systems when predicting reliability metrics when the system is new and actual degradation observations are available; and when updating observable reliability metrics after some failures have occurred or when more degradation observations become available.

CHAPTER 6

RELIABILITY ESTIMATION CONSIDERING MULTI-STATE UNITS

6.1 Problem Definition and Assumptions

In many cases, systems are required to provide a specified capacity such as the case of power distribution systems. A system fails when its capacity does not meet the required minimum capacity. The capacities of the systems are the sum of the capacities of individual units, e.g., several engines collectively generating certain horsepower, or generators with a certain output voltage. In the previous chapters, the capacities of units are considered as equal and are ignored in reliability estimation. For example, in an UAV that consists of identical rotors that provide the same lift power. However, the capacities of units can decrease and vary from each other in some cases and thus should be considered in reliability estimation.

In this chapter, we investigate the reliability estimation of weighted-c-out-of-n pairs:G Balanced system, which is a variant of k-out-of-n pairs:G Balanced system. In the weighted-c-out-of-n pairs:G Balanced system, we have n pairs of units distributed evenly on a circle as in the k-out-of-n pairs:G Balanced systems. We consider the capacity of each unit. The capacities of individual units can decrease from a level to a lower level. The system requires at least a minimum capacity c to function while maintaining balance.

6.1.1 *Assumptions*

In this chapter, we assume the following:

- The units in the system are identical and have the same capacity at the start time of the system's operation.
- The capacity of a unit has multiple levels, i.e. levels $m,(m-1),\cdots,1,0$ where m is the highest level and 0 is the lowest level (failure).
- The capacity of a unit decreases discretely and may decrease more than one level at each instant. For example, the capacity can decrease from level m to level $(m-1)$, and then from level $(m-1)$ to level $(m-3)$.
- The transition times from one capacity level to another follow independent exponential distributions.

6.1.2 *Definitions*

Actual capacity: the maximum capacity that a unit or pair can provide. When two units of the same pair have different maximum capacities due to different degrees of degradation, their actual capacities are the lower maximum capacity of the two units. The additional capacity of either unit in the same pair is forced down permanently. Therefore, the actual capacity of a pair is twice the actual capacity of one of the units.

Serving capacity: the working capacity that a unit or pair actually provides to the system. The serving capacity of a unit or pair is always lower than or equal to its actual capacity.

In some cases, we force down a portion of the actual capacity of a pair to rebalance the system. In such cases, the serving capacity is the remaining portion of the actual capacity.

6.2 System Description

6.2.1 *General System Configuration*

A weighted-c-out-of-n pairs:G Balanced system has n pairs of units distributed evenly on a circle as in k-out-of-n pairs:G Balanced systems. All units have the same initial capacity. The capacity of any unit has $(m+1)$ levels: $\{m, m-1, \ldots, 1, 0\}$ where m represents maximum capacity level and 0 represents failure. The capacity of any unit decreases from one level to another by one or more levels at each change instant.

A minimum capacity c is required to maintain the system's function. The system must be balanced in the sense that the operating units with different capacity levels should be symmetric w.r.t. at least one common pair of perpendicular axes of symmetry.

6.2.2 *System Balance*

As we introduced in Chapter 3, to keep the system balanced, the two units in any pair must be in the same state. Therefore, we assume that whenever the capacity (actual or serving) of a unit changes, the capacity (actual or serving) of the other unit of the same pair changes to the same level instantaneously.

Consider a system with six pairs of units where any pair has three capacity levels as shown in Figure 6.1. In this chapter we use white circles to represent units with level 2 actual capacity and level 2 serving capacity, white triangles to represent units with level 1 actual capacity and level 1 serving capacity, black circles to represent failed units (units of level 0 actual capacity), and blue bold lines to represent a common pair of perpendicular axes of symmetry for operating units with different serving capacities

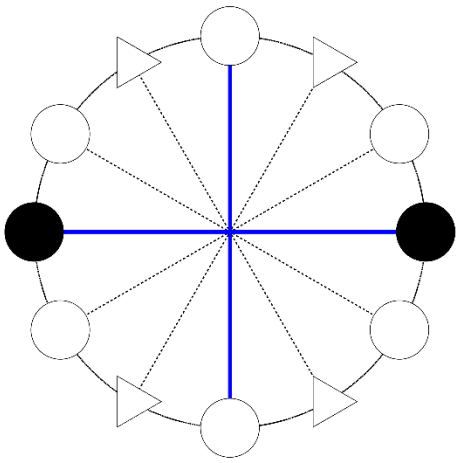
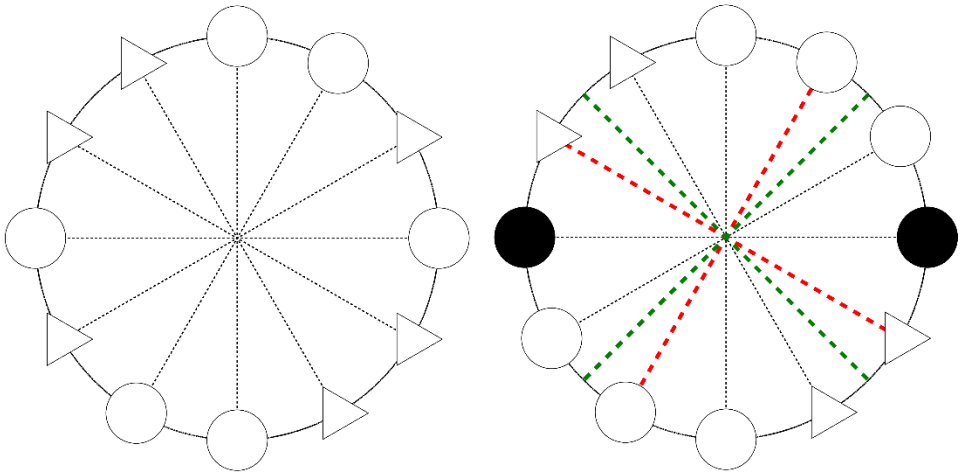


Figure 6.1 Example of a balanced system

When the system is unbalanced, we rebalance the system by switching the serving capacities of some pairs to lower levels. Note that when the serving capacity of a pair is switched to a lower level, it can switch to any higher level within the actual capacity later if it is necessary for rebalancing the system. For example, suppose a pair has a level 4 actual capacity and its serving capacity switches from level 4 to level 3. The serving capacity can recover to level 4 if necessary, unless the actual capacity decreases to levels 2 or lower.

Again, consider a system with six pairs of units where any pair has three capacity levels. Figure 6.2 shows two unbalanced systems. The system in Figure 6.2(a) is unbalanced since there does not exist any axis of symmetry for the operating units. The system in Figure 6.2(b) is also unbalanced since the operating units in white circles and those in white triangles are not symmetric w.r.t. a common pair of perpendicular axes. Even though the operating units in white circles (triangles) are symmetric w.r.t. the red (green) axes, the two axes pairs are not the same. Therefore, the system is unbalanced.

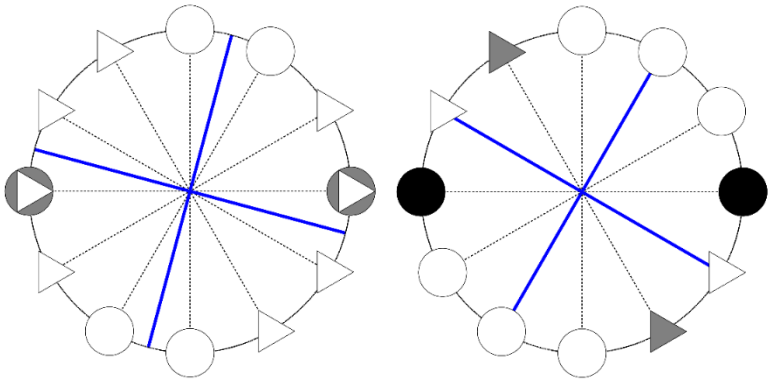


(a) (b)

Figure 6.2 Examples of unbalanced systems

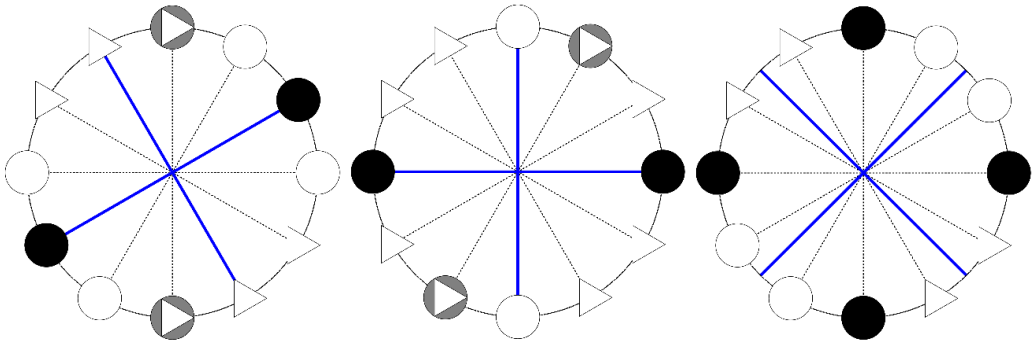
We can rebalance the systems in Figure 6.2 as shown in Figure 6.3. The system in Figure 6.2(a) is rebalanced into the system in Figure 6.3(a) by switching the serving capacity of a pair from level 2 to level 1, as shown by a circle in gray containing a triangle in white. Note that when additional failures occur the serving capacity of this pair can switch back to level 2 if it is necessary for rebalancing the system, as shown in Figure 6.4(a), or fail, as shown

in Figure 6.4(b). The system in Figure 6.2(b) is rebalanced into the system in Figure 6.3(b) by switching the serving capacity of a pair from level 1 to level 0, as shown by a triangle in gray. Similarly, the serving capacity of this pair can switch back to level 1, as shown in Figure 6.4(c). However, it does not fail during the time when its serving capacity is at level 0.



(a) (b)

Figure 6.3 Examples of rebalancing unbalanced systems



(a) (b) (c)

Figure 6.4 Examples of capacity recovery and loss

6.3 Axis of Symmetry Algorithm

For this system, due to a large number of states of the pairs, we enumerate all the candidate axes. We start the enumeration from an arbitrary axis. Any candidate axis is either along a pair or in the middle of two pairs. There are n unique pairs of perpendicular axes. Starting from an arbitrary candidate axis, we enumerate n axes consecutively. We ignore their perpendicular axes in this case since they will result in the same information.

For each candidate axis, we compare the units on the two sides of the axis one by one. We only consider the units within $\pm\pi/2$ around the axis. The states of the other half of the units are determined by considering the opposite units in the same pairs. Each unit is compared with the symmetric unit w.r.t. the candidate axis. If any two units have the same actual capacity, switch their serving capacities to the actual capacity if the serving capacities are lower than the actual capacity; if they have different actual capacities, switch their serving capacities to the lower actual capacity value. For each pair of compared units, we determine their capacity difference: the difference between the two units' actual capacities. The total capacity difference is the sum of the capacity differences of all the compared units.

When the total capacity difference is zero, the system is balanced without switching the serving capacity of any unit into a level lower than its actual capacity. When there exists such a candidate axis, then it is the axis of the symmetry of the system. Otherwise, we find the one that corresponds to the lowest total capacity difference as the axis of the symmetry. We break ties by using other criteria.

6.4 Successful Event Enumeration

6.4.1 *System Modeling*

The state of the system is modeled as a continuous-time Markov chain (CTMC) since we assume that the transition times from one capacity level to another follow independent exponential distributions. We let the states of the failed systems be absorbing states, and let the successful system states be transient states. The system reliability at any time is the probability that the system is in a transient state. Let c_x be the level x capacity of any pair. We model the state of any pair with a two-element vector (i, j) when its actual capacity is c_i and serving capacity is c_j, where $c_m \geq c_i \geq c_j \geq c_0$. Any pair has $\frac{(m+2)(m+1)}{2}$ possible states since j can be 0 to i for any $i \in \{0,1,\ldots,m\}$.

Generally, the state of any pair has two possible transitions: transition due to the actual capacity decrease and transition due to the serving capacity switch. The first type of transition can be modeled by a Markov chain. The second type of transition always occurs simultaneously with the first type. To model the second type of transition, we find the necessary switch by using the algorithm proposed in Section 6.3 as the first type transition occurs. Numerically, since the state of a pair is modeled with a two-element vector (i, j), the first (second) type of transition is represented by the change in the first (second) element of the vector.

The system state is any combination of the states of the n pairs and is modeled as a vector that contains all the states of the pairs in order. For instance, the system state shown in Figure 6.3(b) is [(0,0), (2,2), (2,2), (2,2), (1,0), (1,1)] where (0,0) represents that the first pair is failed, (2,2) represents that the next three pairs have level 2 actual capacity, i.e. full capacity, and operating with full capacity, (1,0) represents that the 5^{th} pair has level 1 actual capacity though it is forced down for system balance and consequently not operating (0 serving capacity), and (1,1) represents that the last pair has level 1 actual capacity and is operating at level 1 capacity. For another instance, the system shown in Figure 6.4(a) is [(2,2), (0,0), (2,2), (2,1), (1,1), (1,1)] where (2,1) represents that the 4^{th} pair has level 2 actual capacity but a part of its capacity is forced down for system balance which results in a level 1 serving capacity.

When all the units are identical, it is possible to aggregate multiple system states into one category when they are the repetition of each other. In this dissertation, to determine if a row vector $\mathbf{b}$, e.g. a system state, is a repetition of another row vector $\mathbf{a}$, we compare them by searching $\mathbf{a}$ in the vector of $(\mathbf{b},\mathbf{b})$. If $\mathbf{a}$ can be found in $(\mathbf{b},\mathbf{b})$, then $\mathbf{a}$ is a repetition of $\mathbf{b}$. In some cases, two system states are repetition to each other if either one of them flips in the left-right direction. In these cases, the two system states are also aggregated into one category. Consider a system state $\mathbf{s}_1 = [(1,1)\ (1,1)\ (0,0)\ (2,1)]$. System state $\mathbf{s}_2 = [(1,1)\ (2,1)\ (0,0)\ (1,1)]$ is its left-right repetition. This is because the left-fight flip of the $\mathbf{s}_2$ is [(1,1) (0,0) (2,1) (1,1)], which is a repetition of $\mathbf{s}_1$.

6.4.2 *Event Enumeration*

To enumerate successful events, we build a system state transition diagram as in Chapter 3. Due to the multiple levels of pair capacity, the state of any system has more than one transition direction since each system state transition is due to the capacity reduction of a pair. We model the system state transition using an m-dimensional coordinate system. The i^{th} dimension of the coordinate system corresponds to a capacity level $(m-i)$ where $i \in \{1,\ldots,m\}$. The value of the i^{th} dimension is the number of pairs that have an actual capacity level $(m-i)$, which we denote as u_i. Each coordinate represents all the system states that have u_i pairs that have an actual capacity level $(m-i)$. The origin of the coordinate system represents the original system state with all pairs of full capacity (capacity level m). Apparently we only consider the origin and the non-negative integer space of the coordinate system. It is immediate that $\sum_{l=1}^{m} u_l \le n$. Let u_0 denote the number of pairs that have an actual capacity level m, it follows that $\sum_{l=0}^{m} u_l = n$.

We use an m-element vector to denote all the possible system state transition directions when any pair has $(m+1)$ capacity levels. Each element in the vector can have any value in the range {1, 0, –1}. When one pair in the system has its actual capacity level reduced to level $(m-i)$ from level $(m-j)$, the i^{th} element takes value 1 and the j^{th} element takes value –1 since the system has one more pair of the level $(m-i)$ actual capacity and, at the same time, one less pair of the level $(m-j)$ actual capacity. Note when $j=0$ there is no

element of value −1. The i^{th} element takes value 0 when the transition does not involve any pair of an actual capacity level $(m-i)$. Since we assume that multiple reductions in actual capacity cannot occur at the same time, any state transition direction vector has one and only one element of value 1. Moreover, it must have one element of value −1 unless the actual capacity of the pair is reduced from the highest level, i.e. level m. We sort the elements in the vector in the descending order of the capacity levels. Then the element of value −1 must precede the element of value 1. It is immediate that there exist $m+(m-1)+\ldots+1=\frac{m(m+1)}{2}$ possible directions.

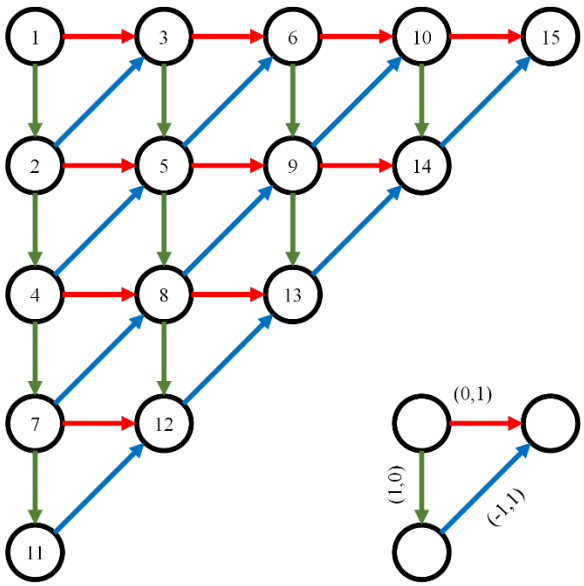


Figure 6.5 Coordinate system and system state transition directions when $m=2$

For example, as shown in Figure 6.5 when any pair has three capacity levels, i.e. $m=2$, we use vector (1, 0), (0, 1) and (−1, 1) to denote the directions of transition where the actual capacity of any pair decreases from level 2 to level 1, from level 2 to level 0, and from

level 1 to level 0, respectively. Also, as shown in Figure 6.6 when $m = 3$, we use vector (1, 0, 0) and (–1, 1, 0) to denote the directions of transition where the actual capacity of any pair decreases from level 3 to level 2 and from capacity level 2 to level 1, respectively.

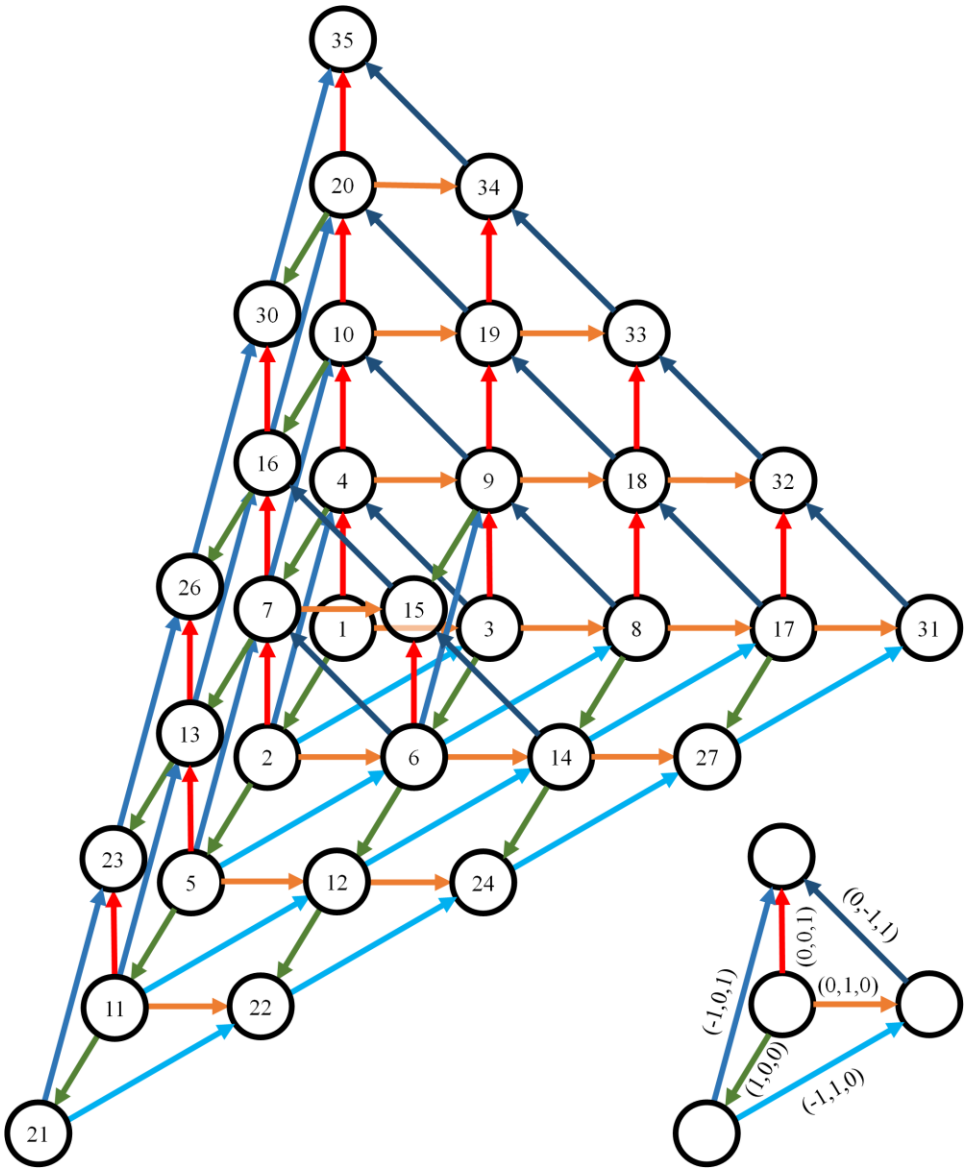


Figure 6.6 Coordinate system and system state transition directions when $m = 3$

6.4.3 *State Transition Diagram*

We now introduce the algorithm for enumerating system states and transitions. The main idea of the algorithm is to enumerate the system states according to an m-dimensional coordinate system. In the remainder of this chapter, we use brackets to represent subsets: when a is a vector, then $a[i]$ is the i^{th} element in a; when A is a set of vectors (or a set of sets), then $A[i][j]$ is the j^{th} element of the i^{th} vector (set) in A.

For each coordinate v, a transition direction d is valid when

- There exists element $d[i]=1$ for one and only one i.
- There exists element $d[j]=-1$ for at most one j and $j<i$.
- Coordinate $v_f=(v+d)$ does not have any negative elements.
- Coordinate $v_f=(v+d)$ results in any system state with a capacity greater than c.

We define a coordinate v_f as a follow-up coordinate of another coordinate v when v_f can be reached from v via a valid transition direction. Also, we define v as a preceding coordinate of v_f. Similarly we define a system state S_f as a follow-up state of a state S and S as the preceding state of S_f when S_f can be reached from S via a valid transition direction.

Starting from the origin and the initial system state, we enumerate all the possible follow-up coordinates and corresponding system states via all the valid transition directions. Generally, for each coordinate, we enumerate all of its follow-up coordinates via all

possible valid transition directions. Simultaneously, we enumerate all of the possible transitions between the system states corresponding to the current coordinate and their follow-up states corresponding to the follow-up coordinates. This procedure is completed when all successful states and the transitions between them are enumerated.

Consider the 2-dimensional coordinate system in Figure 6.5. The first dimension is vertical and the second dimension is horizontal. The corresponding three state transition directions are also as shown in the right bottom corner of Figure 6.5. Each node in Figure 6.5 is a coordinate. The arrows between the nodes represent valid transition directions. For example, node 1 is origin (0, 0), node 2 is (1, 0), node 3 is (0, 1), and node 5 is (1, 1). The transition from node 1 to node 2 is along the direction of (1, 0). When we denote the complete set of coordinates and transition directions as V and D, $V[1]=(0, 0)$, $V[2]=(1, 0)$, $D[1]=(1, 0)$, the transition can be modeled as $V[2]=V[1]+D[1]$.

Consider the 3-dimensional coordinate system in Figure 6.6. The first dimension is along nodes 1 to 2, the second one is along nodes 1 to 3, and the third one is along nodes 1 to 4. There are five transition directions as shown in the bottom right corner of Figure 6.6.

6.4.3.1 *State Transition Diagram Example*

Figure 6.7 shows a system state transition diagram for a system where any pair has three capacity levels, 2, 1 and 0. In Figure 6.7, any system state is a representative of a category of system states where a system is either a repetition or a left-right repetition of the other

states. The transitions between the system states are shown as arrows. The number of possible transitions between two states is annotated on the corresponding arrow unless there is only one possible transition. As in Figure 6.5, we use different colors to represent different transition directions.

We explain several transition paths in Figure 6.7 and the remaining paths can be explained similarly. From the initial state (state 0), there are two possible transition paths: The capacity of any pair drops from level 2 to level 1 via transition direction (1, 0) and from level 2 to level 0 via transition direction (0, 1). The two transition paths lead to states 1 and 2, respectively. From state 1, we have five possible transition paths via three possible transition directions: The first two paths are from state 1 to states 3 and 4, respectively, via transition direction (1, 0) that leads to one more pair with capacity level 1 and one less pair with capacity level 2. State 3 represents all states that have only two consecutive pairs with level 1 capacities. There are two possible ways to transition from state 1 to state 3 since in state 1 there are two pairs with level 2 capacity next to the pair with level 1 capacity. There is only one possible way to transition from state 1 to state 4 since in state 1 there is only one pair with level 2 capacity that is perpendicular to the pair with level 1 capacity. The next two possible transition paths are from state 1 to states 5 and 6, respectively, via transition direction (0, 1) that leads to one more pair with capacity level 0 (failed pair) and one less pair with capacity level 2. Similarly, there are two possible ways to transition from state 1 to state 5 and only one possible way from state 1 to state 6. The fifth possible transition path is from state 1 to state 2 via transition direction (−1, 1) that leads to one

more pair with level 0 capacity and one less pair with level 1 capacity. The pair with level 1 capacity in state 1 simply degrades to a lower capacity level through this transition path.

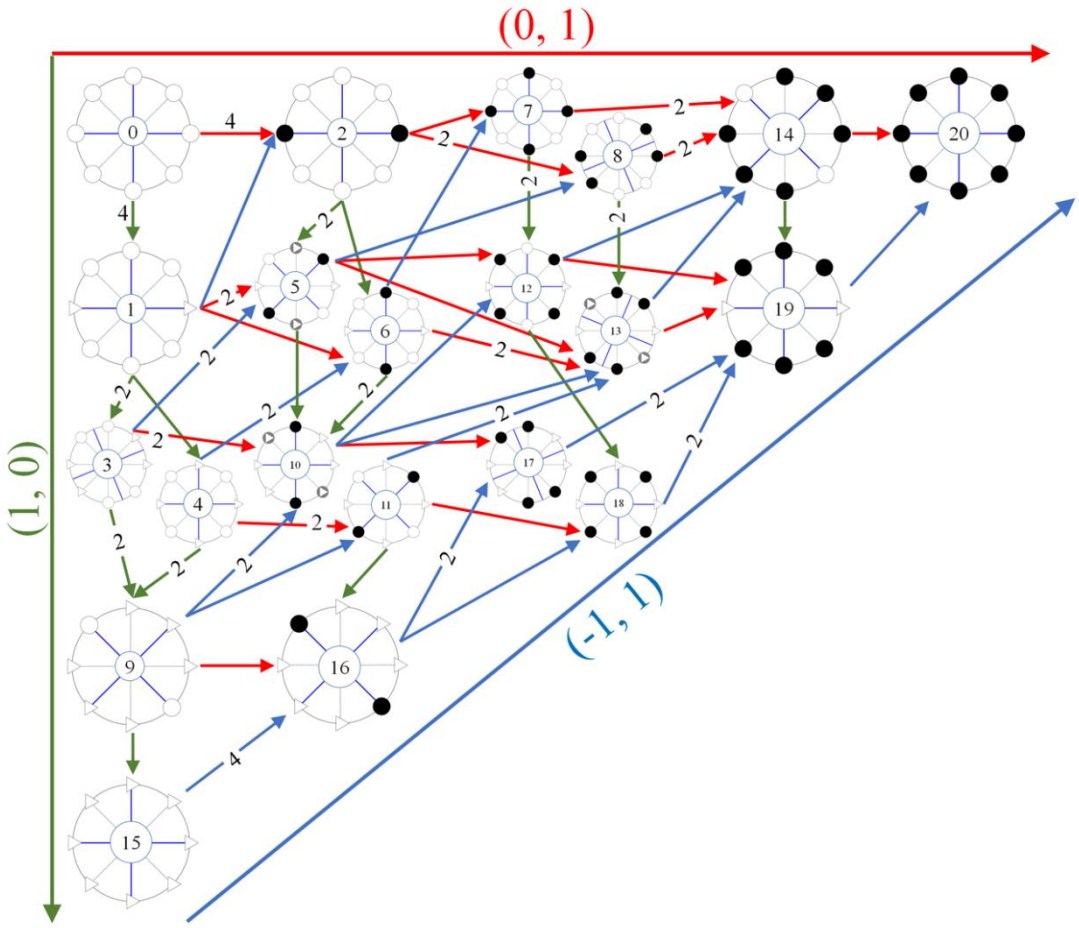


Figure 6.7 System state transition diagram

6.4.3.2 *State Transition Diagram Algorithm*

We build the state transition diagram via two levels of enumeration: the first level is to enumerate the coordinates; the second level is to enumerate the transitions from the system states represented by a coordinate to those represented by its follow-up coordinates. Figure

6.8 and Figure 6.9 show the flowcharts for the algorithm. In this section, we use the notations as follows:

D	the complete set of transition directions
P	the complete set of representative system states
V	the complete set of coordinates: each element in V is a m-dimensional vector and corresponds to at least one representative system states
U	the complete set of indices of representative system states in P that correspond to the coordinates in V: each element in U is a set of indices of representative system states in P corresponding to one coordinate in V
In	the index of the element equal to 1 in a transition direction
De	the index of the element equal to –1 in a transition direction
S	the current system state
S_f	the follow-up system state
IC	the set of indices of pairs in S to change state to get the follow-up system states
$\lambda(S, S_f)$	the transition rate from the current state S to the follow-up state S_f
$\mathbf{\Lambda}$	the transition matrix between system states
d	the index for transition direction
q	the index of the most recently found representative system states in P
v	the index of the current coordinate in V
v_f	the index of the follow-up coordinate in V
l	the index of the most recently added coordinate in V

As mentioned previously, multiple system states corresponding to the same coordinate are categorized into one group, we chose one of them as the representative system state. We use P to record all the representative system states. We use $U[v]$ to record the indices of the representative system states in P corresponding to $V[v]$. Therefore, any representative system state corresponding to $V[v]$ is retrieved from P by $P[U[v][i]]$ where $i = 1$ to the length of $U[v]$.

The first level of enumeration is coordinate enumeration. We enumerate the coordinates by the following procedure:

- We start the enumeration from the origin coordinate and index it as 1. We enumerate the follow-up coordinate of the origin. We assign an index to any coordinate according to the order the first time it is enumerated as a follow-up coordinate. Note that a coordinate can be the follow-up coordinate of more than one coordinate. Therefore, we assign the index when it is enumerated the first time.
- We then enumerate the coordinates in the order of their indices. For each coordinate we enumerate its follow-up coordinates via all the valid transition directions for the coordinate in the order specified as follows: The valid transition directions are ordered according to the location of the "1" element and "–1" element in the direction vector:
    - When a direction has no "–1", we order it before any other direction.
    - A direction with the i^{th} element being "–1" should be ordered before the directions with the j^{th} element being "–1" when $j > i$.

- For two directions both with the i^{th} element being "–1" or both with no "–1", we order them in the same way according to the location of "1".

- The coordinate enumeration stops when we finish enumerating the follow-up coordinates of the most recently added coordinate but no new coordinate is generated.

Consider the coordinate system in Figure 6.6. There are six possible transition directions as mentioned previously: (1, 0, 0), (0, 1, 0), (0, 0, 1), (–1, 1, 0), (–1, 0, 1) and (0, –1, 1) in order. None of the (1, 0, 0), (0, 1, 0) and (0, 0, 1) has "–1", we order them according to the location of "1". (–1, 1, 0) and (–1, 0, 1) are ordered before (0, –1, 1) since the former two directions have "–1" at the first dimension and the latter has "–1" at the second dimension. (–1, 1, 0) is ordered before (–1, 0, 1) since the former direction has "1" in the second dimension and the latter has "1" in the third dimension.

We start the enumeration from the origin (0, 0, 0) with node index 1. It has three valid transition directions: (1, 0, 0), (0, 1, 0) and (0, 0, 1) because [(0, 0, 0) + (1, 0, 0)], [(0, 1, 0) + (1, 0, 0)] and [(0, 0, 1) + (1, 0, 0)] all result in non-negative coordinates (1, 0, 0), (0, 1, 0) and (0, 0, 1). Then we enumerte coordinates in the order of (1, 0, 0), (0, 1, 0) and (0, 0, 1). We aslo assign indices 2, 3 and 4 to the three coordinates. We then move to coordinate (1, 0, 0) since its corresponding node has index 2. This coordinate has four valid directions (1, 0, 0), (0, 1, 0), (0, 0, 1) and (–1, 1, 0). Direction (–1, 1, 0) is valid because [(1, 0, 0) + (–1, 1, 0)] = (0, 1, 0) is all non-negative coordinate. Thus it results in coordinate (2, 0, 0), (1, 1, 0), (1, 0, 1), and (0, 1, 0). For the coordinates that are enumerated as follow-up

coordinates the first time, i.e. (2, 0, 0), (1, 1, 0) and (1, 0, 1), we assign indices 5, 6 and 7, respectively. We then move to coordinate (0, 1, 0) which cooreponds to node 3 and the procedure continues in the fashion.

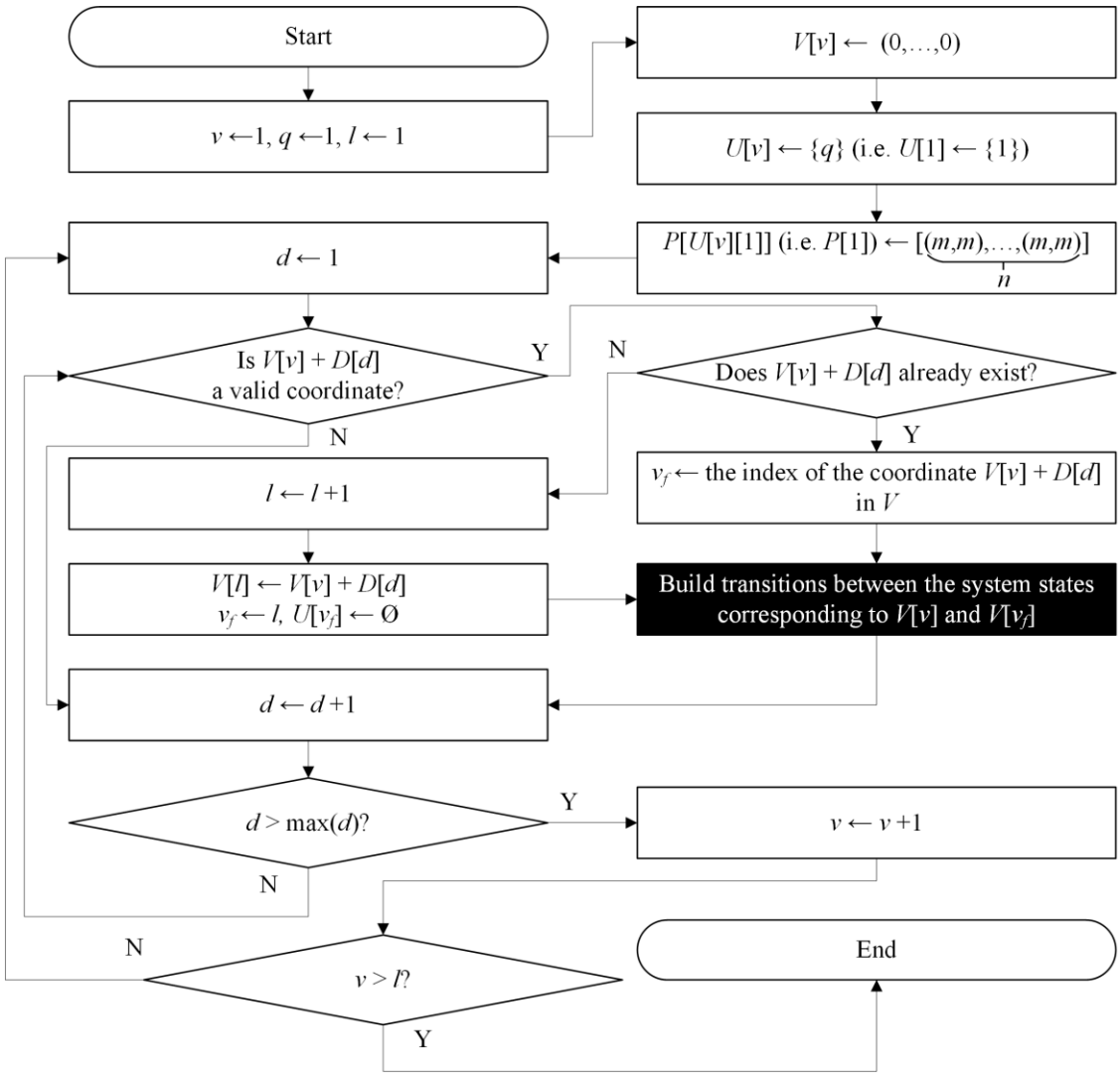


Figure 6.8 Flowchart of the algorithm of enumerating coordinates

The second level of enumeration is to find transitions between the system states corresponding to the current coordinate $V[v]$ and the system states corresponding to any

follow-up coordinate $V\left[v_f\right]$ of $V[v]$ as shown in the black box in Figure 6.8. The flowchart of this procedure is shown in Figure 6.9. In general, the procedure can be summarized as follows:

- For each system states S corresponding to coordinate $V[v]$, we find its follow-up states S_f corresponding to coordinate $V\left[v_f\right]$. To do this, we find the indices of change by examining the direction $D[d]$ via which $V\left[v_f\right]$ is derived from $V[v]$, as shown in Figure 6.9.
- For each follow-up state corresponding to $V\left[v_f\right]$, we balance the system and determine if it is already in the set of the representative system states, P. If not, we add it to the end of P and assign an index q to the state. We also add the index to $U\left[v_f\right]$, the set of indices of the representative system states corresponding the coordinate $V\left[v_f\right]$.
- When a transition from current state S to its follow-up state S_f is found, we add the transition rate $\lambda\left(S, S_f\right)$ between S and S_f to the matrix of transition rate, $\mathbf{\Lambda}$: $\mathbf{\Lambda}[u, w] \leftarrow \mathbf{\Lambda}[u, w] + \lambda\left(S, S_f\right)$ where u and w are the indices of S and S_f in P, respectively. Note that $\mathbf{\Lambda}[u, w]$ has zero value when initialized at the beginning of the algorithm.
- This enumeration procedure is finished when all the follow-up states (corresponding to coordinate $V\left[v_f\right]$) of the system states (corresponding to coordinate $V[v]$) are exhausted.

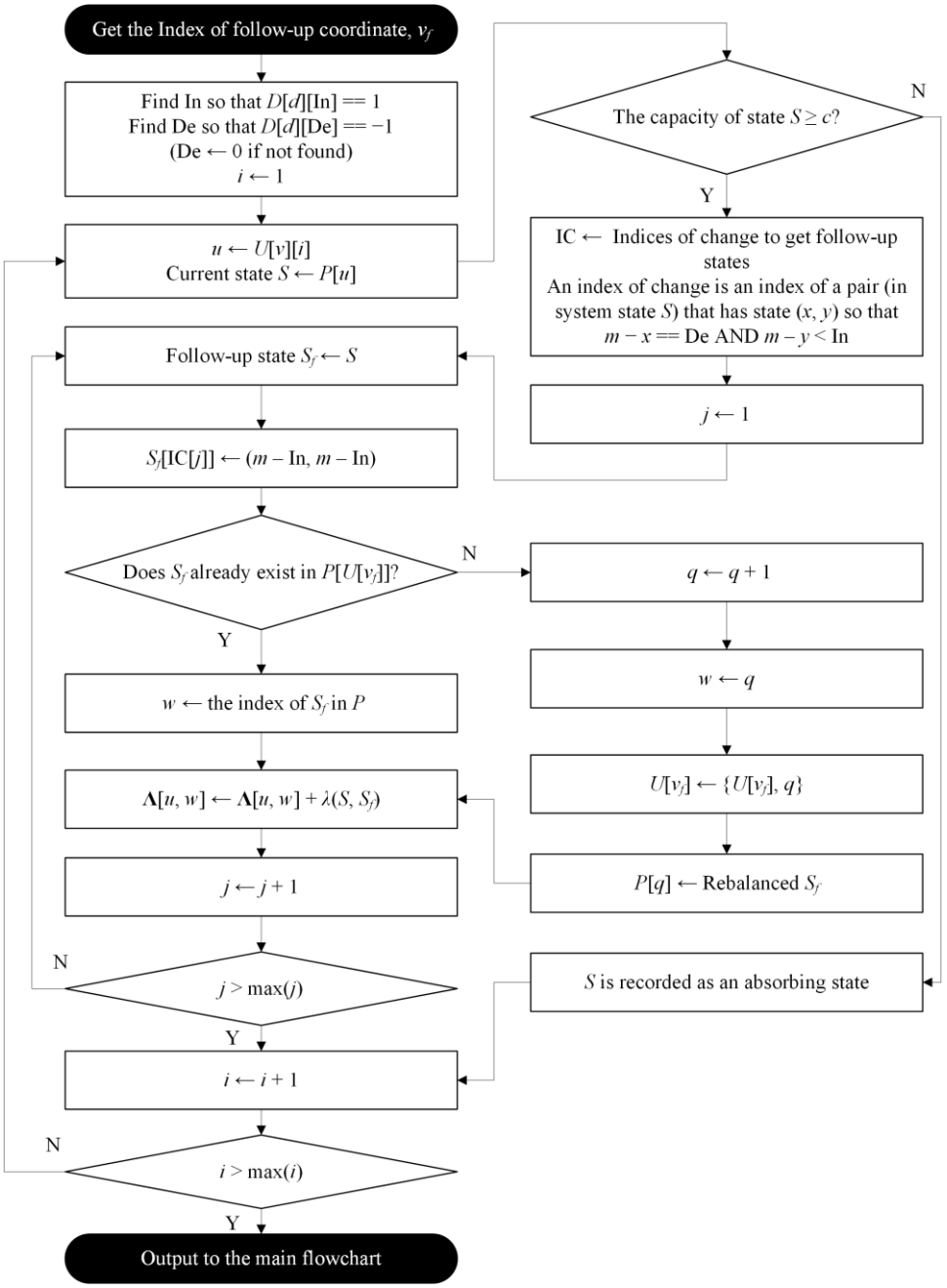


Figure 6.9 Flowchart of enumerating transitions between system states corresponding to two coordinates

6.5 Reliability Estimation

System reliability can be obtained by estimating the probability that the system is at an operating system state at time t. Since we assume that transition rates between system states are constant, then the state transition can be modeled with a continuous-time Markov chain (CTMC). System reliability is the probability that the system is in a transient state when any operating (failed) system state is considered as a transient (an absorbing) state.

6.5.1 *Fundamental Theory on Lumpable CTMC*

As mentioned previously, any system state in a state transition diagram is a representative of multiple similar system states: Any two system states are grouped in the same category when a system state is a repetition of the other state or its left-right flip. To justify that the transitions between two categories of system states can also be aggregated, we need to justify the lumpability of the CTMC. We now introduce the definition of a lumpable CTMC and the necessary and sufficient condition for an CTMC to be lumpable.

Let $\{X_n\}$ be a Markov chain with state space $\boldsymbol{\Theta} = \{e_1, e_2, \ldots, e_r\}$ and initial vector π. Given a partition $\bar{\boldsymbol{\Theta}} = \{E_1, E_2, \ldots, E_v\}$ of the state space $\boldsymbol{\Theta}$, a new chain $\{\bar{X}_n\}$ can be defined as follows: At the j^{th} step, the state of the new chain is the set E_i when E_i contains the state of the j^{th} step of the original chain.

Definition. Consider an CTMC $X(t)$ on a finite state space $\boldsymbol{\Theta}$ with transition probability matrix $P(t)=\left(p_{ij}(t)\right)$. The CTMC $X(t)$ is said to be lumpable with respect to the partition $\bar{\boldsymbol{\Theta}}$ if, for e_i, $e_j \in E_\xi$,

$$\sum_{e_k \in E_\eta} p_{ik}(t) = \sum_{e_k \in E_\eta} p_{jk}(t) \tag{6.1}$$

for all $t \geq 0$.

Definition. The generator Q of an CTMC is lumpable if

$$\sum_{e_l \in E_\eta} q_{il} = \sum_{e_l \in E_\eta} q_{jl} \tag{6.2}$$

for e_i, $e_j \in E_\xi$.

Theorem. A necessary and sufficient condition for an CTMC $X(t)$ to be lumpable is that its generator Q is lumpable. When Q is lumpable, we have $\bar{P}(t)=e^{t\bar{Q}}$.

6.5.2 *Lumpability of the System State Transition Process*

When we model the state transition process without aggregating different system states, the generator matrix Q is quite sparse. Therefore, as introduced previously, we aggregate any two system states into one category when one state is the repetition of the other state or its left-right flip. To obtain the left-right flip of a state we reverse the order of the elements in the vector of the state. The aggregation has the following two properties.

First, if a system state, S_1, is the repetition of another state, S_2, then S_1 has the same number of follow-up states as S_2, and each follow-up state of S_1 corresponds to a follow-up state of S_2. The two follow-up states are also repetition of each other. For example, let $S_1 = [(2,2), (0,0), (2,2), (2,1), (1,1), (1,1)]$, as shown in Figure 6.4(a), and $S_2 = [(1,1), (2,2), (0,0), (2,2), (2,1), (1,1)]$. S_1 is a repetition of S_2. Also each follow-up state of S_1 corresponds to one of S_2. For example, a follow-up state of S_1 is [(2,2), (0,0), (0,0), (2,2), (1,1), (1,1)]. It corresponds to the follow-up state of S_2, [(1,1), (2,2), (0,0), (0,0), (2,2), (1,1)].

Second, if a system state, S_1, is the repetition of the left-right flip of another state S_2, S_1 has the same number of follow-up states as S_2 does, and each follow-up state of S_1 corresponds to a follow-up state of S_2. Any follow-up state of S_1 is a repetition of a follow-up state of S_2 if we flip one of them in left-right direction. For example, we consider the $S_1 = [(2,2), (0,0), (2,2), (2,1), (1,1), (1,1)]$ mentioned above. We let S_2 be [(1,1), (1,1), (2,1), (2,2), (0,0), (2,2)]. The left-right flip of S_2 is exactly S_1. As mentioned earlier, a follow-up state of S_1 is $S_1' = [(2,2), (0,0), (0,0), (2,2), (1,1), (1,1)]$, it corresponds to a follow-up state of S_2, i.e. $S_2' = [(1,1), (1,1), (2,2), (0,0), (0,0), (2,2)]$. Apparently the left-right flip of S_2' is exactly S_1' and hence a repetition of S_1'.

Since the aggregated system states have the two properties mentioned above, the CTMC, $X(t)$ composed of all the system states can be lumped into an CTMC, $\bar{X}(t)$, with each

state representing a category of similar system states. The corresponding generator matrix is $\bar{Q}$. Based on the two properties, for the system states in a category (partition), denoted as E_ξ, their follow-up states should also be in one category, E_η. Moreover, due to the correspondence between the states in the two categories, any state e_i in E_ξ has the same number of follow-up states in E_η. The transition rates from a state $e_i \in E_\xi$ to its follow-up states in E_η and the transition rates from $e_j \in E_\xi$ to its follow-up states in E_η are the same.

The aggregated generator matrix $\bar{Q}$ has elements

$$\bar{q}_{\xi\eta} = \sum_{e_l \in E_\eta} q_{il} \tag{6.3}$$

for any $e_i \in E_\xi$.

6.5.3 *Probabilities of Successful Events*

Here a successful event is that the system is in a transient state at time t. To obtain the probability that the system is in a certain state at time t, we need to obtain the transient probabilities of the CTMC by solving a linear system of differential equations, known as Chapman-Kolmogorov differential equations as described next. Here the aggregated generator matrix $\bar{Q}$ is obtained previously as $\mathbf{\Lambda}$.

Once we obtain the transient probability of any state, we can obtain system reliability by summing the transient probabilities of all the transient states (operating system states).

Let $\bar{\pi}(t)=\left(\bar{\pi}_1(t),\bar{\pi}_2(t),\ldots,\bar{\pi}_v(t)\right)$ be the transient probabilities, e.g. $\bar{\pi}_\xi(t)=\Pr\left\{\bar{X}(t)=E_\xi\right\}$. Moreover, let $\bar{Q}=\left(\bar{q}_{\xi\eta}\right)$ where $\bar{q}_{\xi\eta}=\sum_{e_l\in E_\eta} q_{il}$ for any $e_i\in E_\xi$ when $\xi\neq\eta$, and $\bar{q}_{\xi\xi}=-\sum_{\xi\neq\eta}\bar{q}_{\xi\eta}$. The linear system of differential equations is

$$\frac{d}{dt}\bar{\pi}_\xi(t)=\sum_{E_\eta\in\bar{\boldsymbol{\Theta}}}\bar{Q}(\xi,\eta)\bar{\pi}_\eta(t) \tag{6.4}$$

for any $E_\xi\in\bar{\boldsymbol{\Theta}}$. The solution of the differential equation in vector form is

$$\bar{\boldsymbol{\pi}}(t)=\bar{\boldsymbol{\pi}}(0)e^{\bar{Q}t} \tag{6.5}$$

given the initial vector $\bar{\boldsymbol{\pi}}(0)$. However, this solution is difficult to obtain since $\bar{Q}$ is a large matrix. Therefore, we approximate the probabilities as follows.

We first obtain the Taylor series of $\bar{\boldsymbol{\pi}}(t)$ based on Eq. (6.6).

$$\bar{\boldsymbol{\pi}}(t)=\bar{\boldsymbol{\pi}}(0)\sum_{i=0}^{\infty}\frac{\left(\bar{Q}t\right)^i}{i!} \tag{6.6}$$

Let $U=I+\frac{\bar{Q}}{q}$ where I is an identity matrix of the same dimension as $\bar{Q}$ and $q\geq\max_\xi\left\{\left|\bar{q}_{\xi\xi}\right|\right\}$. It follows that $\bar{Q}=q(U-I)$. Therefore,

$$\bar{\boldsymbol{\pi}}(t)=\bar{\boldsymbol{\pi}}(0)e^{q(U-I)t}=\bar{\boldsymbol{\pi}}(0)e^{-qt}e^{qtU}=\bar{\boldsymbol{\pi}}(0)e^{-qt}\sum_{i=0}^{\infty}\frac{\left(qtU\right)^i}{i!} \tag{6.7}$$

Then $\bar{\boldsymbol{\pi}}(t)$ is approximated by truncating the summation of $\frac{(qtU)^i}{i!}$ to κ. To find the κ so that the error is no more than ε, we let

$$\sum_{i=0}^{\infty} \frac{e^{-qt}(qt)^i}{i!} \hat{\boldsymbol{\pi}}(i) - \sum_{i=0}^{\kappa_\varepsilon} \frac{e^{-qt}(qt)^i}{i!} \hat{\boldsymbol{\pi}}(i) = \sum_{i=\kappa_\varepsilon+1}^{\infty} \frac{e^{-qt}(qt)^i}{i!} \hat{\boldsymbol{\pi}}(i) \le \varepsilon \tag{6.8}$$

where $\hat{\boldsymbol{\pi}}(i) = \bar{\pi}(t)U^i$. Since $\hat{\boldsymbol{\pi}}(i) \le 1$, it follows that we should choose κ_ε such that

$$\sum_{i=\kappa_\varepsilon+1}^{\infty} \frac{e^{-qt}(qt)^i}{i!} = 1 - \sum_{i=0}^{\kappa_\varepsilon} \frac{e^{-qt}(qt)^i}{i!} \le \varepsilon \tag{6.9}$$

Since $\left(1 - \sum_{i=0}^{\kappa_\varepsilon} \frac{e^{-qt}(qt)^i}{i!}\right)$ increases when t increases, we should increase κ_ε as t increases. Or we set κ_ε according to the largest value of t of concern.

6.6 Numerical Examples

6.6.1 *Numerical Example* 1

We first consider a system with only two capacity levels. The system is reduced to the system studied in Chapter 3. As in the second numerical example in Chapter 3, we let the individual units have exponentially *i.i.d.* lifetimes with a failure rate 0.025. That is: we let the transition rate of each pair from capacity level 1 to capacity level 0 be 0.05.

We estimate the reliability of the k-out-of-6 pairs:G Balanced systems where $k = 1$ to 6. We found that the results are very close to those obtained in Chapter 3. The reliability

functions are shown in Figure 6.10 where the lines marked with circles are reliability functions obtained by the method in Chapter 3 and the solid red lines are reliability functions obtained by the method proposed in this chapter. Also, system reliability decreases when k increases.

This numerical example validates that the method is theoretically correct.

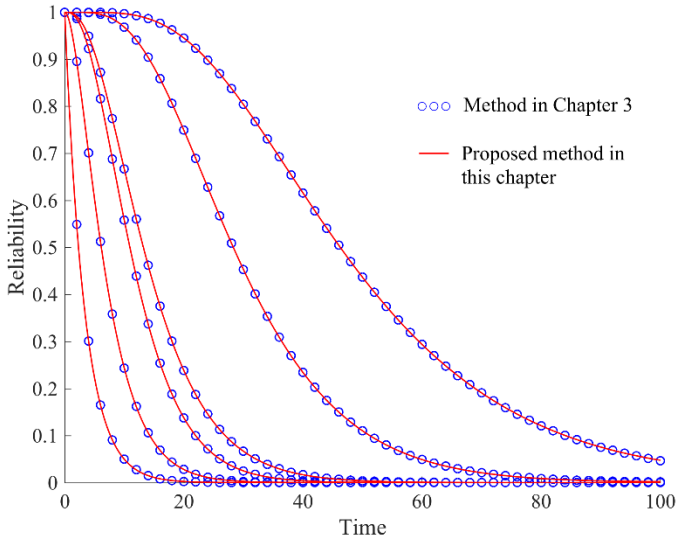


Figure 6.10 Comparison between the method in Chapter 3 and the method proposed in this chapter

6.6.2 *Numerical Example* 2

In this example, we let $m=2$ which means any pair has three capacity levels. Also, we let the capacity of each pair at different levels be $c_2=4$, $c_1=2$, and $c_0=0$. We use λ_{ij} to denote the rate of transition that the actual capacity of any pair decreases from level i to

level j given its serving capacity is equal to its actual capacity. We use $\lambda_{ij}^{'}$ to denote the transition rate given the serving capacity is less than the actual capacity. We let $\lambda_{21} = 0.04$, $\lambda_{20} = 0.02$, $\lambda_{10} = 0.03$, and $\lambda_{20}^{'} = 0.025$.

We consider a system with 6 pairs. Therefore, the maximum system capacity is 24. We estimate the reliability functions when the minimum capacity requirement $c = \{2, 4, 6 \ldots, 24\}$.

Figure 6.11 shows the 12 reliability functions where a lower reliability function corresponds to a greater value of c.

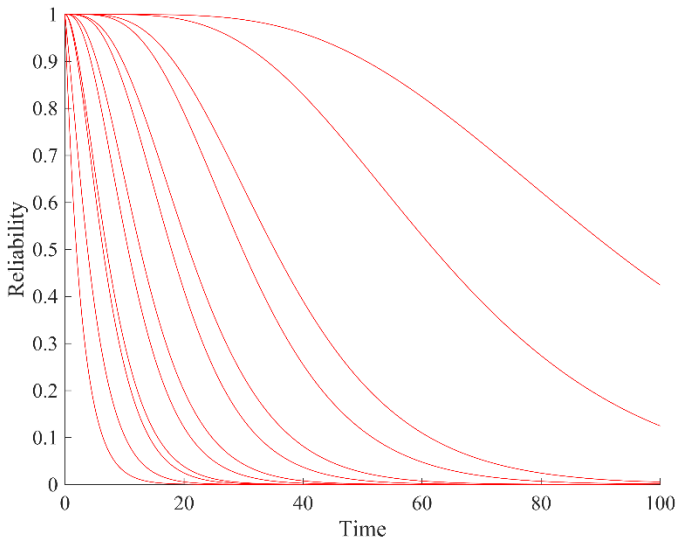


Figure 6.11 Reliability functions of weighted-*c*-out-of-6 pairs:G Balanced systems with different *c*

6.7 Conclusions

In this chapter, we investigate the reliability estimation for weighted-c-out-of-n pairs:G Balanced system with pairs of multiple capacity levels. The capacity of any pair decreases from a level to a lower level. A pair fails when its capacity reaches capacity level 0. The system fails when its total available capacity is lower than a minimum requirement. We use lumpable CTMC to model and estimate the system reliability. A numerical example shows the validity of the method.

CHAPTER 7

LOAD-SHARING EFFECT ON SYSTEM RELIABILITY

7.1 Problem Definition and Assumptions

In this chapter, we investigate reliability estimation of weighted-*c*-out-of-*n* pairs:G Balanced systems by considering load-sharing effect: When one of the units fails its load is shared among the remaining units. The load shared by each operating unit affects its reliability and hence the system reliability.

In this chapter, we only consider weighted-c-out-of-n pairs:G Balanced systems with units of two different capacities. The units have either full capacity when it is operating or zero capacity when it fails. Figure 7.1 shows a weighted-c-out-of-6 pairs:G Balanced system with units of two different capacities. In Figure 7.1, the capacities of units are represented by different shapes, i.e. circle and triangle. The circle shaped units have capacity c_1 and triangle shaped units have capacity c_2 where $c_1 > c_2$.

We assume that when a unit fails the load corresponding to its pair is shared by the remaining operating pairs, which increases their hazard rates. We also assume that the lifetimes of individual units are independent random variables. These two assumptions imply that the lifetimes of individual pairs are independent except that the lifetimes are affected by the load shared by the operating pairs.

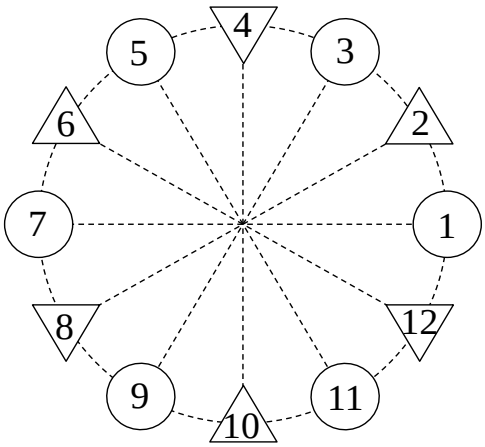


Figure 7.1 Example of weighted-*c*-out-of-*n* pairs:G Balanced system with 6 pairs

In addition, we assume that the individual units of the same type have identical lifetime distributions. Therefore, pairs composed of the same type of units have identical lifetime distributions as well.

Furthermore, we assume that when a pair p^* experiences an event such as failure, being forced down, or resumption of operation at time τ, the state of pair p^* at τ is the same with its state immediately before the event. We use a step function, $\rho_{p^*}(\tau)$, to describe the up (operating) and down (failed or forced-down) of pair p^*. The function $\rho_{p^*}(\tau)$ has a value of 1 if the pair p^* is operating at τ and 0 otherwise. Note that if the pair p^* fails or is forced down at time τ, then $\rho_{p^*}(\tau)=1$ and $\rho_{p^*}(\tau^+)=0$ where τ^+ is the immediate time instant after time τ.

7.2 System Description

In this section, we introduce the weighted-c-out-of-n pairs:G Balanced system considered in this chapter. The system consists of an even number of pairs of units that have either of two capacities c_1 and c_2. Units are arranged in such a way that any two adjacent units have different capacities as shown in Figure 7.1. The system fails when either of the two scenarios occurs: (i) The total capacity of the system is lower than c. (ii) The load shared by any pair is greater than the capacity of the pair.

The system is required to maintain balance at all times, i.e. the operating units are symmetric w.r.t. at least a pair of perpendicular axes, and that the operating units with the same capacity are symmetric w.r.t. the same pair of axes as well. Consider the system in Figure 7.1 We present some states of the system in Figure 7.2 where pairs in white are operating, pairs in black are failed, and pairs in gray are in standby.

Again, the circle shaped units have capacity c_1 and triangle shaped units have capacity c_2 where $c_1 > c_2$. The system state in Figure 7.2(a) is balanced because operating units with both capacities c_1 and c_2 are symmetric w.r.t. the perpendicular axes which are shown as two blue lines. The system state shown in Figure 7.2(b) is unbalanced even though the operating units with capacity c_1 (c_2) are symmetric w.r.t. the perpendicular axes in red (green) dashed lines, the two sets of axes of symmetry do not overlap.

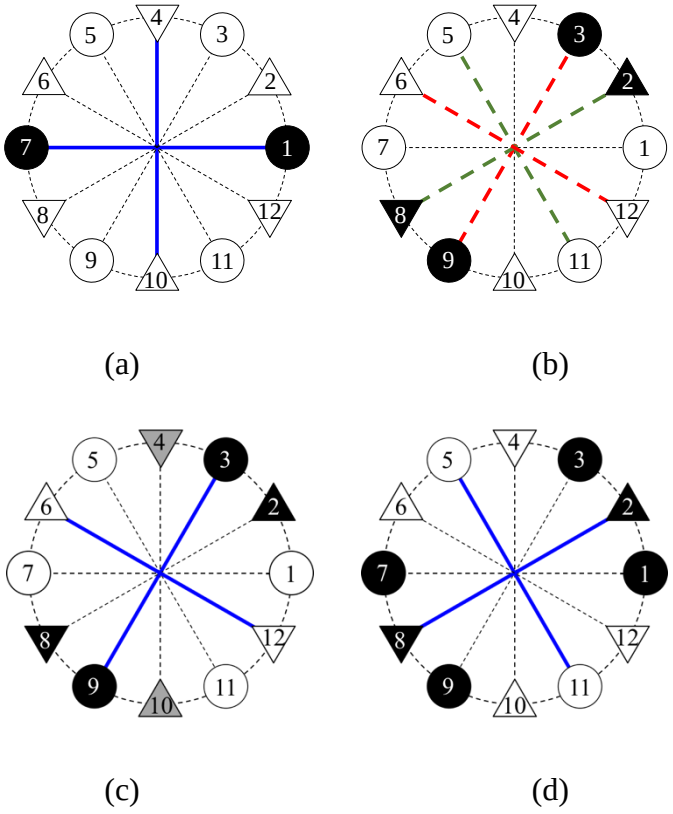


Figure 7.2 Illustration for balanced system state, unbalanced system state, and standby pair

In addition, the unbalanced system is rebalanced by forcing down operating pairs into standby and/or resuming standby pairs into operation when feasible. For example, to regain the balance of the system state in Figure 7.2(b), we force down an operating pair, pair 4^*, into standby as shown in Figure 7.2(c). The corresponding axes of symmetry are shown in blue lines. Note that we can force down pair 1^* instead to regain the balance of the system state in Figure 7.2(b). However, pair 4^* has lower capacity; we lose less capacity if pair 4^* is forced down.

When a standby pair is no longer necessary to keep a system in balance, we resume the standby pair into operation. Suppose a system is in the state shown in Figure 7.2(c), and pair 1^* fails afterwards. Now it is no longer necessary for pair 4^* to be in standby to keep the system in balance. Pair 4^* is then resumed into operation as shown in Figure 7.2(d).

7.3 System Balance Determination

We now introduce the method for determining the balance of a weighted-c-out-of-n pairs:G Balanced system with units of two capacities, and finding the axes of symmetry and operating pairs to be forced down into standby and/or standby pairs to be resumed into operation when the system is unbalanced.

7.3.1 *Simple Rules*

Some rules can be applied if system state falls into some special cases.

Rule 1. When there is only one failed pair, the system is balanced and the axes of symmetry are along and perpendicular to this pair.

Rule 2. When there are two failed pairs:

(i) When the two failed pairs have the same capacity, the system is balanced, and an axis of symmetry is in the middle of the two pairs.

(ii) When the two failed pairs have different capacities: (a) If the two pairs are along two perpendicular axes, then the system is balanced, and the axes of symmetry are

along the two failed pairs. (b) Otherwise, the system is not balanced. We choose the pair with the larger capacity as an axis of symmetry, and force down an operating pair that is symmetric with the other failed pair w.r.t. the axis of symmetry. This results in a balanced system with the maximum available system capacity.

Rule 3. When the total number of operating and standby pairs is two:

(i) When the two pairs have the same capacity, resume standby pair if any. This results in a balanced system and the axis of symmetry is in the middle of the two pairs.

(ii) When the two pairs have different capacities: (a) If the two pairs are perpendicular to each other, resume standby pair if any. This results in a balanced system and the axes of the symmetry are along the two pairs. (b) Otherwise, we choose the pair with the larger capacity as an axis of symmetry and resume its operation if it is in standby. The other pair should stay in standby if it is a standby pair, or be forced down into standby if it is operating.

Rule 4. When the failed pairs are consecutively arranged, resume all the standby pairs if any:

(i) When the number of failed pairs is odd, the axis of symmetry is in the middle of the failed pairs.

(ii) Otherwise, force down the operating pair that has the lower capacity and is next to the consecutive failed pairs. The axis of symmetry is in the middle of the failed pairs and the standby pair which is just forced down.

7.3.2 *Axis of Symmetry Algorithm*

When the system state does not fall into any of the above mentioned special cases, we use the method introduced in Chapter 4 to determine the symmetry (balance) of the system. We need the weights of individual units to obtain the Moment Difference (MD). Since individual units have different capacities, the weights for the units here are the absolute values of their states times their capacities. A unit has state 1 if it is operating, 0 if it is failed, and –1 if it is forced down into standby.

If the system is not symmetric we use MD to determine an initial candidate axis, from which we begin to enumerate a series of candidate axes. A candidate axis of symmetry is along a pair of units. Due to the orthogonal relationship of the candidate axes, there are $n/2$ unique candidate axes of symmetry in total. The other $n/2$ candidate axes are perpendicular to them.

The procedure of enumerating candidate axes is the same as the one introduced in Chapter 3. Among all the candidate axes of symmetry, we choose the one that results in the maximum available system capacity. When available system capacity alone cannot differentiate the candidate axes, we use other criteria based on system requirements.

7.4 Reliability Estimation Considering Load-Sharing Effect

In this section, we discuss the reliability estimation of weighted-c-out-of-n pairs:G Balanced system considering load-sharing effect. The basic procedure is as follows. First, we enumerate all the possible successful events E_h^w where h pairs fail resulting in a

balanced or rebalanced system (the unbalanced system is rebalanced by considering standby) with at least c available capacity. The superscript w is to distinguish different events with the same h. Second, we obtain the probability of any successful event $P_h^w(t)$. Then, system reliability is obtained by summing the probabilities of all the successful events:

$$R_{sys}(t) = \sum_{h=0}^{H} \sum_{w} P_h^w(t) \tag{7.1}$$

where H is the maximum value of h that results in a successful system, i.e. a balanced or rebalanced system with at least c available capacity.

7.4.1 *Successful Event Enumeration*

We follow the fundamental method of event enumeration introduced in Chapter 3: starting from a system state with no failures, we enumerate its follow-up states until we exhaust all the possible successful system states with at least c available capacity and all the possible transitions between these states. A follow-up state of a state can be obtained by turning one of the operating pairs in the current system state into failure and rebalancing it if the additional failure results in an unbalanced system state. A successful event is a transition path that leads to a balanced or rebalanced system state with at least c available system capacity. For instance, as shown in Figure 7.3, the successful events of a 4-out-of-12 capacity:G Balanced system, which consists of units with two types of capacities $c_1 = 2$ (represented by a circle) and $c_2 = 1$ (represented by a triangle), are all the transition paths that lead to system states 1 to 8 and states 14 and 15. For each successful event, we record necessary information to calculate its probability as indicated below.

Note that the units have two different capacities. Hence there are two possible branches of state transition paths which start with the failure of units with two different capacities respectively. As shown in Figure 7.3, the two branches are on the left and right sides of state 0, respectively.

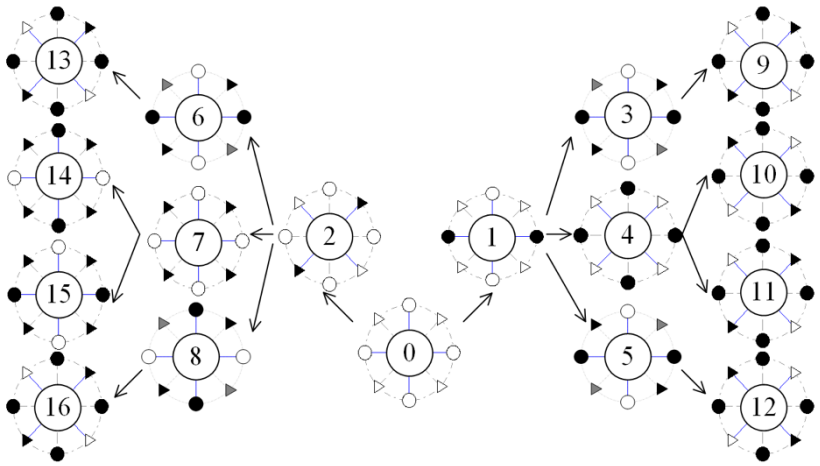


Figure 7.3 Diagram of state transition paths for system with four pairs of units with two different capacities

7.4.2 *Probability Function of Successful Event*

7.4.2.1 *Probability Model*

The probability of a successful event when h pairs fail by time t can be expressed as

$$P_h^w(t) = \int_{\tau_h=0}^{t}\cdots\int_{\tau_2=0}^{\tau_3}\int_{\tau_1=0}^{\tau_2}\left\{\begin{array}{l}\prod_{i=1}^{h}\left[\lambda_{h,x_i^*}^w(\tau_i)\exp\left(-\Lambda_{h,x_i^*}^w(\tau_i)\right)\right]\\ \times\prod_{y^*\in Y_h^w}\exp\left(-\Lambda_{h,y^*}^w\left(T_{h,y^*}^w\right)\right)\\ \times\prod_{z^*\in G_h^w}\exp\left(-\Lambda_{h,z^*}^w(t)\right)\end{array}\right\} d\tau_1 d\tau_2\cdots d\tau_h \quad (7.2)$$

where the superscript w is used to distinguish different all events that have h failed pairs by time t; i is the order of failures; τ_i is the time when the i^{th} failure occurs which is a random variable; x_i^* is the identity number of the i^{th} failed pair; Y_h^w is the set of identity numbers of standby pairs at time t; y^* is the identity number of standby pair; T_{h,y^*}^w is the last time when pair y^* is forced down (and not resumed by time t) the value of which is one of the τ_i's; G_h^w is the set of identity numbers of operating pairs at time t; z^* is the identity number of operating pair; $\lambda_{h,p^*}^w(\cdot)$ is the hazard rate of pair p^*; $\Lambda_{h,p^*}^w(\cdot)$ is the cumulative hazard rate of pair p^*. Eq. (7.2) has three parts in the braces: the first part in the braces is the probability that h pairs fail at time τ_i's by time t; the second part is the probability that pairs in set Y_h^w are forced down into standby at time T_{h,y^*}^w and stay in standby by time t; the third part is the probability that operating pairs in set G_h^w survive time t.

7.4.2.2 *Hazard Rate Model*

In this section, we present the hazard rate model for this load-sharing system. Let overall system load at time τ be $D(\tau)$ and load shared by operating pair p^* at τ be $d_{p^*}(\tau)$. When the equal load-sharing rule is followed, $d_{p^*}(\tau) = D(\tau)/g(\tau)$ where $g(\tau)$ is the number of operating pairs at time τ.

Let the hazard rate for pair p^* be $\lambda\left(\tau, d_{p^*}, \rho_{p^*}, \delta_{p^*}, \beta_{p^*}\right)$. A typical example for λ is

$$\begin{aligned}&\lambda\left(\tau, d_{p^*}, \rho_{p^*}, \delta_{p^*}, \beta_{p^*}\right) = \\ &\rho_{p^*}(\tau)\,\beta_{p^*}\left(\tau - \delta_{p^*}(\tau)\right)\exp\left(a_0 + a_1 d_p^*(\tau)\right)\end{aligned} \tag{7.3}$$

where $\beta_{p^*}(\cdot)$ is the baseline hazard rate of pair p^*; $\delta_{p^*}(\tau)$ is the standby duration of pair p^* by time τ; the exponential function expresses the effect of load-sharing on hazard rate; a_0 and a_1 are constants.

Another example for λ is

$$\begin{aligned}&\lambda\left(\tau, d_{p^*}, \rho_{p^*}, \delta_{p^*}, \beta_{p^*}\right) = \\ &\rho_{p^*}(\tau)\,\beta_{p^*}\left(\tau - \delta_{p^*}(\tau)\right)\left[a_0 + a_1 d_{p^*}(\tau)\right]\end{aligned} \tag{7.4}$$

where the effect of load-sharing on hazard rate is modeled by the sum in the square brackets.

The cumulative hazard rate Λ_{p^*} can be obtained by

$$\begin{aligned}\Lambda_{p^*}(\tau) &= \int_{u=0}^{\tau} \lambda\left(u, d_{p^*}, \rho_{p^*}, \delta_{p^*}, \beta_{p^*}\right) du \\ &= \sum_{v=0}^{h(\tau)} \int_{u=\tau_v}^{\min(\tau_{v+1},\tau)} \lambda\left(u, d_{p^*}, \rho_{p^*}, \delta_{p^*}, \beta_{p^*}\right) du\end{aligned} \tag{7.5}$$

where $h(\tau)$ is the number of failures up to time τ.

7.4.2.3 *Simplified Probability Function of Successful Event*

The problem can be simplified by the following assumptions.

- $D(\tau)$ is constant, which results in a constant $d_{p^*}(\tau)$ value between any two consecutive failure times, $(\tau_{i-1}, \tau_i]$.
- The baseline hazard rate of any pair p^*, β_{p^*}, is constant.

The two assumptions imply that hazard rates of any pairs are constant during $(\tau_{i-1}, \tau_i]$. Under the two assumptions, $\Lambda_{p^*}(\tau)$ can be simplified into a discrete model

$$\begin{aligned}\Lambda_{p^*}(\tau) &= \sum_{v=0}^{h(\tau)} \lambda_{p^*,v} \cdot \left(\min(\tau_{v+1}, \tau) - \tau_v\right) \\ &= \sum_{v=0}^{h(\tau)-1} \lambda_{p^*,v} \cdot (\tau_{v+1} - \tau_v) + \lambda_{p^*,h(\tau)} \cdot \left(\tau - \tau_{h(\tau)}\right) \\ &= -\sum_{v=1}^{h(\tau)} \left(\lambda_{p^*,v} - \lambda_{p^*,v-1}\right)\tau_v + \lambda_{p^*,h(\tau)}\tau\end{aligned} \tag{7.6}$$

where τ_v is the time when the v^{th} failure occurs and $\tau_0 = 0$; $\lambda_{p^*,v}$ is the hazard rate of pair p^* immediately after the v^{th} failure which is constant until the $(v+1)^{\text{st}}$ failure. Examples for hazard rate are

$$\lambda_{p^*,v} = \rho_{p^*,v}\beta_{p^*}\exp\left(a_0 + a_1 d_{p^*,v}\right) \tag{7.7}$$

or

$$\lambda_{p^*,v} = \rho_{p^*,v}\beta_{p^*}\left(a_0 + a_1 d_{p^*,v}\right) \tag{7.8}$$

where $d_{p^*,v}$ is the load pair p^* shares after the v^{th} failure and before the $(v+1)^{\text{st}}$ failure; $\rho_{p^*,v} = 1$ if pair p^* is operating after the v^{th} failure and 0 otherwise.

The probability of successful events can then be simplified as

$$\begin{aligned} P_h^w(t) &= \left(\prod_{i=1}^{h} \lambda^w_{h,x_i^*,i-1}\right) \times \exp\left(-t \cdot \sum_{z^* \in G_h^w} \lambda^w_{h,z^*,h}\right) \\ &\times \int_{\tau_h=0}^{t} \cdots \int_{\tau_1=0}^{\tau_2} \exp\left(\sum_{i=1}^{h} \alpha^w_{h,i}\tau_i\right) d\tau_1 \cdots d\tau_h \end{aligned} \tag{7.9}$$

where the superscript and subscript combination, ^w_h, is used to distinguish the variables corresponding to a specific event and the other subscripts should be interpreted as mentioned above without considering ^w_h; $\alpha^w_{h,i} = \sum_{p^*=1}^{n}\left(\lambda^w_{h,p^*,i} - \lambda^w_{h,p^*,i-1}\right)$ is the coefficient of τ_i. We develop an iterative procedure in Chapter 3 to obtain the closed form expression and the numerical value for Eq. (7.9).

When enumerating successful events, we record $\rho^w_{h,p^*,i}$ and $d^w_{h,p^*,i}$ at each τ_i, from which we obtain a matrix of hazard rate $\mathbf{L}^w_h = \left[\lambda^w_{h,p^*,i}\right]_{n\times(h+1)}$ where $p^* = 1$ to n and $i = 0$ to h. The coefficients for τ_i can then be obtained as

$$\begin{aligned} \boldsymbol{\alpha}^w_h &= \left(\alpha^w_{h,1}, \alpha^w_{h,2}, \cdots, \alpha^w_{h,h}\right) \\ &= \sum_{column}\left[\mathbf{L}^w_h(\cdot, 2:h+1) - \mathbf{L}^w_h(\cdot, 1:h)\right] \end{aligned} \tag{7.10}$$

where $\sum_{column}\mathbf{M}$ is the row vector resulting from summing the columns of a matrix $\mathbf{M}$, $\mathbf{L}(\cdot, i:j)$ is the columns i to j of the matrix $\mathbf{L}$.

7.4.2.4 *Proportional Load-Sharing Rule*

The equal load-sharing rule mentioned in the Introduction section does not consider capacities of individual units. It is not applicable in the case where system load is distributed to units according to their capacities. We introduce the proportional load-sharing rule.

Under the proportional load-sharing rule, overall system load is distributed to units proportional to their capacities. Consider the system in Figure 7.1. Suppose units (identified by a circle) have capacity $c_1 = 2$ and units (identified by a triangle) have capacity $c_2 = 1$, the total system capacity is 18. When all pairs of units are operating, each pair composed of units in circle shares load $4/18D$ and each pair composed of units in triangle shares load $2/18D$. Suppose unit 2 fails, unit 8 is forced down and the system is balanced. Now total system available capacity is 16. Each pair composed of units in circle shares load $4/16D$ and each pair composed of units in triangle shares load $2/16D$.

7.5 Numerical Example

In this section, we present a numerical example for a weighted-c-out-of-n pairs:G Balanced system to show the effect of load-sharing on system reliability. In this system, we have $n = 6$ pairs of units with any two adjacent units having two different capacities:

$c_1 = 2$ and $c_2 = 1$. When the two units in pair p^* have capacity c_i ($i = 1$ or 2), baseline hazard rate of pair p^* is $\beta_{p^*} = 0.1c_i^{-\theta}$ and the capacity of pair p^* is $C_{p^*} = 2c_i$. In addition, we assume overall system load $D = c$.

We maintain the assumptions in the simplified model and specify the hazard rate function as

$$\lambda_{p^*,v} = \rho_{p^*,v}\beta_{p^*} \exp\left[a_0 \left(-1 + a_1 \frac{d_{p^*,v}}{C_{p^*}} \right) \right] \tag{7.11}$$

where a_0 and a_1 are two constants. From Eq. (7.11) we observe that when $d_{p^*,v} = C_{p^*}/a_1$ and $\rho_{p^*,v} = 1$, i.e. pair p^* is operating after the v^{th} failure, its hazard rate is equal to its baseline hazard rate. We name C_{p^*}/a_1 as the baseline load of pair p^*.

We estimate the reliability of the weighted-c-out-of-6 pairs:G Balanced system and let the required capacity c be 6, 9, and 12, respectively. Let $a_0 = 1$; $a_1 = 1$, 1.5, and 2; $\theta = 0.5$, 1, and 2. The reliability functions with different coefficient values are plotted in Figure 7.4 and Figure 7.5. In Figure 7.4 each row represents a value for c and each column represents a value for θ. In Figure 7.5 each row represents a value for c and each column represents a value for a_1. From Figure 7.4 and Figure 7.5, we have the following conclusions:

(a) System reliability decreases when c increases.

(b) System reliability decreases when a_1 increases. This is because the baseline loads of individual pairs decrease when a_1 increases. The load-sharing effect is more significant due to a larger a_1.

(c) System reliability increases when θ increases. This is because baseline hazard rates of individual units decrease when θ increases.

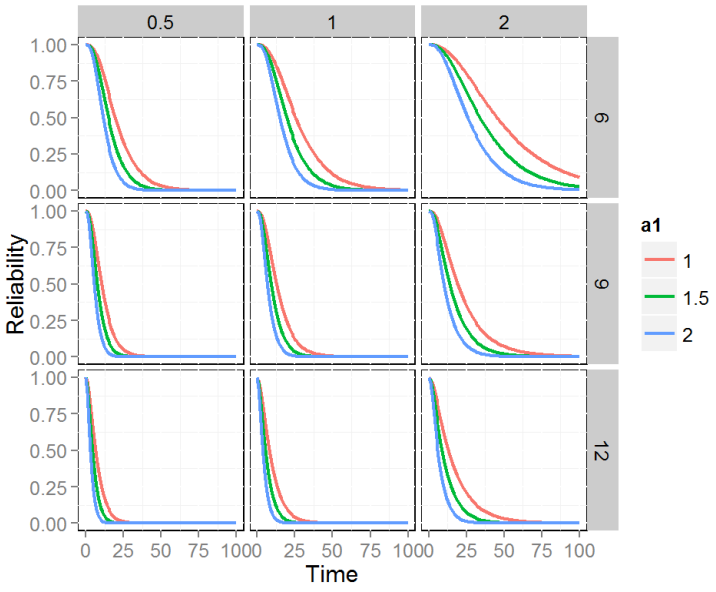


Figure 7.4 Reliability plots against a_1

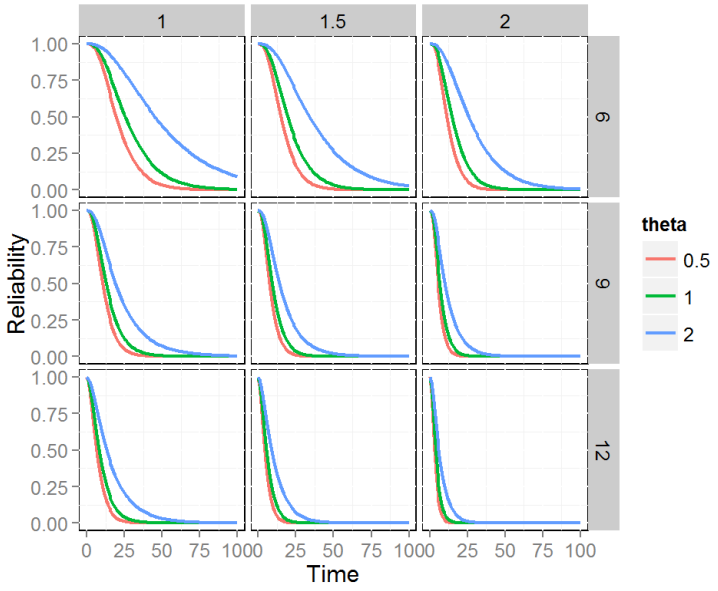


Figure 7.5 Reliability plots against θ

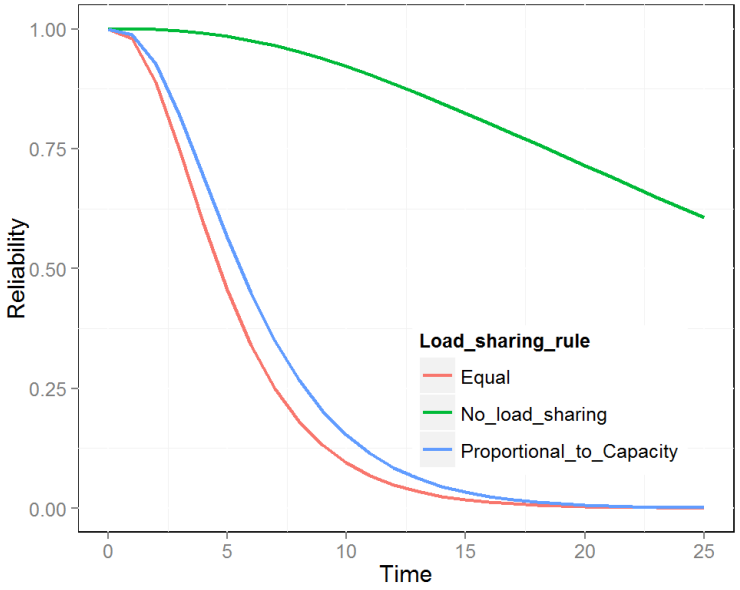


Figure 7.6 Reliability plots against load-sharing rules

Set the coefficient values as $c=6$, $a_1=5$, and $\theta=2$. We estimate system reliability in three scenarios: equal load-sharing rule, proportional load-sharing rule, and no load-

sharing effect. The results are in Figure 7.6. The system has higher reliability when there exists no load-sharing effect which implies that load-sharing effect has an adverse effect on the system reliability. We also observe that system has higher reliability when the load is distributed to units proportional to its capacity.

7.6 Conclusions

In this chapter, we investigate the effect of load-sharing on the reliability of weighted-c-out-of-n pairs:G Balanced system. We develop a heuristic to determine the axes of symmetry, a procedure for enumerating successful events, and the probability functions for successful events. We assume that higher load-sharing increases hazard rate and hence has an adverse effect on system reliability as demonstrated using the numerical example.

The reliability model proposed in this chapter provides more realistic reliability estimation for weighted-c-out-of-n pairs:G Balanced system with significant load-sharing effect. In addition, we also consider the capacities of units in load distribution model.

CHAPTER 8

OPTIMAL DESIGN FOR RELIABILITY

8.1 Problem Definition

The *k*-out-of-*n* pairs:G Balanced systems are emerging systems in many industries such as aerospace, military, and service. It is critical to optimize the system to achieve maximum reliability of given the constraints on the total number of pairs in the systems, the minimum required operating pairs, the spatial configurations of the units in the system, and the load-sharing effect among individual pairs. Additional constraints include the available number of pairs of each type when the system is composed of pairs with different lifetime distributions.

In this chapter, we first maximize the system reliability metrics by searching for the optimal total number of pairs and the optimal standby policy for *k*-out-of-*n* pairs:G Balanced systems. Then we maximize the reliability metrics by allocating the pairs of different lifetime distributions given the total number of pairs in the system. We study the first reliability optimization problem in Section 8.2. We perform the optimization in two sequential parts in Sections 8.2.1 and 8.2.2. Then we study the second optimization problem in Section 8.3 and provide conclusions in Section 8.4.

8.2 Optimal Total Number of Pairs and Optimal Standby Policy

The assumptions in the Introduction section still hold. Moreover, in this section, we assume the following:

- All units are identical and the lifetimes of individual units are *i.i.d.* exponential random variables.
- The hazard rates of the individual units are identical.

In general, the problem can be defined as follows:

Objective:	Maximize reliability metrics such as MTTF of the *k*-out-of-*n* pairs:G Balanced system.
Decision variables:	n, the total number of pairs in the system;
	K_h (h = 0 to ($n - k$)), the minimum number of standby when the system has h failed pairs.
Constraints:	Upper and lower bounds of n;
	Upper and lower bounds of K_h.

Note that *k*, the minimum number of operating pairs in the system is determined by the total load on the system and the capacity of each operating pair since the total capacity of operating pairs should meet the total load on the system. We assume that the load on the system, w_s, consists of two parts: the initial load, w_o, and the load caused by adding more pairs, w_p. Specifically, $w_s = w_o + n \times w_p$. When the capacity of each pair is c_p, the

minimum number of operating pairs needed is $k \geq w_s / c_p$. It is clear that k is determined by n given the values of w_o, w_p and c_p .

We decompose the problem into two sequential parts. First, we find the optimal value for n by fixing the minimum number of standby pairs, i.e. let $K_h \equiv 0$ for $h = 0$ to $(n-k)$. Second, we find the optimal values for K_h given the optimal value for n obtained in the first part. In the second part, we also search for values in the neighborhood of the optimal n to explore the possibility that the optimal n found in the first part may not be optimal in the second part.

8.2.1 *Optimal Total Number of Pairs*

8.2.1.1 *Problem Modeling*

The problem is modeled as follows:

Maximize Reliability metrics, $\bar{R}(n)$ (e.g. MTTF), of k-out-of-n pairs:G Balanced system.

s.t. $n \leq n_{max}$.

In general, we obtain the reliability metrics by using the Monte Carlo simulation-based approximation algorithm introduced in Chapter 4. We only introduce the details related to the reliability optimization problem. As shown in Chapter 7, the probability of the system survival when h failed pairs occur is obtained as

$$P(t)=\left(\prod_{i=1}^{h}\lambda_{i-1}\right)\times\exp\left(-g_h\cdot\lambda_h\cdot t\right)\times\int_{\tau_h=0}^{t}\cdots\int_{\tau_1=0}^{\tau_2}\exp\left(\sum_{i=1}^{h}\alpha_i\tau_i\right)d\tau_1\cdots d\tau_h \qquad (8.1)$$

where λ_i is the failure rate after the i^{th} failure, τ_i is the time of the i^{th} failure, and g_h is the number of operating pairs given there are *h* failed pairs. The coefficients of failure time τ_i are obtained as

$$\alpha_i=\lambda_i g_i-\lambda_{i-1}g_{i-1} \qquad (8.2)$$

where g_i is the is the number of operating pairs given there are i failed pairs.

In addition, we consider the load-sharing effect. The load shared by each pair is $l_p=w_s/\left(n-h-s_h\right)$ where h is the number of failed pairs and s_h is the number of standby pairs given there are *h* failed pairs. We assume that the load affects the hazard rate of each operating pair. Furthermore, we consider the effect of standby. The standby pairs do not share the load when they are in the standby state; hence the failure rates are not affected. When resuming operation, the standby pairs have lower hazard rates than the other pairs. Subsequently, the system has a lower overall hazard rate. Assigning different pairs with different hazard rates complicates the problem modeling significantly. Instead of modifying the hazard rates of specific pairs, we assume that the hazard rates of all operating pairs are the same, and modify the common hazard rate to model the effect of standby on the overall system hazard rate.

We obtain the hazard rate after the h^{th} failure by combining the effects of both load-sharing and standby. For example, the hazard rate after the h^{th} failure can be described by Eq. (8.3).

$$\lambda_h=\lambda_o\times\xi_1\times\xi_2 \qquad (8.3)$$

where λ_o is the baseline hazard rate, ξ_1 is the effect of load-sharing, e.g.

$$\xi_1 = \exp\left(\alpha\left(-1+\beta\times l_p{}^{\theta}\right)\right) \tag{8.4}$$

with coefficients α, β and θ; and ξ_2 is the effect of standby, e.g.

$$\xi_2 = \prod_{i=0}^{h-1}\left(1-s_i/\left(n-i\right)\right)^{1/g_i} \tag{8.5}$$

where s_i is the number of standby pairs and g_i is the number of operating pairs given there are i failed pairs.

8.2.1.2 *Optimization Approach*

Indeed, the objective function of the optimization problem is highly nonlinear and cannot be solved by analytical optimization algorithms but by enumerating all the possible solutions. Fortunately, some observations may simplify the enumeration process. As previously mentioned, k is partially determined by n. Let the ratio between the capacity of a pair and the load it adds to the system be $r_{c2w} = c_p / w_p$, it follows that $k = \left\lceil w_o/c_p + n/r_{c2w} \right\rceil$. k is indeed a function of n and the increment slope of k is determined by the value of r_{c2w}.

We observe that different *n* may correspond to the same *k*. It is true that for two *k*-out-of-*n* pairs:G Balanced systems with the same *k*, the system with a greater *n* has higher reliability since it has more redundancy. It is reasonable to only enumerate the largest *n* value that corresponds to the same *k*. The value of *k* increases linearly with *n* for a number of *n* values, then stays constant for a number of *n* values. This cycle repeats itself. One example is

shown in Figure 8.1. It is observed from numerical experiments that the systems with *n* values at the end of the cycles, annotated as asterisks in Figure 8.1, tend to have higher reliability. Therefore, we enumerate these *n* values first to narrow down the range of further enumeration. We denote the vector composed of these *n* values as $\bar{\mathbf{n}}^*$.

The enumeration is carried out in two phases. In the first phase, we enumerate $\bar{\mathbf{n}}^*$ to narrow down the search range. In this phase, we calculate the reliability metrics until we find the first $\bar{\mathbf{n}}^*(i)$ such that $\bar{R}(\bar{\mathbf{n}}^*(i)) \geq \bar{R}(\bar{\mathbf{n}}^*(j))$ for $j = (i+1)$ to $(i+m)$, or until we reach a $\bar{\mathbf{n}}^*(i)$ greater than n_{max}. In other words, the first phase stops when we find the first $\bar{\mathbf{n}}^*(i)$ that corresponds to the maximum reliability metric followed by a certain number, m, of $\bar{\mathbf{n}}^*(j)$ values that correspond to smaller reliability metric values, or when we exhaust the entire search range. We now narrow the range down to $[\bar{\mathbf{n}}^*(i-1), \min(\bar{\mathbf{n}}^*(i+1), n_{max})]$. In the second phase, we search the narrowed range to find the *n* value that corresponds to the maximum reliability metrics, denoted as n^*.

When we approximate the reliability metrics of the systems, we set the number of simulation runs between *n* and 2*n* to achieve high accuracy. In the first phase of enumeration, we can reduce the number of simulation runs to shorten the computation time. We increase the number of simulation runs after we find a promising search range.

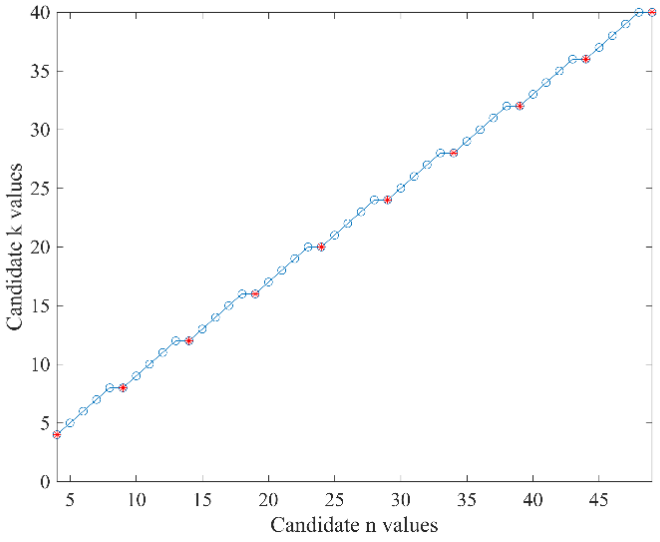


Figure 8.1 The pattern of k increasing with n

8.2.2 *Optimal Number of Standby Pairs*

8.2.2.1 *Problem Modeling*

The problem is described as follows:

Maximize Reliability metrics, $\bar{R}(n, K_i)$, of *k*-out-of-*n* pairs:G Balanced system.

s.t. $n \in$ the values determined by the previous optimization procedure;

$0 \le K_i \le \min\{n-k-i, K_{i,max}\}$ for $i = 0, \cdots, (n-k)$.

It is possible that we need more than K_i standby pairs to balance an unbalanced system in some cases. We allow δ_i additional standby pairs where $0 \le \delta_i \le n-k-i-K_i$, the number of standby pairs is thus $(K_i + \delta_i)$.

Here we introduce the approach for system reliability estimation. We find that the locations of the standby pairs are irrelevant to the probability of any successful event calculated by Eq. (8.1). Eq. (8.2) indicates that α_i's are determined by the number of operating pairs and hazard rates, λ_i. In addition, Eq. (8.3) to Eq. (8.5) indicate that the hazard rates are determined by the number of standby pairs and operating pairs. It follows that the probability of any successful event is determined by the number of standby pairs given the number of failed pairs.

On the other hand, the system states and the set of successful events are determined by the locations of the standby pairs given the locations of failed pairs since standby pairs do not fail during the standby period. Therefore, the locations of standby pairs should also be considered as decision variables. However, this would make the problem intractable since the locations of standby pairs have to be determined for all the possible states. To simplify the problem, we approximate the reliability function using the following procedure.

Step 1. We determine all the possible numbers of standby pairs to keep a balanced system given the number of failed pairs. We enumerate all of the possible system states and the corresponding numbers of realizations based on the procedure in Chapter 3. For each system state with h failed pairs where $h = 0$ to $(n - k)$, we enumerate all the candidate axes of symmetry and determine the corresponding minimum numbers of standby pairs needed to make the operating pairs symmetric to the candidate axes. We then obtain all of the possible numbers of standby pairs for that system state by the following rules:

(i) When a system state with *h* failed pairs can be balanced with s_h standby pairs, it also can be balanced with $(s_h + 2\times i)$ standby pairs where $i \in \{0,1,\cdots,\lfloor (n-k-h-s_h)/2 \rfloor\}$ since a system can maintain symmetry if any two pairs that are symmetric w.r.t. the axis of symmetry are forced down.

(ii) When a system has only one failed pair or consecutive failed pairs, it can be balanced by 0 to ($n - k - h$) standby pairs since we can arrange all the standby pairs next to the failed pairs consecutively and a system with consecutive failed and standby pairs is indeed balanced.

(iii) When there are two failed pairs, and there is an odd number of operating pairs in between, the system can have 0 to ($n - k - h$) standby pairs.

Step 2. We obtain the probability that at least δ_h additional standby pairs are required to regain system balance given that there are *h* failed pairs in the system, and at least K_h standby pairs are set to be forced down. We denote the probability as $p_{\delta_h|h,K_h}$. For example, a 2-out-of-6 pairs:G Balanced system with one failure can be balanced without a standby pair. Here $n=6$, $k=2$, $h=1$ and hence $K_1 \in \{0,1,2,3\}$. As mentioned previously, a system with consecutive failed and standby pairs is balanced. Therefore, no additional standby pairs are needed, i.e. $\delta_1 \equiv 0$, since we can always arrange all the standby pairs next to the failed pairs consecutively. Here $p_{0|1,0} = p_{0|1,1} = p_{0|1,2} = p_{0|1,3} = 1$. But when a 2-out-of-6 pairs:G Balanced system has 3 failed pairs, it is possible that the system is unbalanced and at least one standby pair is required to regain system balance. We found that the system might have four different states: two of them need at least one standby pair and have 6

realizations, respectively; the other two states are balanced without standby and have 2 and 6 realizations, respectively. Thus, we obtain $p_{0|3,0} = (2+6)/(6+6+2+6) = 0.4$, $p_{1|3,0} = (6+6)/(6+6+2+6) = 0.6$ and $p_{0|3,1} = 1$ and $p_{1|3,1} = 0$. In fact, when $K_3 = 1$, δ_3 can only be zero since $0 \le \delta_3 \le n-k-3-K_3 = 0$.

Step 3. The reliability can be obtained by Eq. (8.6)

$$R(t) = \sum_{h=0}^{n-k} \sum_{i=0}^{h} \sum_{\delta_i=0}^{n-k-i-K_i} \left\{ \begin{array}{l} \left(\prod_{i=0}^{h-1} \left(n-i-K_i-\delta_i \right) \right) \times \left(\prod_{i=0}^{h} p_{\delta_i|i,K_i} \right) \times \left(\prod_{i=1}^{h} \lambda_{i-1|K_{i-1},\delta_{i-1}} \right) \\ \times \exp\left[-t \bullet \lambda_{h|K_h,\delta_h} \left(n-h-\left(K_h+\delta_h \right) \right) \right] \times \\ \int_{\tau_h=0}^{t} \ldots \int_{\tau_1=0}^{\tau_2} \exp\left(\sum_{i=1}^{h} \alpha_{i|K_i,\delta_i} \tau_i \right) d\tau_1 \ldots d\tau_h \end{array} \right\} \quad (8.6)$$

where $\lambda_{i|K_i,\delta_i}$ is the failure rate of individual pairs after the i^{th} failure given there are $\left(K_i + \delta_i \right)$ standby pairs after the i^{th} failure; and $\alpha_{i|K_i,\delta_i}$ is the value of α_i given there are $\left(K_i + \delta_i \right)$ standby pairs after the i^{th} failure. Specifically $\lambda_{i|K_i,\delta_i}$ is a function of the number of operating pair $g_i = (n-i-K_i-\delta_i)$, e.g. as described in Eq. (8.3).

8.2.2.2 *Optimization Approach*

The problem can be solved by enumerating all possible combinations for n and K_i ($i = 0$ to $\left(n-k \right)$) when the number of combinations is small or by using heuristics, such as Genetic Algorithms, when the number of combinations is large.

8.2.3 *Numerical Examples*

8.2.3.1 *Numerical Example 1*

We use the model assuming the hazard rate is described by Eq. (8.3) to Eq. (8.5). Let $\lambda_o = 0.01,\ w_o = 1,\ w_p = 1,\ r_{c2w} = 1.25,\ \alpha = 1,\ \beta = 1,\ \theta = 1.$ In addition, we set $m = 5$ and $n_{max} = 50$. We search for the n that maximizes the MTTF of the k-out-of-n pairs:G Balanced system by assuming the minimum number of standby is always 0, i.e. $K_h \equiv 0$. The maximum MTTF (19.81) is obtained when $n = 14$, as shown in Figure 8.2 where the circles and asterisks represent the MTTF values obtained in the first and second enumeration phases, respectively. The corresponding k values are shown in Figure 8.1. The value of k is 12 when $n = 14$.

8.2.3.2 *Numerical Example 2*

When the optimal value for n is determined under the assumption that $K_h \equiv 0$, we can optimize the system reliability metrics by searching the optimal values for K_h. We use the parameter settings in numerical example 1 and $n^* = 14$. We search for the optimal K_h and the maximum reliability metrics when $n \in [9,19]$ which is a narrowed range determined by observing Figure 8.2 and $n^* = 14$. The plot of MTTF against n is as shown in Figure 8.3 which indicates that the optimal n is still $n^* = 14$. The optimal standby sequence, K_h for $h = 0$, 1 and 2, is [1, 0, 0]. The optimal MTTF is 19.83 which is slightly higher than the optimal MTTF when $K_h \equiv 0$ in numerical example 1.

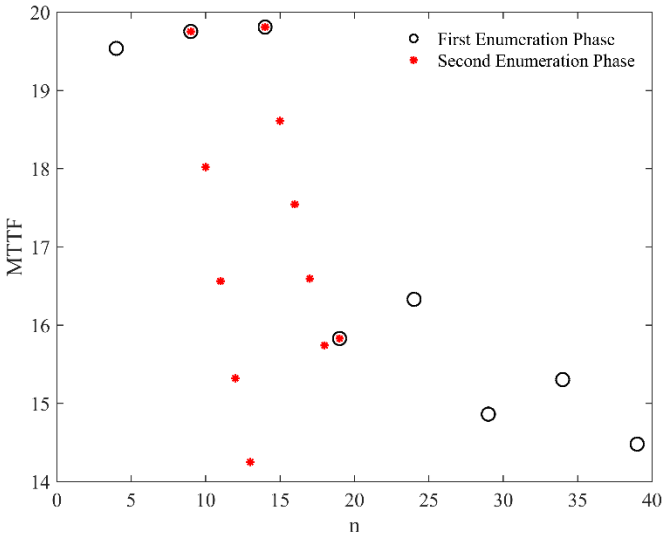


Figure 8.2 Plot of MTTF against n when searching for the optimal value for n

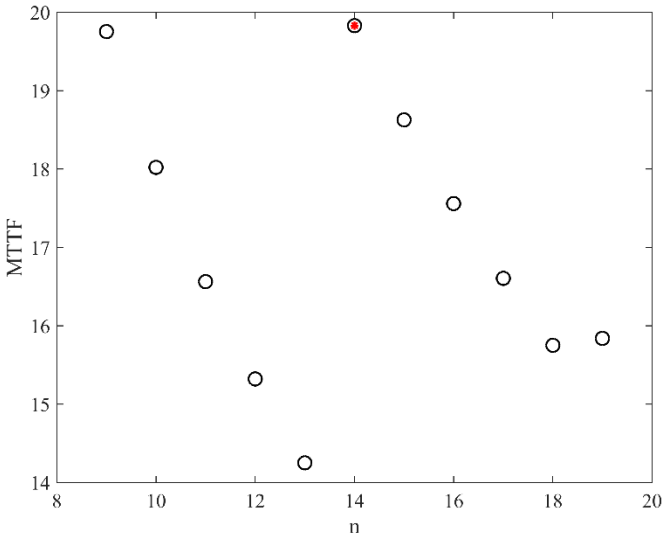


Figure 8.3 Plot of MTTF against n when searching for the optimal values for both n and K_h

8.3 Optimal Reliability Allocation

8.3.1 *Problem Description*

Given the total number of pairs in the system, and the constraints on the number of pairs with different lifetime distributions, we optimize the system reliability by allocating the pairs to different locations in the system. We assume that all the pairs are identical and perform the same function.

Suppose we have M types of pairs. The lifetimes of all of the type i pairs follow the same distribution with *pdf* f_i, a scale parameter a_i, and a shape parameter b_i. The upper bound for the total number of type i pairs is u_i. The minimum number of type i pairs required in the system is l_i. The optimization problem can be described as follows:

Objective:	Maximize reliability metrics of the *k*-out-of-*n* pairs:G Balanced system.
Decision variables:	y_j, the type of pair at location j of the system.
Constraints:	$l_i \le n_i \le u_i$ for $1 \le i \le M$;
	$\sum_{i=1}^{M} n_i = n$ where $n_i := \sum_{j=1}^{n} I\left(y_j = i\right)$ is the number of type i pairs used in the system, and $I\left(y_j = i\right) = 1$ if $y_j = i$ and 0 otherwise.

Note that we use the same numbering method for the locations of the pairs as in Chapter 1: We number the pairs anticlockwise starting from the horizontal pair when we position one of the pairs horizontally.

8.3.2 *Objective Function Estimation*

Since the lifetime distributions of different types of pairs are not identical, the approximation method used previously cannot be applied. Therefore, we propose a new simulation-based reliability approximation method for k -out-of- n pairs:G Balanced system.

8.3.2.1 *Simulation Method without Considering Time-Variant Effects*

In this section, we introduce the simulation procedure to obtain the system reliability without considering time-variant effects such as load-sharing effect and standby effect. Let h denote the number of failed pairs, Z_h , X_h , and Y_h respectively denote the sets of operating pairs, failed pairs, and standby pairs when there are h failed pairs, $t_{h,i}$ denote the randomly generated failure time for the i^{th} pair after the h^{th} failure, and τ_h denote the time of the h^{th} failure. In the following procedure, Y_h is determined by X_h based on the axis of symmetry algorithm introduced in Chapter 3. We also record the results into a table of X_h and its corresponding Y_h to reduce the computation time. The simulation procedure is as follows:

(i) When $h=0$, $Z_0=\{1,2,\cdots,n\}$, $X_0=\varnothing$, $Y_0=\varnothing$. Generate the failure times, $t_{i,0}$, for all the operating pairs, i.e. for $i\in Z_0$.

(ii) When $h=1$, $t_{i,1}=t_{i,0}$ for $i\in Z_0$. $j_1=\arg\min_{i\in Z_0}\{t_{i,1}\}$. $\tau_1=t_{j_1,1}$. $X_1=X_0\cup\{j_1\}$, Y_1 is determined by X_1, and $Z_1=Z_0-X_1-Y_1$.

(iii)When $h=2$, $t_{i,2}=t_{i,1}$ for $i\in Z_1$. $j_2=\arg\min_{i\in Z_1}\{t_{i,2}\}$. $\tau_2=t_{j_2,2}$. $t_{i,2}=t_{i,1}+(\tau_2-\tau_1)$ for $i\in Y_1$. $X_2=X_1\cup\{j_2\}$, Y_2 is determined by X_2, and $Z_2=Z_0-X_2-Y_2$.

(iv)For an arbitrary h, $t_{i,h}=t_{i,h-1}$ for $i\in Z_{h-1}$. $j_h=\arg\min_{i\in Z_{h-1}}\{t_{i,h}\}$. $\tau_h=t_{j_h,h}$. $t_{i,h}=t_{i,h-1}+(\tau_h-\tau_{h-1})$ for $i\in Y_{h-1}$. $X_h=X_{h-1}\cup\{j_h\}$, Y_h is determined by X_h, and $Z_h=Z_0-X_h-Y_h$.

The procedure stops when the system fails, i.e. when the size of Z_h is less than k. Note that the system can have at most $(n-k)$ failures. Therefore, each procedure can result in $(n-k+1)$ failure times. However, when the system has less than $(n-k+1)$ failures but needs to be rebalanced by forcing down additional operating pairs, the number of operating pairs may be less than k, hence the system fails. In this case, the procedure results in less than $(n-k+1)$ failure times. System's failure time is always the last failure time generated in any procedure.

We implement the procedure for N simulation runs to obtain N samples of system failure time and consequently the approximation of system reliability.

8.3.2.2 *Simulation Method Considering Time-Variant Effects*

In this section, we introduce the simulation procedure to obtain the system reliability when time-variant effects such as load-sharing effect are considered. We assume that the shape parameter of a distribution is not affected by the time-varying effects. The load-sharing effect and standby effect on the scale parameter can be modeled by Eq. (8.3) to Eq. (8.5) and by considering the reciprocal of the scale parameter as λ. Let $\delta_{i,h}$ denote the cumulative time by the h^{th} failure during which the i^{th} pair is in standby. The simulation procedure is as follows:

(i) When $h=0$, $Z_0=\{1,2,\cdots,n\}$, $Y_0=\varnothing$, $X_0=\varnothing$. $\delta_{i,0}=0$ for $\forall i$.

(ii) For an arbitrary h, we determine $f_{i,h-1}(t)$, the lifetime distribution of the i^{th} pair given $(h-1)$ pairs have failed, by system states and the previous $(h-1)$ failures. Generate $t_{i,h-1}$ by $f_{i,h-1}(t)$ for $i\in Z_{h-1}$. $j_h=\arg\min_{i\in Z_{h-1}}\{t_{i,h-1}\}$. $\tau_h=t_{j_h,h-1}$. $X_h=X_{h-1}\cup\{j_h\}$, Y_h is determined by X_h, and $Z_h=Z_0-X_h-Y_h$. $\delta_{i,h}=\delta_{i,h-1}$ for $i\in Z_{h-1}$ and $\delta_{i,h}=\delta_{i,h-1}+\tau_h-\tau_{h-1}$ for $i\in Y_{h-1}$.

Similarly, the procedure stops when the system fails, i.e. when the size of Z_h is less than k.

8.3.3 *Optimization Approach*

The decision variables can be described by a vector of n elements. The j^{th} element y_j is the type of pairs at location j of the system, i.e. an integer. The combinations of y_j can easily reach a large number. Therefore, we use Genetic Algorithm to obtain the optimal solution.

8.3.4 *Numerical Example*

In this example, we consider *k*-out-of-6 pairs:G Balanced systems with k = 1 and 2. We consider two types of pairs. Each type of pairs is composed of two identical units with Weibull distributed lifetimes having a scale parameter a_i and a shape parameter b_i. We let $a_1 = 40$, $a_2 = 60$, $b_1 = 3$, and $b_2 = 1$. Type 1 units have shorter expected lifetime and less variance, whereas the type 2 units have greater expected lifetime and greater variance. We let the lower and upper bounds for the number of the type i pairs be $l_i = 0$ and $u_i = 4$. We carry out a few numerical experiments to observe if there exists any rule or pattern for the optimal solutions. In each experiment, we choose an objective function and a value for k. Options for objective function include MTTF, the coefficient of variation of time to failure (CVTTF), i.e. the ratio between the mean and standard deviation of the time to failure, and reliability at the end of the mission time, respectively. The value for k is either 1 or 2.

Table 8.1 Numerical experiment results

Objective Function	*k*	Optimal Solution	Symmetry
MTTF	1	[1 2 2 1 2 2]	Yes
	2	[1 2 1 2 2 2]	Yes
CVTTF	1	[2 1 1 2 1 1]	Yes
	2	[1 1 1 1 2 2]	Yes
Reliability at the end of the mission time (40)	1	[1 2 2 1 2 2]	Yes
	2	[1 2 1 2 2 2]	Yes

Table 8.1 shows the optimal solutions for the reliability allocation problems with different objectives. Any solution is a vector with each element being the type of pair allocated to the corresponding location. The optimal solutions tend to include more type 2 pairs when the objective function is either MTTF or reliability at the end of the mission time, whereas the solutions tend to include more type 1 pairs when the objective function is CVTTF.

In addition, we examine the symmetry of the allocation. An allocation is symmetric if there is an axis of symmetry for either type of the pairs. Based on the results in Table 8.1 we find that the optimal allocation is indeed symmetric.

8.4 Conclusions

This is the first research that investigates the reliability optimization for *k*-out-of-*n* pairs:G Balanced system. The objective function, i.e. the system reliability metric, is calculated by approximation methods due to the extensive computation time required to obtain the exact value.

First, we determine the optimal *n* by fixing the numbers of standby, i.e. $K_h \equiv 0$ for $h = 0$ to $(n - k)$. Second, we obtain the optimal K_h by referring to the optimal *n* obtained in the first phase. Third, given the value of *n*, we determine the optimal allocation of pairs with different lifetime distributions.

CHAPTER 9

CONCLUSIONS AND FUTURE RESEARCH

9.1 Conclusions

The major contribution of this dissertation is the investigation of methodologies for estimating the reliability of systems with spatially distributed units, or spatial systems in short. The function of any spatial system depends on the locations of the operating units and/or failed units. For some spatial systems, both the sequence and locations of failures play an important role in system function. In this dissertation, we focus on an emerging spatial system, k-out-of-n pairs:G Balanced system, and its variant weighted-c-out-of-n pairs:G Balanced system. We develop procedures for estimating and approximating the reliability of such systems. We also investigate the effect of degradation and load-sharing on system reliability. Furthermore, we investigate the optimal reliability design for such systems. The research can be easily extended to other spatial systems.

More specifically, we first investigate the reliability estimation and approximation for multiple k-out-of-n pairs:G Balanced systems. We propose procedures of enumerating successful events, derive a closed form expression of event probabilities, and obtain the system reliability. To estimate the system reliability in a short computational time, especially when the system is large with many units, we propose Monte Carlo simulation-based approximation algorithms. Numerical examples show that the approximation is of

high accuracy. The basic idea of the algorithms can be easily extended to other spatial systems.

Second, we propose a degradation model for spatially distributed units by considering the operating conditions and the dependence among the units. The real-time system reliability of k-out-of-n pairs:G Balanced systems is obtained by analyzing the degradation data of individual units at a system level. We also generalize the degradation model to be used for other spatial systems.

Third, we consider the capacities of the units in the system and investigate the reliability estimation for weighted-c-out-of-n pairs:G Balanced systems. We first investigate the scenario when each unit has multi-state capacity levels. We then examine the effect of load-sharing on system reliability.

Finally, we investigate the optimal reliability design for k-out-of-n pairs:G Balanced systems by considering the effects of load-sharing and standby. The optimal design includes three phases: the first phase determines the optimal value for n, the second phase finds the optimal number of standby pairs given the number of failed pairs, and the third phase allocates pairs with different reliability features to different locations in the system.

9.2 Future Research

The following problems can be explored as future research:

Problem 1: In Chapter 4, we develop reliability approximation algorithms for k-out-of-n pairs:G Balanced systems given the lifetime distributions of individual units. In the future, we will investigate the reliability approximation methods for such systems by considering the degradation processes of individual units.

Problem 2: In Chapter 7, we investigate the load-sharing effect on system reliability. We observe that the system has higher reliability when the load is proportionally distributed to units according to its capacities than when it is equally distributed. This interesting observation suggests some future research on optimal load-sharing rules for weighted-c-out-of-n pairs:G Balanced systems.

REFERENCES

[1] Y. Zhang, "Modeling the effects of the two stochastic-processes on the reliability and maintenance of *k*-out-of-*n* surveillance systems," Doctor of Philosophy Dissertation, Industrial and Systems Engineering, Rutgers, The State University of New Jersey, New Brunswick, New Jersey, 2014.

[2] D. T. Chiang and S.-C. Niu, "Reliability of consecutive-*k*-out-of-*n*:F system," *Reliability, IEEE Transactions on,* vol. R-30, pp. 87-89, 1981.

[3] C. Derman, G. J. Lieberman, and S. M. Ross, "On the consecutive-*k*-of-*n*:F system," *Reliability, IEEE Transactions on,* vol. R-31, pp. 57-63, 1982.

[4] H. Sarper and W. J. Sauer, "New reliability configuration for large planetary descent vehicles," *Journal of Spacecraft and Rockets,* vol. 39, pp. 639-642, 2002.

[5] P. Mantovan and P. Secchi, *Complex data modeling and computationally intensive statistical methods*. Milan: Springer, 2010.

[6] J. F. Fajardo, A. A. Juan, S. S. M. Alsina, and J. E. Ramirez-Marquez, *Simulation methods for reliability and availability of complex systems*: Springer Science & Business Media, 2010.

[7] A. A. Salvia and W. C. Lasher, "Two-dimensional consecutive-*k*-out-of-*n*:F models," *Reliability, IEEE Transactions on,* vol. 39, pp. 382-385, 1990.

[8] M. Gharib, E. M. E. Sayed, and I. I. H. Nashwan, "Reliability of connected (1; 1; 2) or (1; 2; 1) or (2; 1; 1)-out-of-(*n*; 2; 2):F lattice systems," *Journal of Advanced Research in Statistics and Probability,* vol. 3, pp. 47-56, 2011.

[9] T. K. Boehme, A. Kossow, and W. Preuss, "A generalization of consecutive-*k*-out-of-*n*:F systems," *Reliability, IEEE Transactions on,* vol. 41, pp. 451-457, 1992.

[10] M. T. Chao, J. C. Fu, and M. V. Koutras, "Survey of reliability studies of consecutive-*k*-out-of-*n*:F and related systems," *Reliability, IEEE Transactions on,* vol. 44, pp. 120-127, 1995.

[11] M. V. Koutras, G. K. Papadopoulos, and S. G. Papastavridis, "Reliability of 2-dimensional consecutive-*k*-out-of-*n*:F systems," *Reliability, IEEE Transactions on,* vol. 42, pp. 658-661, 1993.

[12] M. V. Koutras, G. K. Papadopoulos, and S. G. Papastavridis, "A reliability bound for 2-dimensional consecutive *k*-out-of-*n*:F systems," *Nonlinear Analysis: Theory, Methods & Applications,* vol. 30, pp. 3345-3348, 1997.

[13] H. Yamamoto and M. Miyakawa, "Reliability of a linear connected-(*r*, *s*)-out-of-(*m*, *n*):F lattice system," *Reliability, IEEE Transactions on,* vol. 44, pp. 333-336, 1995.

[14] F. S. Makri and Z. M. Psillakis, "Bounds for reliability of *k*-within connected-(*r*, *s*)-out-of-(*m*, *n*) failure systems," *Microelectronics Reliability,* vol. 37, pp. 1217-1224, 1997.

[15] A. P. Godbole, L. K. Potter, and J. K. Sklar, "Improved upper bounds for the reliability of *d*-dimensional consecutive-*k*-out-of-*n*:F systems," *Naval Research Logistics (NRL),* vol. 45, pp. 219-230, 1998.

[16] Y. C. Hsieh and T. C. Chen, "Reliability lower bounds for two-dimensional consecutive-*k*-out-of-*n*:F systems," *Comput. Oper. Res.*, vol. 31, pp. 1259-1272, 2004.

[17] X. Zhao, L. Cui, W. Zhao, and F. Liu, "Exact reliability of a linear connected-(*r*, *s*)-out-of-(*m*, *n*):F system," *Reliability, IEEE Transactions on*, vol. 60, pp. 689-698, 2011.

[18] D. Lin and M. J. Zuo, "Reliability evaluation of a linear *k*-within-(*r*, *s*)-out-of-(*m*, *n*):F lattice system," *Probab. Eng. Inf. Sci.*, vol. 14, pp. 435-443, 2000.

[19] T. Akiba and H. Yamamoto, "Reliability of a 2-dimensional *k*-within-consecutive-*r*×*s*-out-of-*m*×*n*:F system," *Naval Research Logistics (NRL)*, vol. 48, pp. 625-637, 2001.

[20] A. S. Habib, T. Yuge, R. O. Al-Seedy, and S. I. Ammar, "Reliability of a consecutive (*r*, *s*)-out-of-(*m*, *n*):F lattice system with conditions on the number of failed components in the system," *Applied Mathematical Modelling*, vol. 34, pp. 531-538, 2010.

[21] M. Boushaba and N. Ghoraf, "A 3-dimensional consecutive-*k*-out-*n*:F models," *International Journal of Reliability, Quality and Safety Engineering*, vol. 09, pp. 193-198, 2002.

[22] M. Boushaba and Z. Azouz, "Reliability bounds of a 3-dimensional consecutive-*k*-out-of-*n*:F system," *International Journal of Reliability, Quality and Safety Engineering*, vol. 18, pp. 51-59, 2011.

[23] M. G. Kulkarni and A. S. Kashikar, "Signature and reliability of conditional three-dimensional consecutive-(*s*, *s*, *s*)-out-of-(*s*, *s*, *m*):F system," *International Journal of Reliability, Quality and Safety Engineering*, vol. 21, p. 1450009, 2014.

[24] T. Akiba, H. Yamamoto, and Y. Kainuma, "Reliability of a 3-dimensional adjacent triangle:F triangular lattice system," in *Proceedings of the 5th Asia Pacific Industrial Engineering and Management Systems Conference*, 2004, pp. 27.8.1-27.8.10.

[25] T. Akiba, H. Yamamoto, and Y. Tsujimura, "Evaluating methods for the reliability of a three-dimensional *k*-within system," *Journal of Quality in Maintenance Engineering*, vol. 11, pp. 254-266, 2005.

[26] L. Geiger. (2013, 4/29). *Unmanned aircraft fly local skies*. Available: http://www.traverseticker.com/story/unmanned-aircraft-fly-local-skies

[27] K. K. Bhamidipati, D. Uhlig, and N. Neogi, "Engineering safety and reliability into UAV systems: mitigating the ground impact hazard," in *AIAA Guidance, Navigation and Control Conference and Exhibit*, 2007.

[28] S. d'Oleire-Oltmanns, I. Marzolff, K. Peter, and J. Ries, "Unmanned aerial vehicle (UAV) for monitoring soil erosion in morocco," *Remote Sensing*, vol. 4, pp. 3390-3416, 2012.

[29] D. o. D. U. S. Military, *21st century unmanned aerial vehicles (UAV) reliability study - predator, pioneer, hunter, UAS - power, propulsion, flight control, communication, human factors*: Progressive Management, 2010.

[30] H. Sarper, "Reliability analysis of descent systems of planetary vehicles using bivariate exponential distribution," in *Proc. Reliability and Maintainability Symposium*, 2005, pp. 165-169.

[31] F. Filippetti, G. Franceschini, C. Tassoni, and P. Vas, "AI techniques in induction machines diagnosis including the speed ripple effect," *Industry Applications, IEEE Transactions on,* vol. 34, pp. 98-108, 1998.

[32] M. Zaggout, P. Tavner, C. Crabtree, and R. Li, "Detection of rotor electrical asymmetry in wind turbine doubly-fed induction generators," *Renewable Power Generation, IET,* vol. 8, pp. 878-886, 2014.

[33] G. Marola, "On the detection of the axes of symmetry of symmetric and almost symmetric planar images," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 11, pp. 104-108, 1989.

[34] F. M. Frey, A. Robertson, and M. Bukoski, "A method for quantifying rotational symmetry," *New Phytologist,* vol. 175, pp. 785-791, 2007.

[35] S. J. Kamat and M. W. Riley, "Determination of reliability using event-based Monte Carlo simulation," *Reliability, IEEE Transactions on,* vol. R-24, pp. 73-75, 1975.

[36] S. J. Kamat and W. E. Franzmeier, "Determination of reliability using event-based Monte Carlo simulation part II," *Reliability, IEEE Transactions on,* vol. R-25, pp. 254-255, 1976.

[37] T. L. Landers, H. A. Taha, and C. L. King, "A reliability simulation approach for use in the design process," *Reliability, IEEE Transactions on,* vol. 40, pp. 177-181, 1991.

[38] C. Kim and H. K. Lee, "A Monte Carlo simulation algorithm for finding MTBF," *Reliability, IEEE Transactions on,* vol. 41, pp. 193-195, 1992.

[39] Z. M. Psillakis, "A simulation algorithm for computing failure probability of a consecutive-*k*-out-of-*r*-from-*n*:F system," *Reliability, IEEE Transactions on,* vol. 44, pp. 523-531, 1995.

[40] S.-K. Au and J. L. Beck, "Estimation of small failure probabilities in high dimensions by subset simulation," *Probabilistic Engineering Mechanics,* vol. 16, pp. 263-277, 2001.

[41] J. Ching, J. L. Beck, and S. K. Au, "Hybrid Subset Simulation method for reliability estimation of dynamical systems subject to stochastic excitation," *Probabilistic Engineering Mechanics,* vol. 20, pp. 199-214, 2005.

[42] J. Ching, S. K. Au, and J. L. Beck, "Reliability estimation for dynamical systems subject to stochastic excitation using subset simulation with splitting," *Computer Methods in Applied Mechanics and Engineering,* vol. 194, pp. 1557-1579, 2005.

[43] S. K. Au, J. Ching, and J. L. Beck, "Application of subset simulation methods to reliability benchmark problems," *Structural Safety,* vol. 29, pp. 183-193, 2007.

[44] L. S. Katafygiotis and S. H. Cheung, "Application of spherical subset simulation method and auxiliary domain method on a benchmark reliability study," *Structural Safety,* vol. 29, pp. 194-207, 2007.

[45] R. M. Geist and M. K. Smotherman, "Ultrahigh reliability estimates through simulation," in *Annual Reliability and Maintainability Symposium,* 1989, pp. 350-355.

[46] H. Kumamoto, K. Tanaka, and K. Inoue, "Efficient Evaluation of System Reliability by Monte Carlo Method," *Reliability, IEEE Transactions on,* vol. R-26, pp. 311-315, 1977.

[47] P. Heidelberger, "Fast simulation of rare events in queueing and reliability models," in *Performance Evaluation of Computer and Communication Systems*. vol. 729, L. Donatiello and R. Nelson, Eds., ed: Springer Berlin Heidelberg, 1993, pp. 165-202.

[48] D. Lieber, A. Nemirovskii, and R. Y. Rubinstein, "A fast Monte Carlo method for evaluating reliability indexes," *Reliability, IEEE Transactions on,* vol. 48, pp. 256-261, 1999.

[49] S. Juneja and P. Shahabuddin, "Splitting-based importance-sampling algorithm for fast simulation of Markov reliability models with general repair-policies," *Reliability, IEEE Transactions on,* vol. 50, pp. 235-245, 2001.

[50] H. Kumamoto, T. Tanaka, and K. Inoue, "A new Monte Carlo method for evaluating system-failure probability," *Reliability, IEEE Transactions on,* vol. R-36, pp. 63-69, 1987.

[51] H. Kumamoto, K. Tanaka, K. Inoue, and E. J. Henley, "Dagger-sampling Monte Carlo for system unavailability evaluation," *Reliability, IEEE Transactions on,* vol. R-29, pp. 122-125, 1980.

[52] H. Cancela and M. El Khadiri, "Series-parallel reductions in Monte Carlo network-reliability evaluation," *Reliability, IEEE Transactions on,* vol. 47, pp. 159-164, 1998.

[53] D. Karger, "A randomized fully polynomial time approximation scheme for the all-terminal network reliability problem," *SIAM Review,* vol. 43, pp. 499-522, 2001.

[54] M. C. Easton and C. K. Wong, "Sequential destruction method for Monte Carlo evaluation of system reliability," *Reliability, IEEE Transactions on,* vol. R-29, pp. 27-32, 1980.

[55] A. Baca, "Examples of Monte Carlo methods in reliability estimation based on reduction of prior information," *Reliability, IEEE Transactions on,* vol. 42, pp. 645-649, 1993.

[56] A. Naess, B. J. Leira, and O. Batsevych, "System reliability analysis by enhanced Monte Carlo simulation," *Structural Safety,* vol. 31, pp. 349-355, 2009.

[57] Q. Yang and Y. Chen, "Monte Carlo methods for reliability evaluation of linear sensor systems," *Reliability, IEEE Transactions on,* vol. 60, pp. 305-314, 2011.

[58] O. Ditlevsen and P. Bjerager, "Plastic reliability analysis by directional simulation," *Journal of Engineering Mechanics,* vol. 115, pp. 1347-1362, 1989.

[59] J. Nie and B. R. Ellingwood, "Directional methods for structural reliability analysis," *Structural Safety,* vol. 22, pp. 233-249, 2000.

[60] H. Dai, H. Zhang, W. Wang, and G. Xue, "Structural reliability assessment by local approximation of limit state functions using adaptive Markov chain simulation and support vector regression," *Computer-Aided Civil and Infrastructure Engineering,* vol. 27, pp. 676-686, 2012.

[61] S. Rahman and D. Wei, "A univariate approximation at most probable point for higher-order reliability analysis," *International Journal of Solids and Structures,* vol. 43, pp. 2820-2839, 2006.

[62] C. S. Kulkarni, "A physics-based degradation modeling framework for diagnostic and prognostic studies in electrolytic capacitors," Ph.D. dissertation, Department of Electrical Engineering, Vanderbilt University, Nashville, Tennessee, US, 2013.

[63] S. E. Tyaginov, I. Starkov, H. Enichlmair, J. M. Park, C. Jungemann, and T. Grasser, "Physics-based hot-carrier degradation modeling," *ECS Transactions,* vol. 35, pp. 321-352, 2011.

[64] J. W. McPherson, *Reliability physics and engineering: time-to-failure modeling*: Springer, 2010.

[65] S. P. Sinha, F. L. Duan, D. E. Ioannou, W. C. Jenkins, and H. L. Hughes, "Time dependence power laws of hot carrier degradation in SOI MOSFETS," in *1996 IEEE International SOI Conference Proceedings*, 1996, pp. 18-19.

[66] M. A. Alam, H. Kufluoglu, D. Varghese, and S. Mahapatra, "A comprehensive model for PMOS NBTI degradation: recent progress," *Microelectronics Reliability,* vol. 47, pp. 853-862, 2007.

[67] K. K. Saluja, S. Vijayakumar, W. Sootkaneung, and X. Yang, "NBTI degradation: a problem or a scare?," presented at the Proceedings of the 21st International Conference on VLSI Design, 2008.

[68] H. J. Choi, C. A. Kim, J.-I. Sohn, and M. S. Jhon, "An exponential decay function for polymer degradation in turbulent drag reduction," *Polymer Degradation and Stability,* vol. 69, pp. 341-346, 2000.

[69] W. Q. Meeker and M. J. LuValle, "An accelerated life test model based on reliability kinetics," *Technometrics,* vol. 37, pp. 133-146, 1995.

[70] C. D. Doyle, "Logarithmic thermal degradation of a silicone resin in air," *Journal of Polymer Science,* vol. 31, pp. 95-104, 1958.

[71] Z. Yang, R. Kang, and E. A. Elsayed, "Reliability estimate of probabilistic-physics-of-failure degradation model," *Chemical Engineering Transactions,* vol. 33, pp. 499-504, 2013.

[72] W. Q. Meeker and L. A. Escobar, *Statistical methods for reliability data*. New York: John Willey & Sons, Inc, 1998.

[73] L. C. Tang, G. Y. Yang, and M. Xie, "Planning of step-stress accelerated degradation test," presented at the Reliability and Maintainability, 2004 Annual Symposium - RAMS, 2004.

[74] N. Z. Gebraeel, M. A. Lawley, R. Li, and J. K. Ryan, "Residual-life distributions from component degradation signals: a Bayesian approach," *IIE Transactions,* vol. 37, pp. 543-557, 2005.

[75] N. Z. Gebraeel, "Sensory-updated residual life distributions for components with exponential degradation patterns," *Automation Science and Engineering, IEEE Transactions on,* vol. 3, pp. 382-393, 2006.

[76] W. Wang, M. Carr, W. Xu, and K. Kobbacy, "A model for residual life prediction based on Brownian motion with an adaptive drift," *Microelectronics Reliability,* vol. 51, pp. 285-293, 2011.

[77] Y. Wu, L. Xie, N. Wu, and J. Li, "Time-dependent reliability model of components with strength degradation based-on gamma process," presented at the Reliability, Maintainability and Safety (ICRMS), 9th International Conference on, 2011.

[78] M. Marseguerra, E. Zio, and L. Podofillini, "Condition-based maintenance optimization by means of genetic algorithms and Monte Carlo simulation," *Reliability Engineering & System Safety,* vol. 77, pp. 151-165, 2002.

[79] A. Grall, C. Bérenguer, and L. Dieulle, "A condition-based maintenance policy for stochastically deteriorating systems," *Reliability Engineering & System Safety*, vol. 76, pp. 167-180, 2002.

[80] E. Deloux, B. Castanier, and C. Berenguer, "Maintenance policy for a non-stationary deteriorating system," presented at the Reliability and Maintainability Symposium, 2008. RAMS 2008. Annual, 2008.

[81] H. Liao, E. A. Elsayed, and L.-Y. Chan, "Maintenance of continuously monitored degrading systems," *European Journal of Operational Research*, vol. 175, pp. 821-835, 2006.

[82] W. Q. Meeker, L. A. Escobar, and C. J. Lu, "Accelerated degradation tests: modeling and analysis," *Technometrics*, vol. 40, pp. 89-99, 1998.

[83] H. Liao and E. A. Elsayed, "Optimization of system reliability robustness using accelerated degradation testing," in *Annual Reliability and Maintainability Symposium 2005 Proceedings*, 2005, pp. 48-54.

[84] H. Liao, "Degradation models and design of accelerated degradation testing plans," Ph.D. dissertation, Department of Industrial and Systems Engineering, Rutgers, The State University of New Jersey, Piscataway, NJ, 2004.

[85] S. T. Tseng, N. Balakrishnan, and C. C. Tsai, "Optimal step-stress accelerated degradation test plan for gamma degradation processes," *Reliability, IEEE Transactions on*, vol. 58, pp. 611-618, 2009.

[86] X. Liu, K. N. Al-Khalifa, E. A. Elsayed, D. W. Coit, and A. S. Hamouda, "Criticality measures for components with multi-dimensional degradation," *IIE Transactions*, vol. 46, pp. 987–998, 2014.

[87] L. A. Escobar and W. Q. Meeker, "A review of accelerated test models," *Statistical Science*, vol. 21, pp. 552-577, 2006.

[88] W. Nelson, *Accelerated testing: statistical models, test plans, and data analysis*. New York: John Wiley & Sons, 1990.

[89] L. Wang, T. Jiang, X. Li, and J. Zhang, "An ADT comprehensive evaluation method based on Bayesian," presented at the Industrial Engineering and Engineering Management (IEEM), 2010 IEEE International Conference on, 2010.

[90] S. L. Jeng, B. Y. Huang, and W. Q. Meeker, "Accelerated destructive degradation tests robust to distribution misspecification," *Reliability, IEEE Transactions on*, vol. 60, pp. 701-711, 2011.

[91] W. Zhao and E. A. Elsayed, "An accelerated life testing model involving performance degradation," presented at the Reliability and Maintainability, 2004 Annual Symposium - RAMS, 2004.

[92] H. Pham, A. Suprasad, and R. B. Misra, "Reliability and MTTF prediction of *k*-out-of-*n* complex systems with components subjected to multiple stages of degradation," *International Journal of Systems Science*, vol. 27, pp. 995-1000, 1996.

[93] S. Song, D. W. Coit, Q. Feng, and H. Peng, "Reliability analysis for multi-component systems subject to multiple dependent competing failure processes," *Reliability, IEEE Transactions on*, vol. 63, pp. 331-345, 2014.

[94] A. Gupta and C. Lawsirirat, "Strategically optimum maintenance of monitoring-enabled multi-component systems using continuous-time jump deterioration models," *Journal of Quality in Maintenance Engineering*, vol. 12, pp. 306-329, 2006.

[95] L. Bian and N. Gebraeel, "Stochastic modeling and real-time prognostics for multi-component systems with degradation rate interactions," *IIE Transactions,* vol. 46, pp. 470-482, 2014.
[96] M. P. Enright and D. M. Frangopol, "Reliability-based condition assessment of deteriorating concrete bridges considering load redistribution," *Structural Safety,* vol. 21, pp. 159-195, 1999.
[97] P. S. Marsh and D. M. Frangopol, "Reinforced concrete bridge deck reliability model incorporating temporal and spatial variations of probabilistic corrosion rate sensor data," *Reliability Engineering & System Safety,* vol. 93, pp. 394-409, 2008.
[98] E. El-Neweihi, F. Proschan, and J. Sethuraman, "Multistate coherent systems," *Journal of Applied Probability,* vol. 15, pp. 675-688, 1978.
[99] Z. Tian, M. J. Zuo, and R. C. M. Yam, "Multi-state *k*-out-of-*n* systems and their performance evaluation," *IIE Transactions,* vol. 41, pp. 32-44, 2008.
[100] R. A. Boedigheimer and K. C. Kapur, "Customer-driven reliability models for multistate coherent systems," *IEEE Transactions on Reliability,* vol. 43, pp. 46-50, 1994.
[101] J. Huang, M. J. Zuo, and Y. Wu, "Generalized multi-state *k*-out-of-*n*:G systems," *IEEE Transactions on Reliability,* vol. 49, pp. 105-111, 2000.
[102] M. J. Zuo and Z. Tian, "Performance evaluation of generalized multi-state *k*-out-of-*n* systems," *IEEE Transactions on Reliability,* vol. 55, pp. 319-327, 2006.
[103] Z. Tian, W. Li, and J. M. Zuo, "Modeling and reliability evaluation of multi-state *k*-out-of-*n* systems," in *Recent Advances in Reliability and Quality in Design*, H. Pham, Ed., ed London: Springer London, 2008, pp. 31-56.
[104] Z. Tian, R. C. M. Yam, M. J. Zuo, and H. Z. Huang, "Reliability bounds for multi-state *k*-out-of-*n* systems," *IEEE Transactions on Reliability,* vol. 57, pp. 53-58, 2008.
[105] X. Zhao and L. Cui, "Reliability evaluation of generalised multi-state *k*-out-of-*n* systems based on FMCI approach," *Intern. J. Syst. Sci.,* vol. 41, pp. 1437-1443, 2010.
[106] S. K. Chaturvedi, S. H. Basha, S. V. Amari, and M. J. Zuo, "Reliability analysis of generalized multi-state *k*-out-of-*n* systems," *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability,* vol. 226, pp. 327-336, 2012.
[107] S. Eryilmaz, "Lifetime of multistate *k*-out-of-*n* systems," *Quality and Reliability Engineering International,* vol. 30, pp. 1015-1022, 2014.
[108] Y. Mo, L. Xing, S. V. Amari, and J. Bechta Dugan, "Efficient analysis of multi-state *k*-out-of-*n* systems," *Reliability Engineering & System Safety,* vol. 133, pp. 95-105, 2015.
[109] J. Huang, M. J. Zuo, and Z. Fang, "Multi-state consecutive-*k*-out-of-*n* systems," *IIE Transactions,* vol. 35, pp. 527-534, 2003.
[110] M. J. Zuo, Z. Fang, J. Huang, and X. Xu, "Performance evaluation of decreasing multi-state consecutive-*k*-out-of-*n*:G systems," *International Journal of Reliability, Quality and Safety Engineering,* vol. 10, pp. 345-358, 2003.
[111] S. Belaloui and B. Ksir, "Reliability of a multi-state consecutive *k*-out-of-*n*:G system," *International Journal of Reliability, Quality and Safety Engineering,* vol. 14, pp. 361-377, 2007.

[112] J. S. Wu and R. J. Chen, "An algorithm for computing the reliability of weighted-*k*-out-of-*n* systems," *Reliability, IEEE Transactions on,* vol. 43, pp. 327-328, 1994.
[113] J. S. Wu and R. J. Chen, "Efficient algorithms for *k*-out-of-*n* and consecutive-weighted-*k*-out-of-*n*:F system," *Reliability, IEEE Transactions on,* vol. 43, pp. 650-655, 1994.
[114] J. C. Chang, R. J. Chen, and F. K. Hwang, "A fast reliability-algorithm for the circular consecutive-weighted-*k*-out-of-*n*:F system," *Reliability, IEEE Transactions on,* vol. 47, pp. 472-474, 1998.
[115] S. Eryilmaz, "On reliability analysis of a *k*-out-of-*n* system with components having random weights," *Reliability Engineering & System Safety,* vol. 109, pp. 41-44, 2013.
[116] K. K. Kamalja and K. P. Amrutkar, "Computational methods for reliability and importance measures of weighted-consecutive-system," *Reliability, IEEE Transactions on,* vol. 63, pp. 94-104, 2014.
[117] S. Eryilmaz, "Capacity loss and residual capacity in weighted *k*-out-of-*n*:G systems," *Reliability Engineering & System Safety,* vol. 136, pp. 140-144, 2015.
[118] P. H. Kvam and E. A. Peña, "Estimating load-sharing properties in a dynamic reliability system," *Journal of the American Statistical Association,* vol. 100, pp. 262-272, 2005.
[119] S. Lee, S. Durham, and J. Lynch, "On the calculation of the reliability of general load sharing systems," *Journal of Applied Probability,* vol. 32, pp. 777-792, 1995.
[120] S. V. Amari, K. B. Misra, and H. Pham, "Reliability analysis of tampered failure rate load-sharing *k*-out-of-*n*:G systems," in *12th ISSAT Int. Conf. on Reliability and Quality in Design,* Honolulu, Hawaii, 2006, pp. 30-35.
[121] H. Liu, "Reliability of a load-sharing *k*-out-of-*n*:G system: non-iid components with arbitrary distributions," *Reliability, IEEE Transactions on,* vol. 47, pp. 279-284, 1998.
[122] E. M. Scheuer, "Reliability of an *m*-out of-*n* system when component failure induces higher failure rates in survivors," *Reliability, IEEE Transactions on,* vol. 37, pp. 73-74, 1988.
[123] J. Shao and L. R. Lamberson, "Modeling a shared-load *k*-out-of-*n*:G system," *Reliability, IEEE Transactions on,* vol. 40, pp. 205-209, 1991.
[124] H.-H. Lin, K.-H. Chen, and R.-T. Wang, "A multivariant exponential shared-load model," *Reliability, IEEE Transactions on,* vol. 42, pp. 165-171, 1993.
[125] T. F. Hassett, D. L. Dietrich, and F. Szidarovszky, "Time-varying failure rates in the availability and reliability analysis of repairable systems," *Reliability, IEEE Transactions on,* vol. 44, pp. 155-160, 1995.
[126] W. Yamamoto, L. Jin, and K. Suzuki, "Optimal allocations for load-sharing *k*-out-of-*n*:F systems," *Journal of Statistical Planning and Inference,* vol. 139, pp. 1777-1781, 2009.
[127] H. Lin and X. Qiang, "Lifetime reliability for load-sharing redundant systems with arbitrary failure distributions," *Reliability, IEEE Transactions on,* vol. 59, pp. 319-330, 2010.
[128] X. Qi, Z. Zhang, D. Zuo, and X. Yang, "Optimal maintenance policy for high reliability load-sharing computer systems with *k*-out-of-*n*:G redundant structure," *Applied Mathematics & Information Sciences,* vol. 8, pp. 341-347, 2014.

[129] W. Kuo and V. R. Prasad, "An annotated overview of system-reliability optimization," *IEEE Transactions on Reliability,* vol. 49, pp. 176-187, 2000.

[130] D. W. Coit and J. C. Liu, "System reliability optimization with *k*-out-of-*n* subsystems," *International Journal of Reliability, Quality and Safety Engineering,* vol. 07, pp. 129-142, 2000.

[131] A. O. C. Elegbede, C. Chengbin, K. H. Adjallah, and F. Yalaoui, "Reliability allocation through cost minimization," *IEEE Transactions on Reliability,* vol. 52, pp. 106-111, 2003.

[132] Z. Tian, M. J. Zuo, and H. Huang, "Reliability-redundancy allocation for multi-state series-parallel systems," *IEEE Transactions on Reliability,* vol. 57, pp. 303-310, 2008.

[133] W. Kuo and R. Wan, "Recent advances in optimal reliability allocation," *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans,* vol. 37, pp. 143-156, 2007.

[134] P. Boddu and L. Xing, "Redundancy allocation for *k*-out-of-*n*: G systems with mixed spare types," in *Reliability and Maintainability Symposium (RAMS), 2012 Proceedings - Annual*, 2012, pp. 1-6.

[135] M.-S. Chern, "On the computational complexity of reliability redundancy allocation in a series system," *Operations Research Letters,* vol. 11, pp. 309-315, 1992.

[136] F. A. Tillman, C. L. Hwang, and W. Kuo, "Optimization techniques for system reliability with redundancy: a review," *IEEE Transactions on Reliability,* vol. R-26, pp. 148-155, 1977.

[137] F. Altiparmak, B. Dengiz, and A. E. Smith, "Optimal Design of Reliable Computer Networks: A Comparison of Metaheuristics," *Journal of Heuristics,* vol. 9, pp. 471-487, 2003.

[138] Y. Ben-Dov, "Optimal reliability design of *K*-out-of-*N* systems subject to two kinds of failure," *J Oper Res Soc,* vol. 31, pp. 743-748, 1980.

[139] M. Zuo, "Optimal system reliability design of consecutive-*k*-out-of-*n* systems," Ph.D., Industrial Engineering, Iowa State University, Ames, Iowa, 1989.

[140] W. Kuo and M. J. Zuo, *Optimal reliability modeling: principles and applications*: John Wiley & Sons, 2003.

[141] M. J. Zuo, "Reliability and design of 2-dimensional consecutive-*k*-out-of-*n* systems," *Reliability, IEEE Transactions on,* vol. 42, pp. 488-490, 1993.

[142] A. Molini, P. Talkner, G. G. Katul, and A. Porporato, "First passage time statistics of Brownian motion with purely time dependent drift and diffusion," *Physica A: Statistical Mechanics and its Applications,* vol. 390, pp. 1841-1852, 2011.