RELIABILITY ESTIMATION OF BALANCED SYSTEMS
WITH MULTI-DIMENSIONAL DISTRIBUTED UNITS

by

JINGBO GUO

A dissertation submitted to the
School of Graduate Studies
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

January 2022
ABSTRACT OF THE DISSERTATION

Reliability Estimation of Balanced Systems with Multi-dimensional Distributed Units

By JINGBO GUO

Dissertation Director:

Elsayed A. Elsayed

Balanced systems with multi-dimensional distributed units are emerging in a diverse range of industries. This includes Unmanned Aerial Vehicles (UAV) with multi-level of rotary wings, Spherical Unmanned Vehicles (SUV), Spherical Phased Array Antenna (SPAA), etc. In this dissertation, we present the reliability estimation for such systems. In particular, we consider two configurations: 1) balanced systems with units distributed circularly on multi-level and 2) balanced systems with units distributed spherically.

First, balanced systems with units distributed circularly on multi-level are generalized as \((k_1, k_2)\)-out-of-\((n, m)\) pairs: G balanced systems. We consider two scenarios: 1) all units perform the same function and 2) adjacent pairs perform complementary functions. For both scenarios, unbalanced system is considered as failed. When units fail and cause the system imbalance, we explore two approaches to rebalance the system: 1) forcing down units on other locations and 2) resuming units that are previously forced down (if any).
When units in a system perform the same function, operational states are defined as balanced states with at least $k_1$ operating pairs and each operating pair has at least $k_2$ units on each side. The system reliability is obtained by enumerating all of the operational states and summing the probabilities of those states. For $(k_1,k_2)$-out-of-$(n,m)$ pairs: G balanced systems with adjacent pairs performing complementary functions, in addition to maintaining system balance, the adjacent operating pairs are required to perform complementary functions. Thus, if a pair fails, one of the adjacent pairs is forced down. Similarly, the system reliability is obtained by enumerating all of the operational states. It becomes computational expensive when the number of units in each pair and/or the number of pairs are large. In that case, efficient algorithms are developed to obtain the reliability for such systems.

The balanced system with units distributed spherically is generalized as a spherical $k$-n-i: G balanced system. We consider two balancing requirements: 1) rotational balance is maintained so that the system is not rotating w.r.t. roll, yaw and pitch axes and 2) symmetrical balance is essential in improving the systems’ stability. We present mathematical approaches to determine the balance status of a system. Similarly, the unbalanced system is rebalanced by 1) forcing down units on other locations and 2) resuming previously forced-down units. The system reliability is obtained by the enumeration of operational states and calculation of operational states’ probabilities. We develop an efficient algorithm for reliability estimation when the number of units in the system is large.
Degradation models are developed for the \((k_1, k_2)\)-out-of-\((n, m)\) pairs: \(G\) balanced systems to further investigate the system reliability when degradation data are available. The degradation processes of units in the system are either stationary (inverse Gaussian process) or non-stationary (improved inverse Gaussian process). We propose a degradation balance mechanism in which the ‘most’ degraded units are forced down temporarily during the degradation process so that the system is less possible to fail due to imbalance. A closed-form lower bound reliability is presented when the balance mechanism is not applied. When it is applied, reliability is obtained by Monte Carlo simulation.

From the reliability study of both configurations, it is observed that the reliability of a balanced system with multi-dimensional distributed units depends not only on the system’s total number of units and the least number of operating units, but also on the system configurations and balance requirements. Systems with more units do not necessarily provide a higher reliability since they are more likely to fail due to imbalance. Thus, optimal system design is key to maximize the system reliability which is investigated through numerical examples in this dissertation.
First of all, I would like to thank my advisor, Dr. Elsayed for his guidance, inspiration, encouragement and support during my Ph.D. study at Rutgers University. His pursuit of excellence, hard work and positivity have had a profound influence on both my career and personal life. I also want to thank the committee members of my dissertation, Dr. Pham, Dr. Xi and Dr. Burlion, for their comments and advice. Their suggestions and questions provided me with valuable insights to accomplish this dissertation. Furthermore, I want to express my gratitude to the faculty and staff of the Department of Industrial and System Engineering for their assistance.

I also would like to thank my husband Tao Ning and my children Daniel and Hebe. It is their accompany that makes this journey adventurous and unforgettable. I also want to thank my parents, my grandparents and parents-in-law for their support on all my decisions.
DEDICATION

To my beloved family and friends.
# TABLE OF CONTENTS

ABSTRACT OF THE DISSERTATION ........................................................................................................... ii

ACKNOWLEDGEMENT ........................................................................................................................... v

DEDICATION ........................................................................................................................................ vi

TABLE OF CONTENTS ........................................................................................................................... vii

LIST OF ILLUSTRATIONS ....................................................................................................................... x

LIST OF TABLES ......................................................................................................................................... xv

CHAPTER 1 INTRODUCTION ................................................................................................................... 1

1.1 Motivation of the Research .................................................................................................................. 1

1.2 Problem Definition and Assumptions .................................................................................................. 3

1.2.1 Multi-dimensional System Description ........................................................................................... 3

1.2.2 Assumptions .................................................................................................................................... 6

1.3 Reliability Estimation of \((k_1, k_2)\)-out-of-\((n, m)\) Pairs: G Balanced Systems with Units Performing Same Function .............................................................................................................................. 7

1.4 Reliability Estimation of \((k_1, k_2)\)-out-of-\((n, m)\) Pairs: G Balanced Systems with Units Performing Complementary Functions ................................................................................................................................. 8

1.5 Reliability Estimation of Spherical \(k-n-i\): G Balanced Systems ....................................................... 9

1.6 Degradation Modeling of \((k_1, k_2)\)-out-of-\((n, m)\) Pairs: G Balanced Systems .......................... 10

1.7 Organization of the Dissertation ........................................................................................................ 11

CHAPTER 2 LITERATURE REVIEW ......................................................................................................... 12

2.1 Multi-dimensional \(k\)-out-of-\(n\): F Systems ....................................................................................... 12

2.1.1 Multi-dimensional Consecutive \(k\)-out-of-\(n\): F Systems .............................................................. 12

2.1.2 Other Multi-dimensional \(k\)-out-of-\(n\): F Systems ........................................................................ 15

2.2 Consecutive \(k\)-out-of-\(n\) Systems ...................................................................................................... 15
2.3 Weighted $k$-out-of-$n$ Systems ................................................................. 17
2.4 $k$-out-of-$n$ Balanced Systems .................................................................... 18
2.5 Systems with Spherically Distributed Units .................................................... 21
2.6 Controllability of UAVs .................................................................................. 23
2.7 Optimum Design of $k$-out-of-$n$ Systems ...................................................... 25
2.8 Degradation Modeling of $k$-out-of-$n$ Balanced Systems ................................. 26
   2.8.1 Degradation Modeling of Individual Units .................................................. 26
   2.8.2 Degradation Modeling of Systems with Multiple Units ............................. 28
2.9 Summary .......................................................................................................... 29

CHAPTER 3 RELIABILITY ESTIMATION OF $(k_1,k_2)$-OUT-OF-$(n,m)$ PAIRS: G

BALANCED SYSTEMS WITH UNITS PERFORMING SAME FUNCTION ......................... 31

3.1 Problem Definition and Assumptions ................................................................ 31
   3.1.1 System Description .................................................................................... 32
   3.1.2 Balance Requirement ................................................................................ 33

3.2 Reliability Estimation of $(k_1,k_2)$-out-of-$(n,m)$ Pairs: G Balanced Systems Considering Forced-down Units as Failed ................................................................. 37
   3.2.1 Reliability Estimation of $(k_1,k_2)$-out-of-$(n,m)$ Pairs: G Balanced Systems: $k$-out-of-$n$ ........................................................................................................... 37
   3.2.2 Reliability Estimation of $(k_1,k_2)$-out-of-$(n,m)$ Pairs: G Balanced Systems: Transition Enumeration .................................................................................................................. 39
   3.2.3 Reliability Estimation of $(k_1,k_2)$-out-of-$(n,m)$ Pairs: G Balanced Systems: Algorithm ............................................................................................................................... 45

3.3 Reliability Estimation of $(k_1,k_2)$-out-of-$(n,m)$ Pairs: G Balanced Systems Considering Forced-down Units as Standbys ......................................................... 47
3.3.1 Reliability Estimation of \((k_1, k_2)\)-out-of\((n, m)\) Pairs: G Balanced Systems: Markov Chain ................................................................. 49

3.3.2 Reliability Estimation of \((k_1, k_2)\)-out-of\((n, m)\) Pairs: G Balanced Systems: Transition Enumeration.......................................................... 55

3.3.3 Reliability Estimation of \((k_1, k_2)\)-out-of\((n, m)\) Pairs: G Balanced Systems: Algorithm................................................................. 57

3.4 Numerical Examples ............................................................................................................ 60

3.4.1 Numerical Examples of \((k_1, k_2)\)-out-of\((n, m)\) Pairs: G Balanced Systems Considering Forced-down Units as Failed ................................................................. 60

3.4.2 Numerical Examples of \((k_1, k_2)\)-out-of\((n, m)\) Pairs: G Balanced Systems Considering Forced-down Units as Standbys................................................................. 65

3.5 Conclusions.......................................................................................................................... 66

CHAPTER 4 RELIABILITY ESTIMATION OF \((k_1, k_2)\)-OUT-OF-\((n, m)\) PAIRS: G BALANCED SYSTEMS WITH UNITS PERFORMING COMPLEMENTARY FUNCTIONS 69

4.1 Problem Definition and Assumptions..................................................................................... 70

4.1.1 System Description........................................................................................................... 70

4.1.2 Balance Requirement....................................................................................................... 71

4.2 Reliability Estimation of \((k_1, k_2)\)-out-of\((n, m)\) Pairs: G Balanced Systems Considering Forced-down Units as Failed ..................................................................................... 72

4.2.1 Reliability Estimation of \((k_1, k_2)\)-out-of\((n, m)\) Pairs: G Balanced Systems: Transition Enumeration............................................................................................................. 72

4.2.2 Reliability Estimation of \((k_1, k_2)\)-out-of\((n, m)\) Pairs: G Balanced Systems: Algorithm ................................................................................................................................. 79
4.2.3 Numerical Examples ........................................................................................................ 82

4.3 Reliability Estimation of \((k_1, k_2)\)-out-of-\((n, m)\) Pairs: G Balanced Systems Considering
Forced-down Units as Standbys .................................................................................................. 90

4.3.1 Reliability Estimation of \((k_1, k_2)\)-out-of-\((n, m)\) Pairs: G Balanced Systems: Transition
Enumeration .................................................................................................................................. 90

4.3.2 Reliability Estimation of \((k_1, k_2)\)-out-of-\((n, m)\) Pairs: G Balanced Systems: Algorithm
.................................................................................................................................................. 97

4.3.3 Numerical Example ........................................................................................................... 100

4.4 Conclusions ............................................................................................................................ 101

CHAPTER 5 RELIABILITY ESTIMATION OF SPHERICALLY BALANCED SYSTEMS .................................................................................. 103

5.1 Problem Definition and Assumptions .................................................................................. 104

5.1.1 System Description .......................................................................................................... 104

5.1.2 Rotational Balance .......................................................................................................... 108

5.1.3 Symmetrical Balance ...................................................................................................... 111

5.2 Reliability Estimation of Spherical \(k-n-i\): G Balanced Systems Considering Forced-down
Units as Failed ............................................................................................................................ 118

5.2.1 Reliability Estimation of Spherical \(k-n-i\): G Balanced Systems: Transition Enumeration
.................................................................................................................................................. 118

5.2.2 Reliability Estimation of Spherical \(k-n-i\): G Balanced Systems: Algorithm ................. 124

5.3 Reliability Estimation of Spherical \(k-n-i\): G Balanced Systems Considering Forced-down
Units as Standbys ....................................................................................................................... 130

5.3.1 Reliability Estimation of Spherical \(k-n-i\): G Balanced Systems: Transition Enumeration
.................................................................................................................................................. 130

5.3.2 Reliability Estimation of Spherical \(k-n-i\): G Balanced Systems: Algorithm ................. 136

5.4 Numerical Examples ............................................................................................................. 141
LIST OF ILLUSTRATIONS

Figure 1.1 \((k_1,k_2)\)-out-of-\((n,m)\) Pairs: G Balanced System with Three Layers and Two Levels.....5

Figure 1.2 Spherical 20-16-3: G Balanced System.......................................................... 6

Figure 1.3 ‘PNPNPN’ Hexacopter .....................................................................................8

Figure 1.4 Transition Path for a \((1,1)\)-out-of-(3,2) Pairs: G Balanced System with Adjacent Units Performing Complimentary Functions ...............................................................9

Figure 2.1 Four and Six Engines Descent Systems..............................................................19

Figure 2.2 Unbalanced and Balanced Six Pair Systems.......................................................20

Figure 2.3 Spherical UAVs................................................................................................21

Figure 3.1 Two-level UAV with Three Pairs of Rotors .......................................................33

Figure 3.2 PPNNPN Hexacopter..........................................................................................34

Figure 3.3 Simplified Figure of the Two-level UAV with Three Pairs of Rotors...............35

Figure 3.4 Simplified Figure of the Two-level UAV with Three Pairs of Rotors...............36

Figure 3.5 (1,2)-out-of-(2,4) Pairs: G Balanced System..................................................38

Figure 3.6 (1,2)-out-of-(3,3) Pairs: G Balanced System....................................................40

Figure 3.7 Transition Diagram of the (1,2)-out-of-(3,3) Pairs: G Balanced System............41

Figure 3.8 Flow Chart of the Algorithm .............................................................................47

Figure 3.9 Balancing Approaches of a Three-level Pair....................................................49
Figure 3.10 Simplified Figure of the (1,2)-out-of-(2,3) Pairs: G Balanced System..........................51
Figure 3.11 Markov Chain of the (1,2)-out-of-(2,3) Pairs: G Balanced System..........................52
Figure 3.12 Transition Diagram of (1,1)-out-of-(2,2) Pairs: G Balanced System..........................57
Figure 3.13 Reliability of the (1,2)-out-of-(2,4) Pairs: G Balanced System.................................61
Figure 3.14 Reliability for Systems with Different $k_1, k_2, n, m$ ...........................................62
Figure 3.15 Reliability for Systems with Fixed $k_1 \times k_2$ and $n \times m$ .....................................63
Figure 3.16 Reliability of (1,2)-out-of-(3,3) Pairs: G Balanced System.................................65
Figure 3.17 Reliability of (1,2)-out-of-(2,4) Pairs: G Balanced System.................................66
Figure 4.1 Coaxial “PNPNPN” Hexacopter ..............................................................................70
Figure 4.2 Transition Diagram of (1,1)-out-of-(3,2) “PNPNPN” Hexacopter ...........................74
Figure 4.3 Simplified Transition Diagram of (1,1)-out-of-(3,2) “PNPNPN” Hexacopter ..........75
Figure 4.4 Reliability Function for Systems with Fixed Ratio $\frac{k_1}{n}$ ........................................83
Figure 4.5 Reliability Function for Systems with Fixed Ratio $\frac{k_2}{m}$ ........................................84
Figure 4.6 Reliability Function for Systems with Proportional Increased $k_1, k_2, n, m$ ..............85
Figure 4.7 Reliability Function for Systems with Proportionally Increased $k_1$ and $n$ ..........86
Figure 4.8 Reliability Function for Systems with Proportionally Increased $k_2$ and $m$ ..........87
Figure 4.9 Reliability Function for Systems with Fixed $k_1 \times k_2$, $n \times m$ and $k_1$, $n$ k_2$, $m$ ..........................88

Figure 4.10 Reliability Function for Systems with Fixed $k_1 \times k_2$, $n \times m$ ..........................89

Figure 4.11 Pair Resumption of the (2,1)-out-of-(4,2) Pairs: G Balanced System ..........................91

Figure 4.12 Transition Diagram of the (1,1)-out-of-(3,2) Pairs: G Balanced System ..........................94

Figure 4.13 Reliability of the (1,2)-out-of-(2,4) Pairs: G Balanced System ..........................101

Figure 5.1 Spherical Unmanned Vehicles ..............................................................................103

Figure 5.2 Spherically Balanced Systems with $i$ Planes ..........................................................105

Figure 5.3 Six $dof$ of a Spherically Balanced System ................................................................107

Figure 5.4 Propulsion of Spherically Balanced Systems with Three and Six Planes ..........................109

Figure 5.5 Torque Calculation ...............................................................................................110

Figure 5.6 Symmetrical Balance ..........................................................................................113

Figure 5.7 Spherically Balanced Systems with Six Planes .........................................................114

Figure 5.8 Unbalanced State for a Spherical 30-8-6: G Balanced System .................................115

Figure 5.9 State Transition for a Spherical 30-8-6: G Balanced System ....................................120

Figure 5.10 State Transitions for a Spherical 30-8-6: G Balanced System .................................121

Figure 5.11 Spherical 30-8-6: G Balanced System ....................................................................125

Figure 5.12 State Transitions of a Spherical 10-8-6: G Balanced System .................................131
Figure 5.13 State Transitions of a Spherical 30-8-9: G Balanced System .................................................................133

Figure 5.14 Reliability Estimation of Spherical k-n-i: G Balanced Systems .................................................................142

Figure 5.15 Reliability Estimation of Spherical k-n-i: G Balanced Systems .................................................................143

Figure 5.16 Reliability Estimation of Spherical k-n-i: G Balanced Systems with Fixed k ..................145

Figure 5.17 Reliability Estimation of Spherical k-n-i: G Balanced Systems with Fixed k ..................146

Figure 5.18 Reliability of Spherical 30-8-6: G Balanced Systems .............................................................................148

Figure 5.19 A Transition of a Spherical 30-8-9: G Balanced System .................................................................149

Figure 5.20 Reliability of Spherical 30-8-9: G Balanced Systems .............................................................................150

Figure 5.21 Reliability Comparison with Different k .............................................................................151

Figure 5.22 Reliability Comparison with Different i .............................................................................153

Figure 5.23 Reliability Comparison with Different n .............................................................................154

Figure 6.1 Degradation Process of a Pair in (2, 2)-out-of-(4, 3) Pairs: G Balanced System ....................................................................................................................................................171

Figure 6.2 Reliability of (2, 2)-out-of-(3, 3) Pairs: G Balanced System with Different Scale Parameter ....................................................................................................................................................181

Figure 6.3 Reliability of (2, 2)-out-of-(3, 3) Pairs: G Balanced System with Different Mean ........182

Figure 6.4 Reliability of (3,3)-out-of-(4,4) Pairs: G Balanced System with Different Shape Parameter ....................................................................................................................................................183
Figure 6.5 Reliability of \((k_1,k_2)\)-out-of-\((n,m)\) Pairs: G Balanced System When Units Follow Inverse Gaussian Process………………………………………………………………………………..184

Figure 6.6 Reliability of \((k_1,k_2)\)-out-of-\((n,m)\) Pairs: G Balanced System When Units Follow Improved Inverse Gaussian Process……………………………………………………………………..184

Figure 6.7 Battery Capacity Degradation Process……………………………………………………………………………………………………………………………………………………………………..186

Figure 6.8 Simulated Battery Capacity Degradation Process………………………………………………………………………………………………………………………………………………………………………………………………………………..187

Figure 6.9 Reliability of (2,2)-out-of-(4,4) and (3,3)-out-of-(4,4) Pairs: G Balanced Systems…..188

Figure 6.10 Reliability of (2,2)-out-of-(4,4) Pairs: G Balanced System………………………………………………………………………………………………………………………………………………………………………………………………..189

Figure 6.11 Reliability of (3,2)-out-of-(4,4) Pairs: G Balanced Systems…………………………………………………………………………………………………………………………………………………………………………………………………..190
LIST OF TABLES

Table 5.1 Comparison Between $k$-out-of-$n$ and Spherical $k$-$n$-$i$: G Balanced Systems...............145

Table 5.2 Comparison Between $k$-out-of-$n$ and Spherical $k$-$n$-$i$: G Balanced Systems...............146

Table 5.3 Comparison of Number of States with Resuming Activities as a Function of $k$..............152

Table 5.4 Comparison of Number of States with Resuming Activities as a Function of $n$ and $i$...154

Table 6.1 Battery Capacity Degradation vs. Number of Cycle..................................................186

Table 6.2 Estimated Parameters.................................................................................................186
CHAPTER 1

INTRODUCTION

1.1 Motivation of the Research

Systems with units arranged in a multi-dimensional spatial configuration are commonly used in many applications. For example, Unmanned Aerial Vehicles (UAV) with multi-level rotary wings have been used in military and defense for a long time and have recently been expanded for commercial use in package delivery, aerial photography and others. One of the special multi-dimensional spatial configurations is the spherical Unmanned Underwater Vehicle (UUV) which has multiple thrusters located on the spherical surface to provide balanced thrust during nuclear storage ponds patrol and mine exploration (Fernandez et al., 2018). Another application is the Spherical Phased Array Antenna (SPAA) which transmits imagery data at high bit rates from the satellite to the ground station through the radiating elements mounted on the spherical surface (Kumar et al., 2018). Systems with multi-dimensional distributed units are redundant systems and hence require that at least \( k \) out of \( n \) units to be operating for the system to perform its function properly. Moreover, depending on different configurations, these systems may also require operating units to form certain pattern in order to maintain a balanced thrust or steady data transmission.

In addition to the \( k \) out of \( n \) requirement, multi-dimensional systems have different balance requirements depending on the units’ configuration. For example, for the micro UUV, if the thrusts of a paired thrusters are different, the UUV will travel in a curved trajectory
(Watson et al., 2011). For the SPAA, the hemispherical surface is divided into \( l \) sectors. It is essential for the satellites to transmit data with sufficient and effective isotropic radiated power (EIRP) which implies that a certain number of active radiating elements in each sector needs to work properly to provide the required directivity and power (Kumar et al., 2013). The reliability of these systems depends not only on the reliability of individual units but also their configurations and the sequence and locations of failed units. In this dissertation, we impose balance as an additional requirement to address the system failure. Such systems are referred to as multi-dimensional \( k\)-out-of-\( n \): G/F Balanced system (G: Good, F: Failed).

Many researchers have studied the reliability of systems with multi-dimensional distributed units. In a linear/circular connected \((r,s)\)-out-of-\((m,n)\): F lattice system, when all the units in an \((r,s)\)-submatrix fail, the system is considered failed (Yamamoto and Miyakawa, 1995; Yamamoto and Miyakawa, 1996; Yamamoto and Miyakawa, 1997; Yamamoto and Akiba, 2005). In a more complicated three-dimensional lattice system, if three units on one triangle fail, the system is considered failed (Akiba et al., 2004). This research is expanded to \( d \)-dimensional systems (Godbole et al., 1998).

System balance was initially considered in a two-dimensional \( k\)-out-of-\( n \) pairs: G balanced system. The system includes \( 2n \) units evenly distributed on a circle and remains balanced when operating units are symmetric with respect to one pair of perpendicular axes (Hua and Elsayed, 2015, Hua et al., 2015). Balanced engine system in planetary descent vehicles has two (or three) engine pairs formed along the diameters on a circle. For a balanced
Reliability estimation for such systems is challenging. The problem becomes more challenging when units are in a multi-dimensional arrangement.

The failure of multi-dimensional balanced systems such as UAVs with multi-level rotary wings and SPAA with radiating elements mounted on the spherical surface may result in major consequences especially when operating in extremely dangerous environments and highly populated areas. However, research that addresses the reliability estimation for such systems is sparse. Therefore, this dissertation investigates the reliability estimation of a multi-dimensional: G/F balanced system with units arranged circularly in a multi-level configuration or arranged spherically under different scenarios.

1.2 Problem Definition and Assumptions

A multi-dimensional balanced system with units distributed circularly or spherically may have different configurations depending on its applications. Among those, we focus on two systems: 1) balanced systems with units distributed circularly in multi-level and 2) balanced systems with units located on a spherical surface. In both cases we investigate approaches for its reliability estimation under different operating scenarios while maintaining the balance of the system.

1.2.1 Multi-dimensional System Description

1.2.1.1 Multi-level Balanced System Description

We generalize the balanced systems with multi-level of units distributed circularly as \((k_1,k_2)\)-out-of-\((n,m)\) pairs: G balanced systems. In a \((k_1,k_2)\)-out-of-\((n,m)\) pairs: G
balanced system, we consider \( n \) pairs of units evenly distributed on a circular plane. Each pair includes \( 2m \) units with \( m \) units arranged vertically on each side of the pair. These vertically stacked \( m \) units on each side of a pair are the \( m \) levels of the system. There is a total of \( 2 \times n \times m \) units for the entire system.

Figure 1.1 shows a 3-pair-2-level system. Each circle represents a unit. The numbers in the circles are the indices of the pairs. For each pair to be operating properly, at least \( k_2 \) out of \( m \) units are required to work on both sides of the pair. At lease \( k_1 \) out of \( n \) pairs should work properly for the system to provide its desired function. For the multi-dimensional systems with circular arrangement in this dissertation, assuming that the weights of all units are identical, each operating unit weights 1 and failed unit weights 0. System is balanced when the gravity center is in the center of the circle. The units in this system can perform the same function or different functions, as in a PNPNPN (‘P’ denotes that a rotor rotates clockwise and ‘N’ denotes that a rotor rotates anticlockwise) hexacopter where the adjacent rotors rotate in different directions.
1.2.1.2 Spherically Balanced System Description

In a spherically balanced system, units are located on $i$ planes that are orthogonal to the sphere’s horizontal plane and pass through the center of the sphere. These $i$ planes intersect with each other at the sphere’s top and bottom points. It is assumed that there are $n$ units evenly distributed on the circumference of each plane. For the system to be operating properly, at least $k$ units are required to work. The operating units can be located on any position of any plane. This system is generalized as a spherical balanced $k$-$n$-$i$: G balanced system. Figure 1.2 shows a spherical 20-16-3: G balanced system. The three planes are in red, blue and green respectively. On each plane, there are 16 units (indexed by numbers inside the small circles). The arrows next to the circles represent the propulsion directions for each unit. A total of 20 units are required to work for the system to be working. The balance of the spherical system is defined in two aspects: 1) the system is rotationally balanced meaning that the torques (calculated based on the units’ propulsion vectors and
position vectors) along the yaw, roll and pitch axes are zero; 2) the system is symmetrical balanced meaning that the operating units in the system are symmetric with respect to three vertical planes that pass through the center of the sphere.

![Figure 1.2 Spherical 20-16-3: G Balanced System](image)

1.2.2 Assumptions

Throughout this dissertation, the following assumptions are held unless stated otherwise:

- The probability of two or more simultaneous failures is negligible.
- The lifetimes of units in a system are *i.i.d.*
- The lifetimes of standby units (standby occurs when we force down operating units to achieve system’s balance) do not change during standby period.
1.3 Reliability Estimation of \((k_1, k_2)\)-out-of-\((n, m)\) Pairs: G Balanced Systems with Units Performing Same Function

In Chapter 3, we investigate the reliability estimation for \((k_1, k_2)\)-out-of-\((n, m)\) pairs: G balanced systems when all of the units perform the same function. As discussed in section 1.2.1, there are \(2 \times n \times m\) units in total. A balanced system can provide its desired function when at least \(k_1\) pairs are operating properly. Within a good pair, at least \(k_2\) units are working on both sides of the pair. To maintain the balance of this \(n\)-pair-\(m\)-level system, when one unit fails, we either force down one of the opposite units in the same pair or resume one of the standbys on the same side of the pair, if there are any. The system fails when: 1) less than \(k_1\) pairs of units are operating or 2) no units can be forced down/resumed in case of imbalance. The reliability estimation for such systems presents two challenges.

First, because of the balancing requirement and multi-dimensional arrangement, it is challenging to find the cut set (a set that causes the system’s failure) for the \((k_1, k_2)\)-out-of-\((n, m)\) pairs: G balanced system. This becomes even more complicated when at least certain number of operating units/pairs are required to work. In Chapter 3, we obtain the system reliability by enumerating the transitions among system’s operational states and calculating the probability for these states.

Second, the probability calculation for states requires multiple integrations. In an operational state’s probability expression, the order of integration is the same as the number
of transitions from the initial state (no failures) to this state. In some cases, closed-form expressions can not be obtained.

1.4 Reliability Estimation of \((k_1,k_2)\)-out-of-\((n,m)\) Pairs: G Balanced Systems with Units Performing Complementary Functions

One practical example of the \((k_1,k_2)\)-out-of-\((n,m)\) pairs: G balanced system with adjacent units performing complementary functions is an ‘PNPNPN’ hexacopter shown in Figure 1.3. ‘P’ denotes that a rotor rotates clockwise and is represented in white circles. ‘N’ denotes that a rotor rotates anticlockwise and is represented by grey circles.

![Figure 1.3 ‘PNPNPN’ Hexacopter](image)

In addition to forcing down or resuming units to maintain the system balance, for this system, pairs also are forced down to ensure that any two adjacent pairs perform complementary functions. Figure 1.4 shows a possible transition path for a \((1, 1)\)-out-of-\((3, 2)\) pairs: G balanced system. In this system, at least one out of three pairs is required to be working. For each pair, at least one of two units on both sides are required to work. The working/failed/forced-down units are represented as white/black/grey circles.
is the initial state with no failures. In Figure 1.4-b, when a unit fails, one of the opposite units is forced down to maintain the system balance. In Figure 1.4-c, when another unit fails, the pair is considered failed. However, the remaining two pairs are rotating in the same direction. Thus, one of the two pairs are forced down as shown in Figure 1.4-d.

Figure 1.4 Transition Path for a $(1, 1)$-out-of-(3, 2) Pairs: G Balanced System with Adjacent Units Performing Complementary Functions

In addition to the challenges presented in section 1.3, when adjacent units are performing complementary functions, the forcing down and resumption of pairs brings a significant complexity in the reliability estimation. This is discussed in Chapter 4.

1.5 Reliability Estimation of Spherical $k$-$n$-$i$: G Balanced Systems

In Chapter 5, we present the reliability estimation of spherical $k$-$n$-$i$: G balanced systems. As introduced in section 1.2.1.2, the system has $i$ vertical planes. There are $n$ units evenly distributed on each plane. At least $k$ units are required to work for the system to function properly. The reliability estimation for such system presents two challenges. First, different from the $n$-pair-$m$-level system in which the locations of forcing-down and resuming units
are straightforward, when imbalance occurs in the spherically balanced system, the best forcing-down/resuming choice is not obvious. Because there are two balance requirements: rotational balance and symmetrical balance. When the failed units match a certain pattern, some rules are given to rebalance the system. If not, one needs to search for the best forcing-down/resuming choices among all of the possible forcing-down and resuming choices. The second challenge is the mathematical representation of symmetrical balance. As a multi-dimensional system, the symmetrical balance of the spherical $k$-$n$-$i$: $G$ balanced system is not readily obvious. It is important to have a mathematical approach in the determination of symmetrical balance.

1.6 Degradation Modeling of $(k_1, k_2)$-out-of-$(n, m)$ Pairs: $G$ Balanced Systems

In many situation, degradation processes of systems are closely monitored by sensors and/or cameras. Thus, it is of great importance to develop degradation models for systems of interest to further investigate system reliability estimation. In Chapter 6, the degradation models of $(k_1, k_2)$-out-of-$(n, m)$ pairs: $G$ balanced systems are investigated when degradation data are available. Besides the challenges presented in reliability estimation of the two types of $(k_1, k_2)$-out-of-$(n, m)$ pairs: $G$ balanced systems, degradation modeling of such systems requires the consideration of randomness in units’ degradation. Operational conditions, environmental factors and manufacturing variations are the main reasons for units’ randomness.
Second, it has been shown that existed degradation models are mostly stationary and not dependent on degradation status. However, in real life, units’ degradation increments often depend on its immediately preceding degradation values. For example, when the crack length is large, the future crack growths are also mostly large. It is important to develop stochastic process to model this dependency.

Third, in a balanced system, when the degradation values among the units are not uniform, operating units are forced down to maintain the system balance. It is of great interest to develop a mechanism to balance the degradation values among units to improve the system reliability,

1.7 Organization of the Dissertation

This dissertation is organized as follows: In Chapter 2, we present a comprehensive literature review for the related research topics. In Chapter 3, we present the reliability estimation of \((k_1, k_2)-\text{out-of-}(n, m)\) pairs: G balanced system when all the units perform the same function. In Chapter 4, we present the reliability estimation of \((k_1, k_2)-\text{out-of-}(n, m)\) pairs: G balanced system when any adjacent pairs are performing complementary functions. In Chapter 5, we present the reliability estimation for spherical \(k-n-i: G\) balanced system. In Chapter 6, degradation models for \((k_1, k_2)-\text{out-of-}(n, m)\) pairs: G balanced systems are investigated. Chapter 7 presents the conclusions of this dissertation and future research topics.
In this chapter, we review the related work for systems with spatially distributed units such as $k$-out-of-$n$ systems, consecutive $k$-out-of-$n$ systems, multi-dimensional $k$-out-of-$n$ systems, weighted $k$-out-of-$n$ systems, balanced $k$-out-of-$n$ systems as well as spherical systems. One category of balanced systems with multi-dimensional spatially distributed units is Unmanned Aerial Vehicles (UAV with at least one level of rotary wings and spherical UAV with rotors arranged on a sphere). For UAVs, it is necessary that the rotors provide the desired thrust while maintaining the system balance. Literature related to the controllability of UAV are thoroughly reviewed to better understand balanced systems. Degradation of units has a major effect on the system’s performance. Thus, the degradation models including stochastic processes for single units and complex systems are also reviewed.

2.1 Multi-dimensional $k$-out-of-$n$: F Systems

2.1.1 Multi-dimensional Consecutive $k$-out-of-$n$: F Systems

In a $k$-out-of-$n$: F system, the system fails when any $k$ out of $n$ units fail and its reliability can be easily obtained (Elsayed 2012). However, in a multi-dimensional consecutive $k$-out-of-$n$: F system, the system’s status depends not only on the total number of failed units, but also on the locations of failed units. The configurations of such systems range from two-dimensional triangles, circles, to three-dimensional cubes and spheres.
Different configurations introduce different requirements depending on the system’s desired function.

Salvia and Lasher (1990) introduce a two-dimensional consecutive \(k\)-out-of-\(n\) system. In this system, units are allocated on each intersection point of a square grid of size \(n\) and the system fails when a square grid of size \(k\) fails. Zuo (1993) expands the work by proposing a rectangular and cylindrical two-dimensional consecutive \(k\)-out-of-\(n\) system. Bounds of the failure probabilities for such systems are obtained. Chang and Huang (2010) evaluate the reliability of the two-dimensional consecutive \(k\)-out-of-\(n\) system based on Markov Chain. Systems with more complicated configurations are explored by other researchers. For example, Akiba and Yamamoto (2001) introduce the two-dimensional \(k\)-within-connected-\((r, s)\)-out-of-\((m, n)\): F system. This system fails when \(k\) units in an \(r \times s\)-submatrix fail. A recursive algorithm is developed to estimate the reliability for this system. Yamamoto and Akiba (2005) later develop a new algorithm for reliability estimation of a circularly connected-\((r, s)\)-out-of-\((m, n)\): F lattice system. In this system, \(m \times n\) units are arranged in a cylindrical grid that contains \(m\) circles and each circle has \(n\) units. The system fails when all units in a grid of size \(r \times s\) fail.

Three-dimensional consecutive \(k\)-out-of-\(n\): F systems are also widely investigated. Boushaba and Ghoraf (2002) propose a three-dimensional system consisting of a cubic grid of size \(n\) and when all units in a cubic grid of size \(k\) fail, the system fails. They obtain the upper and lower reliability bounds for the system. Mahmoudboushaba and Zinebazouz (2011) introduce an intermediate system to estimate the lower bound of the
three-dimensional systems. Gharib et al. (2010) project the three-dimensional system to two and one dimensional system as special cases and study the reliability of the three-dimensional $k$-within-$(2, 2, 2)$-out-of-$(m, 2, 2)$: F system where $(i, j, k)$ represents the number of units in each side of a cuboid arrangement. In this system, the system fails when $k$ or more units fail in any $(2, 2, 2)$ cuboids. Gharid et al. (2011) later extend the three-dimensional cubic grid system to a $(1, 1, 2)$ or $(1, 2, 1)$ or $(2, 1, 1)$ -out-of-$(n, 2, 2)$: F lattice system. This system is arranged as a cuboid of $n$ layers. Each layer consists of $2 \times 2$ units. The system fails when at least two connected units fail. Akiba et al. (2004) introduce a more complicated three-dimensional lattice system that fails when three units on at least one triangular fail. One possible generalization for multi-dimensional consecutive $k$-out-of-$n$ system is that for a system represented by $n_1 \times n_2 \times \ldots \times n_m$ nodes, the system fails when a contiguous $r_1 \times r_2 \times \ldots \times r_m$ nodes fail. Cowell (2015) derives a closed reliability expression for this system.

In addition to the system configurations presented above, there are studies that explore realistic situations. For example, Habib et al. (2010) consider the system fails when the total number of failed units exceeds a specified maximum number or when a consecutive-$(r, s)$-out-of-$(m, n)$ fails. This model is an extension of Yuge et al. (2003) in which a two-dimensional consecutive $k$-out-of-$n$: F system is investigated under the condition of restricted number of failed units.
2.1.2 Other Multi-dimensional $k$-out-of-$n$: F Systems

In addition to the multi-dimensional consecutive $k$-out-of-$n$: F systems, other systems with similar requirements are also studied. In these systems, consecutive failures do not necessarily cause the system failure. For example, a $k$-within-linear/circular-connected-$(r, s)$-out-of-$(m, n)$: F system is studied by Makri and Psillakis (1997). In this system, when $k$ or more units fail in an $r \times s$-submatrix out of $(m, n)$ matrix, the entire system is considered failed. Boehme et al. (1992) generalize the systems to a connected-$X$-out-of-$(m, n)$: F lattice system. The reliability expressions for simple systems are provided. Godbole et al. (1998) extend the work into $d$-dimensional systems, which makes reliability estimation more challenging and only the reliability upper bound is obtained for these systems.

2.2 Consecutive $k$-out-of-$n$ Systems

Consecutive $k$-out-of-$n$ systems are widely used in power systems, pipeline systems, signal submission and many other systems. For example, an oil pipeline series system fails when $k$ or more consecutive failed pump stations out of total $n$ stations fail (Cowell, 2015). Similarly, the success of signal transmission from one location to another using microwave requires at least $k$ consecutive stations out of $n$ to be successful. The vacuum system in an electron accelerator includes many identical vacuum bulbs. To assure the successful traveling of the beam, at least $k$ adjacent bulbs in the vacuum system need to be operating. In quality control of certain products, $n$ cameras are installed adjacent to each other. The quality is assessed by having at least $k$ consecutive cameras working properly.
The consecutive $k$-out-of-$n$ systems are usually categorized based on the units’ configuration and failure/good condition (F system or G system). Indeed, $k$-out-of-$n$: F system and $k$-out-of-$n$: G system are dual problem and the reliability of one is the unreliability of the other. Therefore, in this dissertation, we mainly focus on the consecutive $k$-out-of-$n$: F systems. In such systems, units are either arranged linearly or circularly. Regardless of the configurations, the consecutive $k$-out-of-$n$ systems are studied when: 1) system fails when $k$ consecutive units fail; 2) system fails when $r$ units fail in $k$ consecutive units; 3) system is tested for more than one time and fails when there are at least $m$ tests having non-overlapping $k$ consecutive failed units (Eryilmaz and Mahmoud, 2012).

Approaches for estimating the reliability for such systems include combinatorial analysis (Bollinger and Salvia, 1982; Bollinger, 1982; Hwang, 1986; Papastavridis, 1990; Sfakianakis et al., 1992), Markov chain imbedding technique (Fu, 1986; Fu and Hu, 1987; Lam and Zhang, 2000; Papastavridis and Lambiris, 1987), recursive algorithms (Chiang and Niu, 1981; Derman et al., 1982; Hwang, 1982; Shanthikumar, 1982). Chang et al. (2000) and Eryilmaz (2010) review the reliability of consecutive $k$-out-of-$n$ system and the related systems. In this dissertation, we mainly focus on the new models and methods developed after 2010. Eryilmaz and Kan (2010) introduce a consecutive $k$-within-$m$-out-of-$n$: F system with exchangeable dependent units. The system is considered as a generalization of consecutive $k$-out-of-$n$ system as well as $k$-out-of-$n$ system. If at least $k$ within $m$ units fail in $n$ linearly ordered arrangements, the system is considered failed. Eryilmaz (2012) expands the work when units are nonidentical. The reliability expression
is obtained when $2m > n$. Similar systems are also explored (Eryilmaz et al., 2009; Eryilmaz, 2011).

Combinatorial methods and cut enumeration are applied to estimate the reliability of consecutive $k$-out-of-$n$ system (Forghani-elahabad and Bonani, 2017; Mohammadi et al., 2017a; Mohammadi et al., 2017b; Sáenz-de-Cabezón and Wynn, 2011). Mohammadi et al. (2017a) introduce an efficient multi-cut enumeration method for both $k$-out-of-$n$ systems and consecutive $k$-out-of-$n$ systems. In their study, it is not assumed that the probability of two simultaneous failures is zero.

### 2.3 Weighted $k$-out-of-$n$ Systems

A weighted $k$-out-of-$n$ system consists of $n$ units; each unit has a positive integer weight $\omega_i$, the system is operational when the summation of $\omega_i$ is at least $k$. The main difference between weighted $k$-out-of-$n$ systems and $k$-out-of-$n$ systems is that in the weighted $k$-out-of-$n$ systems, $k$ might be larger than $n$ (the total number of units). This is because that the weight for each unit is larger or equal to 1.

Wu and Chen (1994) propose a weighted $c$-out-of-$n$: G system and its dual system. In this study, recursive algorithms are introduced to estimate the system reliability. Chang et al. (1998) investigate a circular consecutive weighted $c$-out-of-$n$: F system and develop an algorithm to estimate its reliability. Chen and Yang (2005) expand the one-stage weighted $k$-out-of-$n$ system to a two-stage weighted $k$-out-of-$n$ system. The first stage of the proposed system is the $m$ subsystem with certain coherent structure. Each subsystem
is a weighted $k$-out-of-$n$ system and is considered as the second stage of the system. Reliability bounds for $s$-dependent units are obtained. Eryilmaz and Tutuncu (2009) study the reliability of a linear consecutive weighted $k$-out-of-$n$ system. The units in the system are non-identical and independent. Eryilmaz (2013) studies a system consisting of $n$ units each with random weight represented by integer numbers. The system is operating when the total weights of all the units is larger or equal to $c$. Eryilmaz et al. (2009) and Eryilmaz (2011) study the dynamic change of the number of working units in an operating weighted $k$-out-of-$n$ system.

Weighted $k$-out-of-$n$ systems with multi-state units are also studied in Li and Zuo (2008). Recursive algorithm and universal generating function (UGF) are applied to obtain the reliability for such systems. Song et al. (2017) propose a stochastic multiple-valued approach (SMVA) to evaluate the system’s reliability. It is shown that the proposed SMVA is more efficient than UGF. Similar studies are explored by other researchers (Gao et al., 2018; Faghih-Roohi et al., 2014; Lin et al., 2017).

### 2.4 $k$-out-of-$n$ Balanced Systems

Sarper (2005) discusses the reliability estimation of Descent Systems of Planetary Vehicles for the future crewed missions to Mars. This vehicle has four or six engines allocated evenly on a circle. Each engine along with its opposite engine form a pair that passes through the center of the circle, as shown in Figure 2.1. To maintain the balance, when one engine fails, the opposite engine in the same pair is forced down. He considers each pair as a correlated two-unit subsystem and uses bivariate exponential distribution to
model the entire system. However, this approach is only applicable when the number of engines (and pairs) is small.

![Figure 2.1 Four and Six Engines Descent Systems](image)

Hua and Elsayed (2016) define balance of the system differently for Unmanned Ariel Vehicle (UAV). In their system, multiple pairs of rotors (engines) are evenly distributed on a circle. The system is balanced when there exists at least one pair of perpendicular axes of symmetry. They introduce the concept of Moment Difference (MD) to determine the degree of symmetry with respect to any candidate axes that can be either along a pair or in the middle of two adjacent pairs. After examining the MD for every potential pair of perpendicular axes, if there are any pairs that have zero MD values for both axes, the system is considered balanced. As shown in Figure 2.2, white circles and black circles represent operating and failed rotors respectively. They are located evenly on a circle.

Figure 2.2-a is a balanced six pair system since the system is symmetric with respect to a pair of perpendicular axes (blue lines). They also define this model as $k$-out-of-$n$ pairs: G balanced system. In this system, at least $k$-out-of-$n$ pairs of the rotors need to operate properly for the UAV to provide its function while maintaining balance. Not only reliability estimation and approximation method are given, degradation analysis and load
sharing effect are also considered (Hua and Elsayed, 2015; Hua and Elsayed, 2016; Hua et al., 2015).

![Figure 2.2 Unbalanced and Balanced Six Pair Systems](image)

More recently, reliability estimation for balanced systems has attracted many researchers’ attention. Cui et al. (2018) introduce a \( k \)-out-of-\( n \): \( F \) balanced system with \( m \) sectors. Each sector includes \( n \) units and fails when \( k \) out of \( n \) units are failed or forced down. The balance is defined as each sector having same number of operating units. Fang and Cui (2020) further consider the balanced system with \( m \) sectors as a cold standby system. In addition to failures due to degradation, they assume that units can also fail during start-up process. Once the unit starts up, it is subjected to degradation. Cui et al. (2019) later assume that each unit (subsystem) follows an irreducible finite state homogeneous continuous-time Markov process and obtain the reliability when the system is repairable. Wu et al. (2020) introduce a multi-state balanced system under multiple criteria. In their
study, the system is unbalanced when its sojourn time in an unbalanced state exceeds certain threshold.

2.5 Systems with Spherically Distributed Units

Systems with spherically distributed units are widely used in a diverse range of industries. The most common examples are spherical autonomous systems including Unmanned Aerial Vehicles (UAVs), Unmanned Underwater Vehicles (UUVs), Unmanned Ground Vehicles (UGVs) and Spherical Robots. As shown in Figure 2.3-a, spherical Micro UUVs can be used to patrol nuclear storage ponds to detect undesirable leaks (Watson and Green, 2010; Watson and Green, 2014). Figure 2.3-b is a spherical UUV ODIN-III that monitors the underwater environment (Choi et al., 2003).

![Figure 2.3 Spherical UAVs](image)

(a) Micro-Autonomous Underwater Vehicle (Watson and Green, 2014)  
(b) Omni-Directional Intelligent Navigator (Choi et al., 2003)
The reliability of such systems depends not only on the reliability of individual units but also their configuration and system balance requirements. For instance, for a Micro-Autonomous Underwater Vehicle (μAUV) with six propeller-based thrusters arranged around the equator of the sphere, it is noticed that if a single horizontal thruster is working, it will cause the vehicle to rotate. And if the thrust outputs of the paired thrusters are different, the μAUV will travel in a curved trajectory (Watson et al., 2011). The balance requirements of spherical systems have a profound effect on the reliability estimation. In this example, it is easy to define the balance requirements of the system since the μAUV is ballasted, which means it cannot roll or pitch. Since all the thrusters are mounted on the same plane, one only needs to analyze the thrusters’ relative spatial relationship on a 2-dimensional horizontal plane to determine the balance of the μAUV. However, if the vehicle is not ballasted or thrusters are not arranged on the same plane, the spherical system becomes more difficult to analyze.

Generally, spherical systems have six degrees of freedom (dof): surge, sway, heave, roll, pitch and yaw. For simplicity, existing spherical systems only utilize part of these dof during actual movements (Yue et al., 2013). Roll and pitch are usually not considered which makes it easier to keep the spherical system balanced. However, it is possible that the thrusters fail to provide a balanced propulsive force when thrusters in certain locations fail. At the same time, redundant thrusters are introduced to the system to improve the reliability of the whole system. More often, at least $k$ out of $n$ thrusters needs to operate to provide the required propulsive force for the system. There are situations when more tasks are required by spherical autonomous systems and more advanced
spherically balanced systems are developed in the future. At that time, balance analysis of the spherical system becomes more complicated and important.

Spherical robot with internal mechanisms is also one of the most popular spherical systems. The difference between the Spherical Unmanned Vehicles and spherical robot with internal mechanisms is the configuration of the rotors. The rotors of Spherical Unmanned Vehicles are usually mounted around the sphere. However, the rotors of spherical robot are usually located inside the sphere to provide the propulsion for the sphere to roll. For spherical robots with internal mechanisms, the main purpose is to control the movement of spherical robot over a plane. According to Borisov et al. (2012), there are mainly three different types of control mechanism for spherical robots: 1) By changing the center of mass using various internal actuators; 2) By changing the angular momentum of the system using internal rotors; 3) By shape deformation. The second mechanism is the most common and has the widest applications. The mechanisms for different spherical robots can be found in (Borisov et al., 2012; Chaplygin, 2002; Gajbhiye and Banavar, 2015; Javadi A and Mojabi, 2004; Kayacan et al., 2011; Kilin, 2001; Javadi and Mojabi, 2004).

2.6 Controllability of UAVs

As discussed in sections 2.4 and 2.5, some of the systems with spatially distributed units require balanced propulsions. As the application of UAV expands from military and aerospace to photo taking as well as package delivery, the reliability study for such systems is crucial since the failure of UAVs may cause loss of human lives and property
damage. Many different design and control methods are investigated to make the UAVs more robust in case of rotor failures. Control allocation is one of the most common methods applied when one or more of rotors fail in an UAV. This may be achieved by varying the thrust of rotors to provide enough force and control the vehicle’s position. For example, Achtelik et al. (2012b) study the control reallocation when one rotor fails for quadrotor, hexagon-shaped hexacopter and the triangle-shape two-level helicopters. Freddi et al. (2011) investigate the controlling strategy for a quadcopter when one rotor fails. It is shown that by giving up the controlling of yaw direction, the remaining rotors can maintain a horizontal spin. Mueller and D’Andrea (2014) introduce the fault-tolerant control design for a quadcopter when one, two opposing or three propellers fail. The strategy is to have the quadcopter rotate about an axis while maintaining its flying height. Marks et al. (2012) propose a control allocation scheme for an eight-rotor UAV. According to the proposed reallocation scheme, the UAV can maintain stability when up to four rotors fail. When a partial failure occurs, a fault tolerant controlling method is necessary. Alwi and Edwards (2013) introduce a Linear Parameter Varying based scheme that combines sliding mode methods and control allocation. The scheme is able to redistribute the control signals when there are partial failures. Du et al. (2015) analyze the controllability and fault tolerant control for hexacopters. A fault tolerant control strategy is introduced to control a degraded system when the maximum lift of each rotor is larger than a certain value.

In addition to the control allocation method, new configurations are designed recently. Giribet et al. (2017) design a hexagon-shaped hexacopter where its rotors can be titled to
reject disturbance torques when a failure occurs. The condition for this approach is that
the electronic speed controllers (ESCs) can stop and start motor in opposite directions
using bidirectional rotors and bidirectional ESCs.

2.7 Optimum Design of $k$-out-of-$n$ Systems

Systems with spatially distributed units are usually redundant which means that the
system consists of more units than it “really” needs. The benefit of the redundancy is to
maintain a high system reliability. However, to achieve the highest reliability, the
optimum allocation of the units becomes necessary. Thus, the optimum design of $k$-out-
of-$n$ systems is studied by many researchers. In this dissertation, we mainly focus on the
optimum design for non-repairable $k$-out-of-$n$ systems.

Ben-Dov (1980) studies the reliability optimization of $k$-out-of-$n$ systems when the units
are $i.i.d.$ and the total number of units is fixed. Zuo (1993) investigates the optimal design
for a two-dimensional consecutive $k$-out-of-$n$: G system. The study considers both
reliability and cost of failure during optimization. Kuo and Zuo (2003) provide a
thorough review of reliability optimization for $k$-out-of-$n$ systems. Cai et al. (2016)
investigate the optimization of linear consecutive $k$-out-of-$n$ system using Birnbaum
importance based genetic algorithm. Amrutkar and Kamalja (2017) study the importance
measures for a weighted consecutive $k$-out-of-$n$ system and develop an efficient
algorithm to obtain the reliability. Rahmani et al. (2016) investigate the importance
measures for a weighted $k$-out-of-$n$ system when units have random weights. In general,
the reliability optimization for multi-dimensional $k$-out-of-$n$ systems is rarely investigated.
2.8 Degradation modeling of $k$-out-of-$n$ Balanced Systems

2.8.1 Degradation Modeling of Individual Units

In traditional reliability analysis, units’ failure times are used in model selection and parameter estimation. However, degradation modeling is more appropriate when the units exhibit degradation and when there are limited testing samples. There are two degradation modeling approaches: physics-based degradation model and stochastic-process-based degradation model. In this dissertation, we mainly focus on the stochastic-process-based degradation models.

Meeker and Escobar (1998) present a general degradation model for any individual unit $i$ at time $t_j$: $y_{ij} = Y_{ij} + \varepsilon_{ij}$. In this model, $Y_{ij}$ and $\varepsilon_{ij}$ are the actual degradation path and error (residual deviation) respectively. The forms of $Y_{ij}$ and $\varepsilon_{ij}$ determine the types of the degradation process. For example, when $\varepsilon_{ij}$ follows a normal distribution with zero mean and $Y_{ij}$ is linear, the degradation path $y_{ij}$ is a Wiener process. Stochastic processes are preferred when the prior knowledge is not available to provide a better explanation for the physics-based degradation model. Three classes of the stochastic degradation process have been developed to account for the inherent random degradation increments over time, they are: Wiener process, Gamma process and the inverse Gaussian process. The general degradation path for an individual unit is modeled as a stochastic process as shown in (Meeker and Hamada, 1995; Lu and Meeker, 1993; Lu et al., 1997; Lu et al., 1996).
In the Wiener process, the degradation increments are assumed to be normally distributed with mean and variance as a function of time. Observable factors such as temperature and humidity as well as unobservable factors such as sensors’ capability are incorporated in the model (Doksum and Normand, 1995; Wang, 2010; Ye et al., 2013). The first hitting time (failure time) can be obtained in a closed form (Molini et al., 2011; Si et al., 2013; Wang et al., 2014a). One disadvantage for Wiener process is that the degradation path is not necessarily monotone which limits its application in the monotonically increasing degradation modeling such as crack growth. Gamma process and the inverse Gaussian process, on the other hand, are applied when degradation path is monotonically increasing.

Gamma process is widely used in modeling the damage of materials (Sun et al., 2018; Zhang and Xiao, 2018; Park and Padgett, 2005). In a Gamma degradation process, increments are independent and assumed to follow a gamma distribution. The observable and unobservable random factors are considered by incorporating them in the shape function (Lawless and Crowder, 2004). Based on different forms of shape function, Gamma process is classified into homogeneous Gamma process and non-homogeneous Gamma process. An improved Gamma process is developed to model the non-linear degradation growth (Fan et al., 2015).

In an inverse Gaussian degradation process, the degradation increments are assumed to be independent and non-negative following an inverse Gaussian distribution. Inverse Gaussian process is applied in laser device degradation modeling where Expectation-
Maximization (EM) algorithm is used to estimate the parameters (Wang and Xu, 2010). More accurate parameter estimation is obtained by updating the degradation data continuously (Pan et al., 2016). Ye et al. (2014), Ye and Chen (2014) discuss accelerated degradation testing (ADT) planning with the inverse Gaussian process. An improved inverse Gaussian is proposed as a complement to the original inverse Gaussian model (Guo et al. 2018). In this model, the degradation increment at time $t_j$ depends on the degradation value at time $t_{j-1}$ which implies that the degradation growth is non-linear.

### 2.8.2 Degradation Modeling of Systems with Multiple Units

Degradation analysis is critical in reliability assessment of systems with multiple units, especially when the units are interdependent. A few attempts have been made to expand the degradation model from one-dimension to multi-dimension. Based on the current literatures, researchers approach multi-dimensional degradation modeling by 1) developing multivariate stochastic processes. Satish (1985) develops a bivariate Wiener process model and obtains the hitting time for the model. Because of its flexibility, Li et al. (2020) and Sadoughi et al. (2018) discuss the implementation of multivariate Gaussian process in reliability assessment for systems with dependent units. 2) applying a copula function to construct dependency among multiple degradation processes. Peng et al. (2016) develop a multivariate inverse Gaussian process based on a multivariate copula function. Pan and Balakrishnan (2011) and Pan et al. (2013) introduce a bivariate Wiener process and multivariate gamma process using copula functions. Wang et al. (2014b) construct a bivariate Wiener process with time-scale transformation based on Frank copula function. The copula approach is inapplicable when there exit many degradation
processes. Hua and Elsayed (2015) propose degradation models (both physics and statistics) for balanced systems. They investigate the effect of units’ spatial distribution on the overall system reliability. Cui et al. (2019) discuss the reliability estimation for balanced system based on Markov processes. Later, Fang and Cui (2021) develop a degradation model based on an aggregated stochastic process for a multi-state balanced system with two subsystems. They assume the system does not fail immediately when loses its balance. Instead, when the system returns to balanced state within a time threshold, the system is considered operative.

2.9 Summary

In this chapter, we provide the state-of-the-art literature related to reliability estimation of balanced system with spatially distributed units. In sections 2.1-2.3, the k-out-of-n systems including multi-dimensional consecutive k-out-of-n systems with spatial configurations, consecutive k-out-of-n systems and weighted k-out-of-n systems are reviewed. The reliability estimation of these systems is obtained for simple configurations and reliability bounds are given for complex systems. These systems are applied to model: one-dimensional systems with linearly/circularly ordered units such as oil pipeline systems, signal transmission systems and electron accelerator; two-dimensional systems with units arranged on intersections of grid such as energy grid systems; three-dimensional systems with units allocated on cube or cuboid. The existing literature neither addresses systems with units arranged circularly in a multi-level configuration nor systems with units arranged on a spherical surface.
Sections 2.4-2.6 present the literature related to balanced systems such as Mars Module, UAV and spherical systems and the controllability of UAV. Different from other multi-dimensional systems, Mars Module, UAV and spherical systems requires balance for systems’ proper operation. In sections 2.4 and 2.5, balance requirements for these systems are discussed. The existing dynamic controllability of UAV is achieved by varying the thrust of rotors or tilting the rotors to provide enough force and to control the system’s position when the system loses balance due to a rotor failure. Reliability estimate of UAVs with balance requirements have not been investigated.

Section 2.7 introduces the literature review of the optimum design of multi-dimensional k-out-of-n systems. It is observed that research related to the reliability estimation and optimal configurations for multi-dimensional k-out-of-n systems is sparse. Section 2.8 reviews the work related to degradation modeling for both single units and complex systems with interdependent units. Expanding the one-dimensional degradation models into multi-dimensional degradation models is challenging. Closed form expressions for reliability are not available for most of multivariate stochastic processes. System balance is not considered in current multivariate degradation models, hence the motivation of this study.
CHAPTER 3

RELIABILITY ESTIMATION OF \((k_1,k_2)\)-OUT-OF-\((n,m)\) PAIRS: G

BALANCED SYSTEMS WITH UNITS PERFORMING SAME FUNCTION

Chapter 2 provides a review of different \(k\)-out-of-\(n\) systems and highlights systems with spatially distributed units. Unmanned Aerial Vehicles (UAV) with rotors in circular arrangement represent a special class of such systems. However, the research on the balanced UAVs is limited. Moreover, new designs of UAVs with different configurations are currently in use. For example, UAVs with more than one level of rotors (coaxial UAVs) are widely used in many applications. However, the reliability estimation of such systems has not been investigated. In this chapter, we develop reliability estimation models for balanced systems with units arranged circularly in multi-level. Two scenarios are considered: 1) forced-down units are considered failed; 2) forced-down units are considered as standbys and can be resumed when needed. For the first scenario, the reliability estimation of the system is investigated when the units in the system are identical or non-identical. For the second scenario, the reliability estimation of the system is studied assuming different failure time distributions. We also develop an algorithm to estimate reliability of such systems when the number of units is large.

3.1 Problem Definition and Assumptions

The exponential proliferation of spatial distributed systems such as multi-dimensional Unmanned Aerial Vehicle (UAV) and their wide use in commercial applications have heightened the need for thorough analysis of their reliability. In this chapter, we introduce
reliability estimation models for one of these spatial systems, namely: 
\((k_1, k_2)\)-out-of-\((n, m)\) pairs: G balanced system. The balance of such systems is defined and discussed in detail in section 3.1.2.

3.1.1 System Description

Considering a rotary winged UAV with more than one level of rotors (multi-dimensional), several pairs of rotors are evenly arranged on a circle on each level. The axis of each pair passes through the center of the circle and the angles between any adjacent pairs are equal. There is more than one rotor arranged vertically in a multi-level configuration on each side of pairs. Figure 3.1 shows a 3-pair-2-level UAV (xFoldRig.com). We generalize this UAV to be \((k_1, k_2)\)-out-of-\((n, m)\) pairs: G balanced system. In this system, \(n\) pairs of units are evenly distributed on a circular arrangement in ‘\(m\)’ levels. On each side of a pair, there are \(m\) units arranged vertically in multi-level and in parallel with the opposite side of the pair. There are \(2m\) units for each pair and a total of \(2nm\) units for the entire system. For each pair to be functioning, at least \(k_1\) out of \(m\) units need to operate properly on both sides of the pair. Otherwise, the pair is considered failed and the remaining functioning units of the same pair are forced down. The system is good when at least \(k_1\) out of the \(n\) pairs work properly. For example, the UAV in Figure 3.1 is a \((2, 1)\)-out-of-(3, 2) system. It implies that at least 1 out of 2 rotors on both sides of a pair need to work properly to ensure the function of the pair and 2 or more pairs out of 3 pairs should be working for the UAV to be considered functioning properly.
3.1.2 Balance Requirement

As discussed earlier, some spatially distributed systems require balance in order to perform its function and unbalanced system is considered failed. For example, to improve the reliability, the UAV designers use dynamic controllability methods such as control allocation to balance the system when one or more rotors fail as discussed in Achtelik et al. (2012). In this Chapter, the concept of the operational balance is introduced as an alternative to the controllability methods. In this $n$-pair-$m$-level system, assuming that all units are identical, each operating unit is assigned 1 and failed unit is assigned 0. The system is balanced when the center of gravity of the units is in the center of the circular arrangement.

Based on the controllability testing method introduced by Du et al. (2015), for a PPNNPN hexacopter shown in Figure 3.2, the system is controllable when it is balanced and 2-out-of-3 pairs of rotors are operating properly (“P” denotes that a rotor rotates clockwise and is represented using white circles. “N” denotes that a rotor rotates counter-clockwise and...
is represented by grey circles). We demonstrate that the concept of balance and the $k$-out-of-$n$ requirement combined can be regarded as an alternative for maintaining the controllability (balance) of UAVs.

![Figure 3.2 PPNNPN Hexacopter](image)

Figure 3.2 PPNNPN Hexacopter

Figure 3.3 shows a simplified 3-pair-2-level UAV. White circles represent operating rotors. Black and grey circles represent failed and forced down rotors respectively. The locations of rotors are indexed by numbers. The black dot is the center of gravity for the system. As shown, there are three pairs of rotors in total where two rotors are vertically arranged in parallel in each side of a pair. In Figure 3.3-a, all the rotors are operating, and the center of gravity is in the center of the system. In Figure 3.3-b, the bottom rotor on the right side of the third pair fails. The center of gravity is then shifted along the third pair and the system is thus unbalanced.

Since an unbalanced system fails whenever the failure occurs, we force down other rotors for the system to maintain balance while satisfying the requirement of the minimum
working rotors. There might be more than one way to force down units to maintain the system balance. In this example, Figures 3.3-c and 3.3-d are two ways to rebalance the system when the rotor in the third pair fails. In Figure 3.3-c, one of the opposite rotors in the third pair is forced down. In Figure 3.3-d, one of the rotors in the first and the second pair are forced down. Both systems in Figures 3.3-c and 3.3-d are balanced. It is obvious that the number of forced-down units in 3.3-c is less than those in 3.3-d. Thus, forcing down units in 3.3-c is preferred. The system is always balanced when the numbers of working rotors on both sides of each pair are equal. For simplicity, when one rotor fails, one of the opposite rotors in the same pair is forced down to maintain the system balance. In a system with \( m \) identical rotors assigned to each side of a pair, the choices of forcing down rotors are random. For systems with non-identical rotors, the rotor with lowest reliability is forced down first.

**Figure 3.3** Simplified Figure of the Two-level UAV with Three Pairs of Rotors
Forcing down rotors is not the only choice to maintain system balance when failure occurs, in some cases, resuming the operation of previously forced down rotors may result in re-balancing the system. This choice is only applicable when the forced down rotors are considered as standbys and can resume operations when needed. In Figure 3.4-b, when the bottom rotor on the left side of the third pair fails, the system is unbalanced due to this failure, and two options are considered: 1) forcing down the only surviving rotor of the third pair, as shown in Figure 3.4-c or 2) resuming the forced down rotor in the same side of the third pair, as shown in Figure 3.4-d. Obviously, the second option is preferred. In this chapter, both of the two scenarios are considered in the reliability estimation of the system.

**Figure 3.4** Simplified Figure of the Two-level UAV with Three Pairs of Rotors
3.2 Reliability Estimation of \((k_1, k_2)\)-out-of-\((n, m)\) Pairs: G Balanced Systems
Considering Forced-down Units as Failed

3.2.1 Reliability Estimation of \((k_1, k_2)\)-out-of-\((n, m)\) Pairs: G Balanced Systems: \(k\)-out-of-\(n\)

Suppose that all the units in the system are identical, then the system is modeled as a double \(k\)-out-of-\(n\) system. The first \(k\)-out-of-\(n\) requires at least \(k_1\)-out-of-\(n\) pairs to be operating properly for the system to be functional. Each operating pair is a sub-\(k\)-out-of-\(n\) system which requires at least \(k_2\)-out-of-\(m\) units on both sides of a pair to be operating properly for the pair to be functioning.

Figure 3.5 shows an \((1, 2)\)-out-of-(2, 4) system. In this system, two pairs of units (the first pair is in blue color and the second pair is in orange color) are evenly distributed on a circle and at least one of the two pairs need to be operational for the system to function properly. Each pair has four units \((m = 4)\) allocated on each side of the pair and two units need to be functioning properly for the pair to be operational. System balance requires that if one unit fails, one of the operational units on the opposite side of the same pair is forced down immediately. Since the failure of one unit causes the forcing down of another unit in the same pair, these two units can be considered as a series system. In other words, each pair has \(m\) series systems of two units and \(k_2\)-out-of-\(m\) series systems need to be operating for the pair to function properly. In this 2-pair-4-level system, each pair is a sub-\(k\)-out-of-\(n\) system which requires two out of four series systems to be operational.
Suppose that the probability density function (pdf) and the cumulative distribution function (CDF) of a single unit are \( f \) and \( F \) respectively. The pdf \( f_z(t) \) and CDF \( F_z(t) \) of a two-unit series system are as shown in Equation (3.1) and (3.2):

\[
f_z(t) = 2 \times (1 - F(t)) \times f(t) \tag{3.1}
\]

\[
F_z(t) = 1 - (1 - F(t))^2 \tag{3.2}
\]

The probability that a pair (a sub-\( k \)-out-of-\( n \) system) is operational at time \( t \) is obtained in Equation (3.3):

\[
G(t) = \sum_{l=k}^{m} \binom{m}{l} (1 - F_z(t))^l (F_z(t))^{m-l} \tag{3.3}
\]

The reliability of the system at time \( t \) is shown in Equation (3.4):
\[ R_s(t) = \sum_{j=h}^{n} \binom{n}{j} G(t)^j (1 - G(t))^{n-j} \] (3.4)

3.2.2 Reliability Estimation of \((k_1, k_2)\)-out-of-\((n, m)\) Pairs: G Balanced Systems:

**Transition Enumeration**

When units in the system are non-identical, the \(k\)-out-of-\(n\) method introduced in 3.2.1 is not applicable. In this section, we estimate the reliability of \((k_1, k_2)\)-out-of-\((n, m)\) pairs: G balanced systems when the units are non-identical. The operational states are defined as the states that the system is operational and balanced. The reliability of the system at time \(t\) is the summation of all the probabilities that the system is in one of the operational states at time \(t\). To obtain the reliability of the system, we need to obtain all the operational states and the probabilities that the system is in these states. This is achieved by enumerating the transitions from one operational state to other possible operational states. Each transition between states is caused by one unit’s failure. The probability that the system is in a specific operational state at time \(t\) is the probability that the system transits from the initial state (no failure) to this operational state during \((0, t)\).

The reliability of \((1, 2)\)-out-of-\((3, 3)\) pairs: G balanced system as shown Figure 3.6 is studied as an example to illustrate the transition enumeration. In this system, three pairs of rotors are evenly distributed on a circular plane with three rotors allocated vertically on each side of a pair. Thus, it is a 3-pair-3-level system.
Figure 3.6 (1, 2)-out-of-(3, 3) Pairs: G Balanced System

Figure 3.7 shows the transitions between the operational states of this system. In the transition diagrams, vectors \((l_1, l_2, l_3)\) represent the states of the system where \(l_i\) is the number of failed units in the \(i^{th}\) pair. When one unit fails, one of the opposite units in the same pair is forced down immediately. Thus \(l_i\) also represents the number of forced-down units in the \(i^{th}\) pair. Since at least \(k_2\)-out-of-\(m\) units are required to function on both sides of a pair, when the number of failed or forced down rotors in \(i^{th}\) pair exceeds \(m-k_2\): 
\[
l_i = m-k_2 + 1, \quad \text{the remaining } k_2 - 1 \text{ operating units on both sides are forced down and this pair is considered failed. In this system, 2-out-of-3 units are required to work on each side of a pair, any states with more than 2 failed or forced down rotors such as (3, 0, 0) or (3, 1, 1) are nonexistent and state (2, 2, 2) is the failure state. All the transitions from the operational states eventually lead to this failure state. When the number of failed or forced-down units on each side of a pair reaches 2, the remaining units are forced down and the pair is considered failed. For example, the first pair in state (2, 0, 0) has two failed or forced-down units. This pair is considered failed and no further failures could occur in this
pair. The transitions from this state to other states occur when new failures occur in the second or the third pair. These transitions lead to states (2, 0, 1) and (2, 1, 0).

Figure 3.7 Transition Diagram of the (1, 2)-out-of-(3, 3) Pairs: G Balanced System

After obtaining all the operational states based on the transition diagram, the probability that the system is in operational state \((i, j, l)\) at time \(t\) is denoted as \(P_{(i,j,l)}(t)\) and is a product of two parts: the number of possible transition paths \((\eta)\) from state \((0, 0, 0)\) to state \((i, j, l)\) and the probability of transiting from state \((0, 0, 0)\) to state \((i, j, l)\). There can be more than one transition path that lead the system to state \((i, j, l)\). For example, as shown in Figure 3.7, there are two different transition paths from the initial state to state \((0, 1, 1)\):
\((0,0,0) \rightarrow (0,1,0) \rightarrow (0,1,1)\) and \((0,0,0) \rightarrow (0,0,1) \rightarrow (0,1,1)\). The numbers on each arrow are the numbers of possible transitions from one operational state to another. There are three possible transitions between states \((0, 0, 0)\) and \((0, 1, 0)\), as well as between states \((0, 1, 0)\) and \((0, 1, 1)\). The total number of possible transition paths for transition \((0,0,0) \rightarrow (0,1,0) \rightarrow (0,1,1)\) is obtained: \(3 \times 3 = 9\) Similarly, for transition \((0,0,0) \rightarrow (0,0,1) \rightarrow (0,1,1)\), there are also nine possible transition paths. The total number of transition paths from the initial state to state \((0, 1, 1)\) is obtained: \(\eta = 9 + 9 = 18\).

The probability of transiting from state \((0, 0, 0)\) to state \((i, j, l)\) is obtained by iterative calculation. It is determined by its preceding states (the last states before \((i, j, l)\)) and the failure position during the last transition. For simplicity, suppose that the units from the same pair are identical and follow the same failure time distribution. However, units from different pairs are non-identical and the failure distribution are different. The failure time pdf and CDF of a single unit for the \(i^{th}\) pair are represented as \(f^i(t)\) and \(F^i(t)\), \(i = 1, \ldots, n\).

The pdf \(f_2^i(t)\), the CDF \(F_2^i(t)\) and \(R_2^i(t)\) are obtained respectively in Equations (3.5)-(3.7):

\[
f_2^i(t) = 2 \left(1 - F^i(t)\right) f^i(t)
\]

\[
F_2^i(t) = 1 - \left(1 - F^i(t)\right)^2
\]

\[
R_2^i(t) = \left(1 - F^i(t)\right)^2
\]
Suppose that state \((i, j, l)\) has one preceding state and during the previous transition, one unit fails in pair \(a\). The number \(u\) is the number of forced-down units during this transition.

The \(P_{(i,j,l)}(t)\) is obtained in Equation (3.8):

\[
P_{(i,j,l)}(t) = \eta \int_{\tau=0}^{t} P_{\text{prec}}(\tau) \times f^a_2(\tau) \times \left( R(\tau)_2^a \right)^{-x} \left( R(\tau)_2^b \right)^{-y} \ldots \left( R(\tau)_2^f \right)^{-w} \ d\tau
\]

\[
\times \left( R(t)_2^a \right)^{-\frac{x}{2}} \left( R(\tau)_2^b \right)^{-y} \ldots \left( R(\tau)_2^f \right)^{-w}
\]

where \(x, y, \ldots, w\) are the exponents of \(R(t)_2^a, R(t)_2^b, \ldots, R(t)_2^f\) in the probability expression of preceding state: \(P_{\text{prec}}(t)\). When state \((i, j, l)\) has more than one preceding state, the probability that the system is in state \((i, j, l)\) at time \(t\) is the summation of probabilities calculated based on each of the preceding states.

Based on Equation 3.8, the probability that the \((1, 2)\)-out-of-\((3, 3)\) pairs: \(G\) system is in states \((1, 0, 0)\) at time \(t\) is obtained in Equations (3.9):

\[
P_{(1,0,0)}(t) = m \times \int_{\tau=0}^{t} f^1_2(\tau_1) d\tau_1 \times R^1_2(t)^{m-1} \times R^2_2(t)^m \times R^3_2(t)^m
\]

\[
= m \times F^1_2(\tau_1) \times R^1_2(t)^{m-1} \times R^2_2(t)^m \times R^3_2(t)^m
\]

\[
= 3 \times F^1_2(\tau_1) \times R^1_2(t)^2 \times R^2_2(t)^3 \times R^3_2(t)^3
\]

Similarly, \(P_{(0,1,1)}(t)\) is obtained in Equation (3.10):
\[ P_{(0,1)}(t) = m^2 \int_{\tau_2=0}^{\tau_2} \int_{\tau_1=0}^{\tau_1} f_2^2(\tau_1) f_2^3(\tau_2) d\tau_1 d\tau_2 \times R_2^1(t) \times R_2^2(t) \times R_2^3(t) \]

\[ + m^2 \int_{\tau_2=0}^{\tau_2} \int_{\tau_1=0}^{\tau_1} f_2^3(\tau_1) f_2^2(\tau_2) d\tau_1 d\tau_2 \times R_2^1(t) \times R_2^2(t) \times R_2^3(t) \]

\[ = m^2 \int_{\tau_2=0}^{\tau_2} F_2^2(\tau_2) f_2^3(\tau_2) d\tau_2 \times R_2^1(t) \times R_2^2(t) \times R_2^3(t) \]

\[ + m^2 \int_{\tau_2=0}^{\tau_2} F_2^3(\tau_2) f_2^2(\tau_2) d\tau_2 \times R_2^1(t) \times R_2^2(t) \times R_2^3(t) \]

\[ = 9 \times \int_{\tau_2=0}^{\tau_2} F_2^2(\tau_2) f_2^3(\tau_2) d\tau_2 \times R_2^1(t) \times R_2^2(t) \times R_2^3(t) \]

\[ + 9 \times \int_{\tau_2=0}^{\tau_2} F_2^3(\tau_2) f_2^2(\tau_2) d\tau_2 \times R_2^1(t) \times R_2^2(t) \times R_2^3(t) \]

\[ \text{(3.10)} \]

In Equations (3.11) and (3.12), the notation \( P_j^i \) denotes the \( j \)-permutations of \( i \) and is used for calculation of possible transition path.

\[ P_{(2,0,0)}(t) = P_2^m \int_{\tau_2=0}^{\tau_2} f_2(\tau_2) R_2(\tau_2) d\tau_2 \times R_2^2(t) \]

\[ = P_2^m \int_{\tau_2=0}^{\tau_2} f_2(\tau_2) R_2(\tau_2) d\tau_2 \times R_2^2(t) \]

\[ = P_2^m \int_{\tau_2=0}^{\tau_2} F_2(\tau_2) f_2(\tau_2) R_2(\tau_2) d\tau_2 \times R_2^2(t) \]

\[ = P_2^m \int_{\tau_2=0}^{\tau_2} F_2(\tau_2) f_2(\tau_2) d\tau_2 \times R_2^2(t) \]

\[ = \frac{P_2^m}{2} F_2(t) \times R_2^2(t) \]

\[ \text{(3.11)} \]

\[ P_{(0,2,1)}(t) = P_1^m \int_{\tau_3=0}^{\tau_3} f_2(\tau_3) R_2(\tau_3) d\tau_3 \times R_2^2(t) \]

\[ + P_1^m \int_{\tau_3=0}^{\tau_3} f_2(\tau_3) R_2(\tau_3) d\tau_3 \times R_2^2(t) \]

\[ = P_1^m P_2^m \int_{\tau_3=0}^{\tau_3} F_2(\tau_3)^2 - F_2(\tau_3)^3 \]

\[ = \frac{P_2^m}{3} F_2(t) \times R_2^2(t) \]

\[ \text{(3.12)} \]
The reliability of the system is the summation of these probabilities.

$$R(t) = P_{(0,0)}(t) + P_{(1,0)}(t) + \ldots + P_{(1,2,2)}(t) \quad (3.13)$$

### 3.2.3 Reliability Estimation of \((k_1, k_2)\)-out-of-(\(n, m\)) Pairs: \(G\) Balanced Systems:

**Algorithm**

As discussed in section 3.2.1, estimating the reliability of \((k_1, k_2)\)-out-of-(\(n, m\)) pairs: \(G\) balanced systems is computational expensive even for systems with small \(n\) and \(m\). For systems with more than 4 pairs and 2 levels of rotors, the enumeration of the operational states becomes complicated. Moreover, the probability calculation for the operational states, i.e., each transition from one operational state to another, requires an integral calculation. Thus, as the number of transitions increases, the number of integrals in a single probability equation increases accordingly. Therefore, we develop an efficient algorithm to estimate the reliability of such systems.

The algorithm includes three parts: (1) Obtain all the operational states; (2) Obtain the number of transition paths for all the operational states; (3) Calculate the probability that the system is in certain operational states at any time. They are described below.

---

<table>
<thead>
<tr>
<th><strong>Part I: Obtain Operational States ()</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Calculate the maximum number of failed units in one pair: (l_{\text{pair, max}} = m - k_2 + 1).</td>
</tr>
<tr>
<td>Calculate the maximum number of failed pairs in the system: (l_{\text{sys, max}} = n - k_1).</td>
</tr>
<tr>
<td>Construct a vector to store all the new states and old state combination: (C).</td>
</tr>
<tr>
<td>Set an identity matrix of size (n \times n): (I_n) and a vector of ones of size (n \times 1): (\mathbf{1}_n).</td>
</tr>
<tr>
<td>Set the initial state (no failure) as a zero vector of size (1 \times n): (S_{\text{init}} = \mathbf{0}_{1 \times n}).</td>
</tr>
<tr>
<td>Place the state (S_{\text{init}}) into a list: (Q_{\text{state}}).</td>
</tr>
</tbody>
</table>

**While** \(Q_{\text{state}}\) **not empty** **do**

---
Withdraw one state from list $Q_{\text{state}}$ and set it as an old state $S_{\text{old}}$. Delete the state from $Q_{\text{state}}$

$$S_{\text{new}} = \tilde{1}_n \times S_{\text{old}} + \tilde{1}_n.$$  

for \( i = 1, i \leq n, i++ \) do

\[ h_i = 0. \]

if \( S_{\text{new}}^i (j) \geq l_{\text{pair max}} \) then

\[ h_i ++. \]

if \( h_i \leq l_{\text{sys max}} \) then

place \( S_{\text{new}}^i \) in $Q_{\text{state}}$ and store the combination of \( S_{\text{new}}^i \) and \( S_{\text{old}}^i \) in the list $C$.

end if

end if

end for

end while

**Part II: Obtain Number of Transitions ()**

Set a vector $T$ of size $r \times 1$: $T = \tilde{1}_{rx1}$ where $r$ is the total number of combinations in list $C$.

for \( i = 1, i \leq r, i++ \) do

for \( j = i+1, j \leq r, j++ \) do

if \( C_j = C_i \) then

delete $C_j$.

\[ T_i ++. \]

end if

end for

compare $S_{\text{old}}^i$ and $S_{\text{new}}^i$, find the location $d_i$ where numbers in the two vectors are different.

\[ T_i = T_i \times (m - S_{\text{old}}^i (d_i)). \]

end for

**Part III: Probability Estimation ()**

Order list $C$ based on the number of failed units in $S_{\text{new}}$ supposing there are $g$ items in $C$.

Set $p_{\text{old}} = 1$ for $S_{\text{old}}^i$ the in $C$.

for \( i = 1, i \leq g, i++ \) do

\[ p_i = T_i \int_{\tau_{i-1}}^{\tau_i} p_{\text{old}} (\tau_1) f_{\tau_{i+1}}^i (\tau_1) d\tau_1 \text{ where } p_{\text{old}} (\tau_1) \text{ is the expression of the } S_{\text{old}}^i \text{ in } C. \]

end for

for \( i = 1, i \leq g, i++ \) do

Update $p_i (t)$ by $p_i (t) \times \prod_{d_i = 1}^{a} R_{d_i}^i (t)$ where $L_{d_i} = m - S_{\text{new}} (d_i)$ and $R_{d_i}^i (t) = \left(1 - F_{d_i}^i (t)\right)^2$.

end for

Reliability of the system is $R(t) = \sum_{i=1}^{g} p_i (t)$. 
The flow chart for this algorithm is shown in Figure 3.8.

![Flow Chart of The Algorithm]

**Figure 3.8 Flow Chart of The Algorithm**

3.3 Reliability Estimation of \((k_1,k_2)\)-out-of-\((n,m)\) Pairs: G Balanced Systems

**Considering Forced-down Units as Standbys**

In section 3.2, the forced-down units are considered as failed units. However, forced-down units can be considered as standbys and can resume its operation when needed. In this section, there are two approaches to maintain the balance of the system when a failure
occurs. The first approach is to force down one of the opposite units in the same pair, which is presented in the previous section. The second approach is to resume one of the standby units on the same side of a pair. When all of the units are identical, the selection of standbys and resuming units is random. However, if units are non-identical, the unit with the highest failure rate is forced down first. Likewise, the unit with the lowest failure rate is resumed first to ensure the highest reliability.

For example, Figure 3.9 presents a pair with three levels of units. The black and grey circles are failed and forced-down units respectively. The six units are indexed from one to six. Assuming the failure rates $h_i(t)$ of the six units are in an increasing order: $h_1(t) \leq h_2(t) \leq \ldots \leq h_6(t), \forall t \geq 0$. Assuming that at time $t_1$, the fourth unit fails, the system is balanced by forcing down one of the opposite units in the same pair. Since the third unit has the highest failure rate on the left side of the pair, it is forced down at time $t_1$. Assuming that the second failure (unit one) occurs at time $t_2$, there are two approaches to balance the system: 1) resume the forced-down unit on the same side of the pair, which is the third unit; 2) force down the unit with the highest failure rate on the opposite side of the pair, which is the sixth unit. It is obvious that the first approach is preferred. In this section, estimation of system reliability is analyzed when forced-down units can be resumed when needed.
3.3.1 Reliability Estimation of \((k_1,k_2)-\text{out-of-}(n,m)\) Pairs: \(G\) Balanced Systems: 

*Markov Chain*

In this section, we study a special case of \((k_1,k_2)-\text{out-of-}(n,m)\) pairs: \(G\) balanced system when the failure times of the units follow exponential distribution. A continuous-time discrete-state stochastic process is developed to describe the state transition of the system.

The definition of the stochastic process at time \(t\) is as follows:

\[
Y(t) = s_i, \text{ when the system is in state } s_i, \ s_i \in S
\]

where \(S\) is the state space.
As discussed in section 3.2.2, vectors \((l_1, l_2, \ldots, l_n)\) include all information about the system when forced-down units are considered failed. However, when forced-down units can resume operations when needed, we require more information other than the number of failed units in each pair. In this section, we use the vector \(\left((l_1, l_2)_1, (l_1, l_2)_2, \ldots, (l_1, l_2)_n\right)\) to represent the states of system. In this vector, \(n\) pairs are listed in a decreasing order according to the total number of failed units within each pair. For example, \((l_1, l_2)_1\) has the most number of failed units where \(l_1\) and \(l_2\) are number of failed units on both sides of each pair. They are also ordered such that \(l_1 \geq l_2\) for every pair. Such ordering reduces the number of the system’s state significantly.

In this section, all units are assumed to be identical. The selection of standby as well as resumed units is random. We use \((1, 2)\)-out-of-(2, 3) pairs: G balanced system to illustrate the calculation of system reliability when the failure time distributions of all units follow an exponential distribution with parameter \(\lambda\). In this system, two pairs of units are assigned evenly in a circular arrangement and each pair has three units stacked vertically on both sides. A pair is operating properly when at least 2-out-of-3 units are working on each side of the pair. The system is good when one or both pairs are working properly while maintaining the balance. Figure 3.10 is a simplified presentation of this system.
The transition from one state to other states is described using Markov Chain as shown in Figure 3.11. State \([(0,0),(0,0)]\) is the initial state and states \([(2,0),(2,0)], [(2,1),(2,0)]\) and \([(2,1),(2,1)]\) are the failure states. We use \(s_i\) where \(i=1,2,...,15\) to represent all the states of the system:

\[
\begin{align*}
  s_1 &= ((0,0),(0,0)); \\
  s_2 &= ((1,0),(0,0)); \\
  s_3 &= ((2,0),(0,0)); \\
  s_4 &= ((1,1),(0,0)); \\
  s_5 &= ((1,0),(1,0)); \\
  s_6 &= ((2,0),(1,0)); \\
  s_7 &= ((2,1),(0,0)); \\
  s_8 &= ((1,1),(1,0)); \\
  s_9 &= ((2,0),(1,1)); \\
  s_{10} &= ((2,1),(1,0)); \\
  s_{11} &= ((1,1),(1,1)); \\
  s_{12} &= ((2,1),(1,1)); \\
  s_{13} &= ((2,0),(2,0)); \\
  s_{14} &= ((2,1),(2,0)); \\
  s_{15} &= ((2,1),(2,1))
\end{align*}
\]

The stochastic process \(Y(t)\) moves from state \(s_i\) at time \(t = 0\) to other states. Only one event can occur at any time point. The transition rates from one operational state to another are shown on the arrows in Figure 3.11. The transition rates between different states follow these rules:

Transition \([(i,j),(0,0)]\rightarrow[(i,j),(1,0)]\) with rate \(2m\lambda, ij \neq 0\)

Transition \([(i,j),(l,k)]\rightarrow[(i+1,j),(l,k)]\) with rate \((m-i)\lambda, \forall i > j\)

Transition \([(i,j),(l,k)]\rightarrow[(i,j+1),(l,k)]\) with rate \((m-j)\lambda, \forall i > j\)
Transition \( ((i,j),(l,k)) \rightarrow ((i+1,j),(l,k)) \) with rate \( 2(m-i)\lambda \), \( \forall i = j \)

Transition \( ((i,j),(i,j)) \rightarrow ((i+1,j),(i,j)) \) with rate \( n(m-i)\lambda \), \( \forall i \geq 0, \forall j \geq 0, i \neq j \)

\[
\begin{align*}
(0,0) & \rightarrow (0,0) \\
(1,0) & \rightarrow (0,0) \\
(2,0) & \rightarrow (0,0) \\
(2,1) & \rightarrow (0,0) \\
& \vdots \\
(2,2) & \rightarrow (0,0)
\end{align*}
\]

**Figure 3.11** Markov Chain of the (1, 2)-out-of-(2, 3) Pairs: G Balanced System

The probability of being in any state is obtained using the infinitesimal generator matrix:
Since states $s_{13}, s_{14}, s_{15}$ are failed states, the generator matrix is summarized as follows:

$$Q = \begin{bmatrix} -q_{11} & q_{12} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -q_{22} & q_{23} & q_{24} & q_{25} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -q_{33} & 0 & 0 & q_{36} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -q_{44} & 0 & 0 & q_{47} & q_{48} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -q_{55} & q_{56} & 0 & q_{58} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -q_{66} & 0 & 0 & q_{69} & 0 & 0 & 0 & q_{613} \\ 0 & 0 & 0 & 0 & 0 & -q_{77} & 0 & 0 & q_{710} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -q_{88} & q_{89} & q_{810} & q_{811} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -q_{99} & 0 & 0 & 0 & 0 & 0 & q_{914} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -q_{1010} & 0 & q_{1012} & 0 & q_{1014} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -q_{1111} & q_{1112} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -q_{1212} & 0 & 0 & q_{1215} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

where $Q_w$ is a $12 \times 12$ sub-matrix representing the transition from operational states to other operational states, $Q_f$ is a $12 \times 3$ sub-matrix representing transition from operational states to failed states, $\tilde{0}_1$ is a $3 \times 12$ zero sub-matrix, and $\tilde{0}_2$ is a $3 \times 3$ zero sub-matrix.

In the generator matrix:
\[ q_{11} = q_{12} = 2nm\lambda; \]
\[ q_{22} = (4m - 2)\lambda; \]
\[ q_{23} = q_{24} = (m - 1)\lambda; \]
\[ q_{25} = 2m\lambda; \]
\[ q_{33} = q_{36} = am\lambda; \]
\[ q_{44} = (4m - 2)\lambda; \]
\[ q_{47} = 2(m - 1)\lambda; \]
\[ q_{48} = 2m\lambda; \]
\[ q_{55} = 2n(m - 1)\lambda; \]
\[ q_{56} = n(m - 1)\lambda; \]
\[ q_{58} = n(m - 1)\lambda; \]
\[ q_{66} = 2(m - 1)\lambda; \]
\[ q_{69} = (m - 1)\lambda; \]
\[ q_{613} = (m - 1)\lambda; \]
\[ q_{77} = q_{710} = 2m\lambda; \]
\[ q_{88} = 4(m - 1)\lambda; \]
\[ q_{89} = (m - 1)\lambda; \]
\[ q_{810} = 2(m - 1)\lambda; \]
\[ q_{811} = (m - 1)\lambda; \]
\[ q_{99} = q_{914} = 2(m - 1)\lambda; \]
\[ q_{1010} = 2(m - 1)\lambda; \]
\[ q_{1012} = (m - 1)\lambda; \]
\[ q_{1014} = (m - 1)\lambda; \]
\[ q_{1111} = q_{1112} = n(m - 1)\lambda; \]
\[ q_{1212} = q_{1215} = 2(m - 1)\lambda; \]

Since this is a finite state continuous-time Markov Chain, the transition probability matrix is obtained by Equation (3.14):

\[ P(t) = e^{Qt}, \quad t \geq 0 \]  \hspace{1cm} (3.14)

where \( P(t) \) is the probability matrix of all the states and \( Q \) is the generator matrix.

The reliability of the system is the summation of probabilities of all the operational states and is assessed using Equation (3.15).

\[ R(t) = e^{Qt}1_w \]  \hspace{1cm} (3.15)

where \( 1_w \) is a column vector with element 1, \( 1_w = (1, \ldots, 1)^T_{1w12} \).
3.3.2 Reliability Estimation of $\{k_1,k_2\}$-out-of-$\{n,m\}$ Pairs: G Balanced Systems:

**Transition Enumeration**

The Markov Chain approach is inapplicable when not all units follow the exponential distribution. As discussed in section 3.2.2, enumerating transitions and calculating the probabilities of all operational states is an alternative approach for system reliability calculation when forced-down units are considered failed. This approach also applies when forced-down units resume operation whenever needed.

In this section, we use vectors $((l_1, l_2), (l_1, l_2), \ldots, (l_1, l_2))$ to represent the states of the system. The probability that the system is in state $((l_1, l_2), (l_1, l_2), \ldots, (l_1, l_2))$ at time $t$ is denoted by $P_{\{(l_1, l_2), (l_1, l_2), \ldots, (l_1, l_2)\}}(t)$. This probability is a product of two parts: 1) the number of transition paths ($\eta$) from initial state to the given state and 2) the probability of transiting from the initial state to the given state. The first part is obtained based on the method introduced in section 3.2.2.

The probability of transiting from initial state to given state is obtained by iterative calculation. It is determined by its preceding states’ probability expressions and the transition between the preceding states and given state. When the previous transition is to force down a unit, the probability calculation is discussed in section 3.2.2. In this section, we provide the probability calculation when the previous transition is to resume one of the previously forced-down units. For simplicity, suppose that all of the units in the system are identical and follow the same failure time distribution. The failure time $pdf$ and CDF of a
single unit are represented as \( f(t) \) and \( F(t) \). Suppose that there is only one preceding state for state \((l_1, l_2)_1, (l_1, l_2)_2, \ldots, (l_1, l_2)_n\). The probability \( P_{(l_1, l_2)_k, (l_1, l_2)_{k-1}, \ldots, (l_1, l_2)_1}(t) \) is obtained in Equation (3.16):

\[
P_{(l_1, l_2)_k, (l_1, l_2)_{k-1}, \ldots, (l_1, l_2)_1}(t) = \eta \int_{\tau_1 = 0}^{t} P_{\text{prec}}(\tau_1) f(\tau_1) R(t-\tau_1) R(\tau_1)^{-u} d\tau_1 R(t)^{u-1} \quad (3.16)
\]

where \( u \) is the number of exponents of \( R(t) \) in \( P_{\text{prec}}(t) \). When the given states have more than one preceding state, the probability that the system is in given state at time \( t \) is the summation of probabilities calculated based on each of the preceding states.

Figure 3.12 is the transition diagram for (1, 1)-out-of-(2, 2) pairs: G balanced system. There are 12 operational states in total. The reliability of the system at time \( t \) is the summation of all the probabilities that the system is in any of the operational states at time \( t \). We use state \(((1, 1),(0, 0))\) as an example to illustrate the calculation of probability of operational states. The transition path from the initial state \(((0, 0),(0, 0))\) to state \(((1, 1),(0, 0))\) is marked in blue in Figure 3.12. The probability is calculated based on Equations (3.8) and (3.16) and is obtained in Equation (3.17).

\[
P_{((1,1),(0,0))}(t) = P_{2}^{m} \int_{\tau_2 = 0}^{t} \int_{\tau_1 = 0}^{\tau_2} f(\tau_1) R(\tau_1) f(\tau_2) R(t-\tau_2) d\tau_1 d\tau_2 R_2(t)^{6} \quad (3.17)
\]

The reliability of the system is given by summing the probabilities of all operational states as shown in Equation (3.18):

\[
R(t) = P_{((0,0),(0,0))}(t) + \ldots + P_{((2,2),(1,1))}(t) \quad (3.18)
\]
Figure 3.12 Transition Diagram of the (1, 1)-out-of-(2, 2) Pairs: G Balanced System

3.3.3 Reliability Estimation of \((k_1,k_2)\)-out-of-\((n,m)\) Pairs: G Balanced Systems:

Algorithm

When forced-down units are considered as standbys, the reliability estimation for this system is more complicated than when the forced-down units are considered as failed. For example, when a failure occurs, one need to check the following conditions: 1) Does this failure cause the failure of the pair/the system? 2) Are there any units that can be resumed? 3) Are there any units that can be forced down? These conditions need to be checked every time a failure occurs. When the number of operational states of the system increases, enumerating the transition diagram becomes implausible. In this section, similar to section 3.2.3, an algorithm is introduced to calculate the reliability of the \((k_1,k_2)\)-out-of-\((n,m)\) pairs: G balanced system when forced-down units in the system can be resumed.
The algorithm includes three parts: (1) Obtain all the operational states; (2) Obtain the number of transition paths for all the operational states; (3) Calculate the probability that the system is in certain operational states at any time. They are described below.

**Part I: Obtain Operational States**

Calculate the maximum number of failed units in one pair: \( I_{\text{pair max}} = m - k_2 + 1 \).

Calculate the maximum number of failed pairs in the system: \( I_{\text{sys max}} = n - k_1 \).

Construct a vector to store all the new states and old state combination: \( C \).

Set an identity matrix of size \( 2n \times 2n \) : \( \mathbf{I}_{2n} \).

Set the initial state (no failure) as a zero vector of size \( 1 \times 2n \) : \( S_{\text{init}} = 0_{1 \times 2n} \).

Place the state \( S_{\text{init}} \) into a list: \( Q_{\text{state}} \).

**While** \( Q_{\text{state}} \) not empty do

Withdraw one state from list \( Q_{\text{state}} \) and set it as an old state \( S_{\text{old}} \). Delete the state from \( Q_{\text{state}} \).

\[
S_{\text{new}} = \mathbf{I}_{2n} \times S_{\text{old}}^T + \mathbf{I}_{2n}.
\]

for \((i=1, i \leq 2n, i++)\) do

\[
h_i = 0.
\]

\[
I_{\text{max}} = \tilde{0}_{1 \times n}.
\]

for \((j=1, j \leq 2n-1, j++, j \text{ is odd})\) do

\[
I_{\text{max}(p)} = \max \left( S_{\text{new}}^i (j), S_{\text{new}}^i (j+1) \right) \text{ where } p = \text{mod}((j-1), 2) + 1.
\]

if \( I_{\text{max}(p)} \geq I_{\text{pair max}} \) then

\[
h_i = + + .
\]

if \( h_i \leq I_{\text{sys max}} \) then

place \( S_{\text{new}}^i \) in \( Q_{\text{state}} \) and store the combination of \( S_{\text{new}}^i \) and \( S_{\text{old}}^i \) in the list \( C \).

end if

end if

end for

end for

end while

**Part II: Obtain Number of Transitions**

Set a vector \( T \) of size \( r \times 1 \): \( T = 1_{r \times 1} \) where \( r \) is the total number of combinations in list \( C \).

for \((i=1, i \leq r, i++)\) do

for \((j=i+1, j \leq r, j++)\) do

if \( C_j = C_i \) then

delete \( C_j \).

end if

end for

end for
\( T_i++ \).

end if

end for

end for

for \((i = 1, i \leq r, i++ \) \) do

\[ \text{Compare } S_i^{old} \text{ and } S_i^{new}, \text{ find the location } d_i \text{ where numbers in the two vectors are different.} \]

Set an indicator vector \( q = 0 \).

for \((j = 1, j \leq 2n-1, j++, j \text{ is odd} \) \) do

if \( S_i^{old}(j) = S_i^{old}(j+1) \neq 0 \) then

\[ q_i = q_i + 1, l = (j-1) \text{mod} 2 + 1. \]

end if

end for

\( T_i = T_i \times q_i \left( m - S_i^{old}(d_i) \right). \)

end for

\[ \text{Part III: Probability Estimation}() \]

Order list \( C \) based on the number of failed units in \( S_{new} \) supposing there are \( g \) items in \( C \).

Set \( p_{old} = 1 \) for \( S_{old}^i \) the in \( C \).

for \((i = 1, i \leq g, i++ \) \) do

if \( d_i \) is odd and \( S_{new}^i(d_i) = S_{new}^i(d_i+1) \) or \( d_i \) is even and \( S_{new}^i(d_i) = S_{new}^i(d_i-1) \) then

\[ p_i = T_i \int_{\tau_i=g}^{\tau_i} p_{old}(\tau_i) f(\tau_i) R(\tau_i) d\tau_i \]

where \( p_{old}(\tau_i) \) is the probability expression of the \( S_{old}^i \) in \( C \). \( f(t) \) and \( F(t) \) denote the pdf and CDF of a single unit.

else

\[ p_i = T_i \int_{\tau_i=g}^{\tau_i} p_{old}(\tau_i) f(\tau_i) R(t-\tau_i) d\tau_i. \]

end if

end for

for \((i = 1, i \leq g, i++ \) \) do

Update \( p_i(t) \) by \( p_i(t) \times \prod_{i=1}^{n} R_2(t)^{l_i} \) where \( l_i = m - S_{new}^i(d_i) \) and \( R_2(t) = (1-F(t))^2 \).

end for

Reliability of the system is \( R(t) = \sum_{i=1}^{g} p_i(t) \).
3.4 Numerical Examples

3.4.1 Numerical Examples of \( (k_1,k_2) \)-out-of-\( (n,m) \) Pairs: G Balanced Systems

**Considering Forced-down Units as Failed**

**Scenario 1: Reliability estimation for systems with identical units**

In this section, an example is provided to illustrate reliability estimation for the \( (k_1,k_2) \)-out-of-\( (n,m) \) pairs: G balanced system when units are identical. Suppose that the failure time of all units in a 2-pair-4-level system follow an exponential distribution with failure rate \( \lambda \). The expressions for this system are obtained in Equations (3.19) and (3.20):

\[
\begin{align*}
    f_2(t) &= 2 \times (1 - F(t)) \times f(t) = 2 \times \exp^{-3t} \times \lambda \exp^{-3t} = 2\lambda \exp^{-2t} \quad (3.19) \\
    F_2(t) &= 1 - \left(1 - F(t)\right)^2 = 1 - \left(1 - \left(1 - \exp^{-2t}\right)\right)^2 = 1 - \exp^{-2t} \quad (3.20)
\end{align*}
\]

The reliability functions are shown in Figure 3.13 assuming that the units in the system have three different failure rates: \( \frac{1}{90}, \frac{1}{60}, \frac{1}{30} \).
**Scenario 2: Reliability estimation for systems with different $k_1, k_2, n, m$**

Reliability estimations for the $(k_1,k_2)$-out-of-$(n,m)$ pairs: G balanced systems with different combinations of $k_1$, $k_2$, $n$ and $m$ are calculated when units are identical. Figure 3.14 shows that increasing $k_1$ and $k_2$, decreasing $n$ and $m$ leads to the decrease of the system reliability.
Figure 3.14 Reliability for Systems with Different $k_1, k_2, n, m$

Scenario 3: System design with fixed $k_1 \times k_2$ and $n \times m$

Suppose a system has a requirement for the minimum number of operating units (thrust forces), thus the product $k_1 \times k_2$ is fixed. Also, suppose that the number of available units is also limited, i.e. $n \times m$ is fixed. We are interested in designing a system under these assumptions that results in the arrangement with highest reliability. Figure 3.15 shows the reliability plots for systems with 16 available units and minimum number of operating units is four when all units follow exponential distribution with mean lifetime equal to 30. From the Figure 3.15, (2, 1)-out-of-(4, 2) pairs: G balanced system has the highest reliability; (1, 2)-out-of-(4, 2) pairs: G balanced system has the lowest reliability. We observe that a system tends to have higher reliability when both ratios $\frac{k_1}{n}$ and $\frac{k_2}{m}$ are close to 0.5.
**Figure 3.15** Reliability for Systems with Fixed $k_1 \times k_2$ and $n \times m$

**Scenario 4: Reliability estimation for systems with non-identical units**

In this section, an example is introduced to illustrate the reliability estimation of $(k_1, k_2)$-out-of-$(n, m)$ pairs: $G$ balanced system when the units are non-identical. For a $(1, 2)$-out-of-$(3, 3)$ pairs: $G$ balanced system, there are three pairs located evenly on a circular arrangement with three levels in each pair. As discussed in section 3.2.2, in this system, units belong to the same pair are identical have the same failure time distribution. However, units from different pairs are not identical and thus the failure times for these units follow
different distributions. Suppose the failure times of the units in the $i^{th}$ pair follow exponential with failure rate: $\lambda_i$.

Suppose $\lambda_1 = \frac{1}{30}, \lambda_2 = \frac{1}{60}, \lambda_3 = \frac{1}{90}$, reliability function for this system is shown in green color in Figure 3.16. The yellow, red and blue colors are the reliability functions for $(1, 2)$-out-of-(3, 3) pairs: G balanced system when all units are identical and follow exponential distribution with failure rates $\lambda: \frac{1}{90}, \frac{1}{60}, \frac{1}{30}$ respectively. The reliability of a system with non-identical units is higher than a system with identical units when the failure rate is the highest: $\frac{1}{30}$. However, this reliability function is lower than the other two systems when units are identical, and the failure rates are $\frac{1}{90}, \frac{1}{60}$. The reason is that the pair with the highest failure rate $\frac{1}{30}$ fails sooner and influences the reliability for the entire system.
Numerical Examples of $(k_1,k_2)$-out-of-$(n,m)$ Pairs: G Balanced Systems

Considering Forced-down Units as Standbys

In this section, an example is introduced to illustrate the reliability estimation of $(k_1,k_2)$-out-of-$(n,m)$ pairs: G balanced system when the forced-down units are considered as standbys and can be resumed back to operation when needed. For a $(1,2)$-out-of-$(2,4)$ pairs: G balanced system, when all units are identical and their failure times follow an exponential distribution with failure rate $\lambda = \frac{1}{50}$, two scenarios are considered: 1) forced-down units are considered as failed; 2) forced-down units are considered as standbys. The reliability of this system is calculated based on the algorithms introduced in section 3.2.3 and 3.3.3. From Figure 3.17, it is shown that when forced-down units can be resumed to
balance the unbalanced system; the system reliability is higher at every time point. This is obvious since whenever a new failure occurs, the choice of resumption or forcing down in the algorithm of section 3.3.3 ensures that the system maintains highest reliability.

![Figure 3.17 Reliability of the (1, 2)-out-of-(2, 4) Pairs: G Balanced System](image)

### 3.5 Conclusions

In this chapter, we explore the reliability estimation for \((k_1,k_2)\)-out-of-\((n,m)\) pairs: G balanced systems. The balance of the system is defined. A rotary-winged UAV with multi-level of rotors is studied as an example to illustrate the definition of balance. The operational balance and \(k\)-out-of-\(n\) requirements of the system are proved to be a viable alternative of dynamic controllability method for UAV. Even when the controllability
method cannot be applied when certain number of rotors fail, forcing down rotors according to the balance requirement can maintain the operation of the UAV. We consider two approaches for balancing the system when failures cause unbalance of the system, they are: (1) force down units at specific locations serving as standbys; (2) resume standby units at specific locations. The second approach is preferred compared with the first approach since it brings more functioning units to operation and therefore maintains a higher reliability for the system.

The reliability estimation of such systems is investigated under the following scenarios: (1) forced-down units are considered failed; (2) forced-down units can resume operation whenever needed. When the forced-down units are considered failed, the system is considered as a $k$-out-of-$n$ system assuming all units are identical. However, if the units are not identical, transition paths of the system from initial state to other states are enumerated. The system is operational when the system is in one of the operational states. The probability of each state is calculated based on iterative method. The reliability of the system is obtained by summing the probabilities of all the operational states.

When the forced-down units can be resumed whenever needed, a Markov Chain method is used to obtain the system reliability when the units’ failure time follows exponential distribution. When the failure time follows an arbitrarily distribution, the transition path enumeration is applied to obtain the system reliability. When the number of transition path and operational states are large, enumerating all the states is time consuming and algorithms are developed accordingly to calculate the reliability of the system. We provide
numerical examples for systems operating under different scenarios. From the example, it is observed that a system tends to have higher reliability when both ratios \( \frac{k_i}{n} \) and \( \frac{k_2}{m} \) are close to 0.5.
CHAPTER 4

RELIABILITY ESTIMATION OF

\((k_1, k_2)\)-OUT-OF-\((n, m)\) PAIRS: G BALANCED SYSTEMS

WITH UNITS PERFORMING COMPLEMENTARY FUNCTIONS

In Chapter 3, reliability estimation models are developed for a general multi-dimensional configuration with circularly arranged units. The system is modeled as the \((k_1, k_2)\)-out-of-\((n, m)\) pairs: G balanced system. It is assumed that all the units perform the same function. However, in reality, this assumption is not valid at all the time. For example, for UAVs with multiple units allocated on a circular arrangement, horizontally adjacent units can rotate in opposite directions which means that the units are performing complementary functions. Hua and Elsayed (2015) introduce the reliability estimation method for systems with one level of units allocated evenly on a circle with horizontally adjacent units performing complementary functions. In this chapter, we estimate the reliability of the \((k_1, k_2)\)-out-of-\((n, m)\) pairs: G balanced system when adjacent units are performing complementary functions. The reliability of such systems is obtained under two scenarios: 1) forced-down units are considered as failed; 2) forced-down units are considered as standbys and are resumed when needed. Analytical solutions and algorithms are investigated for both scenarios. Numerical examples are provided for both scenarios.
4.1 Problem Definition and Assumptions

4.1.1 System Description

In this chapter, we study the $n$-pair-$m$-level systems when horizontally adjacent units performing complementary functions. In such systems, $m$ units on the same side of a pair perform the same function. Horizontally operating adjacent (‘adjacent’ is used in the rest of the dissertation for brevity) units are required to perform different functions. It is not required that all the units in the same pair perform the same function. For example, Figure 4.1 is a (1, 1)-out-of-(3, 2) PNPNPN hexacopter. “P” denotes that a rotor rotates clockwise and is represented using white circles. “N” denotes that a rotor rotates counter-clockwise and is represented by grey circles. In this 3-pair-2-level system, rotors from both sides of the same pair rotate in opposite directions. However, rotors from the same side of a pair rotate in the same direction. At least 1-out-of-2 rotors needs to work on both sides of a pair for the pair to function properly. At least 1-out-of-3 pairs of rotors are required to function for the hexacopter to work.

Figure 4.1 Coaxial PNPNPN Hexacopter
4.1.2 Balance Requirement

In this \((k_1, k_2)\)-out-of-(\(n, m\)) pairs: G balanced systems, assuming that all units, regardless of their functions, follow an identical lifetime distribution. Each operating unit is assigned a weight of 1 and a failed unit is assigned a weight of 0. Similar to Chapter 3, a system is balanced when its center of gravity is in the center of the circle and unbalanced system is considered failed. Different from Chapter 3, any two operating adjacent units are required to have different rotational directions.

As discussed in Chapter 3, this definition of balance along with the \(k\)-out-of-\(n\) requirement are regarded as an alternative approach for maintaining the controllability of UAVs. In fact, to make UAVs more robust in case of rotor failures, changing rotors’ rotational direction is one of the most common methods applied to rebalance the system. Achtelik et al. (2012) suggest reversing the rotation directions of rotors in a triangle-shape two-level helicopter when one rotor fails in order to achieve controllability. Giribet et al. (2016) design a hexagon-shaped hexacopter where rotors can be tilted to reject disturbance torques when failure occurs. It is obvious that the rotors’ rotation directions are important in maintaining the controllability of UAVs. In this chapter, system balance and rotors’ rotation directions are both taken into consideration in the reliability estimation.

In this \(n\)-pair-\(m\)-level system, if one unit fails, the system fails due to the loss of balance. Thus, when one unit fails, one of the opposite units is forced down immediately to maintain the balance of the system. At the same time, when one pair of units is not operational (failed
or forced down), the adjacent pair with the most failed number of units is forced down so that any two adjacent units can have opposite rotational directions.

4.2 Reliability Estimation of \((k_1, k_2)\)-out-of-\((n, m)\) Pairs: G Balanced Systems

Considering Forced-down Units as Failed

4.2.1 Reliability Estimation of \((k_1, k_2)\)-out-of-\((n, m)\) Pairs: G Balanced Systems: Transition Enumeration

The reliability estimation of \((k_1, k_2)\)-out-of-\((n, m)\) pairs: G balanced systems (these systems function when a minimum of \(k_1\) pairs out of \(n\) pairs and \(k_2\) units out of \(m\) units on each side of the pair are functioning, otherwise the system fails) is obtained by enumerating all the transitions between operational states and calculating the probability of occurrences of these states. Operational states are those states that meet the minimum number of pairs and units in each pair while maintaining system’s balance requirements. The reliability of the system is the summation of the probabilities of being in these states.

When one unit fails, the system’s center of gravity shifts and causes immediate imbalance of the system. Thus, forcing down operating units in specific position to maintain the system balance at all the time is of interest. There are two forcing down approaches: 1) forcing down one of the opposite units in the same pair when one unit fails (to maintain system balance); 2) forcing down the rest of operating units when the number of failed units is larger than \(m - k_2 + 1\) in a pair. Considering this pair is not operational, one of the adjacent pairs that perform complementary function is forced down. In this section, all the forced-down units/pairs are considered failed, i.e. they are not resumed after forcing down.
Vectors \((l_1, l_2, ... l_n)\) are used to represent the state of the system. \(l_i\) is the number of failed units in the \(i^{th}\) pair. Since when one unit fails, a unit on the opposite side of the same pair is forced down immediately and the forced-down units cannot be resumed, the number of forced-down units in the \(i^{th}\) pair is the same as the number of failed units. The forced-down unit is randomly selected among the units in the opposite side of the same pair. Considering \(k_2\) units are required to function properly on each side of a pair, when \(l_i = m - k_2 + 1\), the remaining \(k_2 - 1\) functioning units on each side of the \(i^{th}\) pair are forced down. Then, the \(i^{th}\) pair is considered failed. At the same time, since any adjacent units are required to perform complementary functions, when one pair fails, one of the neighboring operating pairs that performs complementary functions and having the greatest number of failed units is then forced down. The forced down pair is represented by the letter “\(S\)” in state vectors.

Transition occurs when a new failure happens in the system. The transition and the new operational state are considered effective when the failed unit does not cause the failure of the entire system. If the new operational state already exists in the transition diagram, the duplicated states are deleted, and the number of transition paths increase by the number of possible transition path to the deleted new operational state.

Suppose the new operational state is \(a = (l_1, l_2, ... l_n)\), a vector \(a' = (l_1, l_2, ... l_n, l_1, l_2, ... l_n)\) is created to be compared with all the existed operational states. Suppose the existed operational state \(b = (h_1, h_2, ... h_n)\) is compared with \(a'\), if vector \(b\) can be found in vector \(a'\) as a segment, operational states \(b = (h_1, h_2, ... h_n)\) and \(a = (l_1, l_2, ... l_n)\) are considered
as the same state. The transitions from arbitrary operational state $c = (u_1, u_2, ... u_n)$ to states $b$ and $a$ are duplicated transitions. Thus, the newly created state $a = (l_1, l_2, ... l_n)$ is deleted. An example is introduced to illustrate the transitions between states. Consider the $(1, 1)$-out-of-(3, 2) PNPNPN hexacopter system mentioned in last section, the transition diagram is shown in Figure 4.2.

![Transition Diagram of the (1, 1)-out-of-(3, 2) PNPNPN Hexacopter](image)

**Figure 4.2** Transition Diagram of the (1, 1)-out-of-(3, 2) PNPNPN Hexacopter

In the transition diagram, there are 9 states in total. State $(0, 0, 0)$ is the initial state where there is no failed unit in the system. State $(2, S, 2)$ is the failure state. The subscripts $(2, 0, 0)$ and $(2, 1, 0)$ of the state $(2, S, 0)$ (in blue color) are the actual states where the number of failed units in the forced down pairs (second pair) is given. The ‘S’ in state $(2, S, 0)$ is
used to represent that the second pair in the states (2, 0, 0) and (2, 1, 0) is forced down. For these two states, since at least 1-out-of-2 unit is required for each pair to be functioning, when the second failure occurs in the first pair, this pair is considered failed. The neighbor pair with the greatest number of failed units is forced down. The forced down pair may have \( m \) failed units, \( m = 0, 1, \ldots, k_2 \) when it is forced down. We combine these states in order to reduce the total number of states. The new diagram is shown in Figure 4.3.

Figure 4.3 Simplified Transition Diagram of (1, 1)-out-of-(3, 2) PNP PNP Hexacopter

After determining all of the operational states as well as the transition diagram, the next step is to calculate the probability that the system is in these operational states. The probability that the system is in an operational state \((i, j, l)\) at time \(t\) is denoted as
The probability \( P_{(i,j,l)}(t) \) is a product of two parts: 1) the number of possible transition path from the initial state \((0, 0, 0)\) to the state \((i, j, l)\); 2) the probability of transitioning from state \((0, 0, 0)\) to state \((i, j, l)\).

The number of possible transition paths from the initial state \((0, 0, 0)\) to the state \((i, j, l)\) is the sum of the products of the numbers shown on the arrows in the transition diagram from the initial state \((0, 0, 0)\) to the state \((i, j, l)\). For example, the number of possible transition paths from the initial state \((0, 0, 0)\) to the state \((2, S, 0)\) can be calculated as: 
\[
6 \times 1 + 6 \times 4 \times 2 = 54
\]
The numbers on the arrows are the numbers of duplicated transitions.

For example, the number of possible transitions from state \((0, 0, 0)\) to state \((1, 0, 0)\) is 6. This is because the three states \((1, 0, 0)\), \((0, 1, 0)\) and \((0, 0, 1)\) represent the same state and are represented by state \((1, 0, 0)\). For each state, there are two levels of units in the failed pair which means there are two possible transitions for a pair to transit from 0 to 1. The total possible transitions from \((0, 0, 0)\) to state \((1, 0, 0)\) is \(3 \times 2 = 6\).

Closed form expressions for the probabilities \( P_{(i_l, j_2, \ldots, j_s)}(t) \) could not be obtained because the transition paths from the initial state to the state \( (l_1, l_2, \ldots, l_s) \) can be complex and random. Iterative calculations are provided instead. In this chapter, the lifetimes for units are \( i.i.d. \) random variables. The \( pdf \), CDF and reliability function for a series system with \( i \) units are \( f_i(t) \), \( F_i(t) \) and \( R_i(t) \) respectively. Since when one unit fails, one of opposite units is forced down immediately, these two units form a series system. The \( pdf \) \( f_2(t) \),
CDF $F_2(t)$ and the reliability function $R_2(t)$ for this series system are obtained in Equation (4.1)-(4.3):

\begin{align*}
f_2(t) &= 2(1-F_1(t))f_1(t) \quad (4.1) \\
F_2(t) &= 1-(1-F_1(t))^2 \quad (4.2) \\
R_2(t) &= (1-F_1(t))^2 \quad (4.3)
\end{align*}

Based on the transition diagram, the probabilities of operational states are calculated from the initial state to the last operational state. For any transition activity, there are one preceding state and one following state. The probability of the current state $(l_1, l_2, \ldots l_n)$ is based on its preceding states and the transitions between them. As discussed earlier, two transition activities are considered: 1) unit forcing down: when one unit fails, one of the opposite units is forced down; 2) pair forcing down: when one unit fails and the number of failed units in this pair is greater than $m-k_z$, then the remaining operating units of this pair are forced down and one of the adjacent pairs that performs complementary function and having the most number of failed units is forced down. The iterative probability calculations are discussed next for these two transitions.

Suppose that the probability of the system in the preceding state is: $P_{\text{prec}}(t)$. The probability that the system is in the following state at time $t$ is: $P_{\text{fol}}(t)$. If the transition activity from the preceding state to the following state is unit forcing down, then the iterative probability expression is obtained in Equation (4.4):
\[ P_{\text{f/dlt}}(t) = \eta \int_{\tau_1=0}^{t} P_{\text{prec}}(\tau_1) f_2(\tau_1) R_2(\tau_1)^{-b} d\tau_1 R_2(t)^{b-1} \quad (4.4) \]

where \( \eta \) is the number of transition paths between the preceding state and the following state, \( b \) is the exponent of \( R_2(t) \) in \( P_{\text{prec}}(t) \) which represents the number of operating pairs in the preceding state.

For pair forcing down transitions, the probability expression is shown in Equation (4.5):

\[ P_{\text{f/dlt}}(t) = \eta \int_{\tau_1=0}^{t} P_{\text{prec}}(\tau_1) f_2(\tau_1) R_2(\tau_1)^{U-b} d\tau_2 R_2(t)^{b-U-1} \quad (4.5) \]

where \( U \) is the number of forced down pairs during the transition. If there are more than one preceding states, the probability of the following state is the sum of probabilities of transitions from all the preceding states to following state.

Given the initial state of the system, the probabilities of all the operational states can be obtained iteratively. The initial state is given by Equation (4.6):

\[ P_{(0,0,0,\ldots)}(t) = R_2(t)^{nm} \quad (4.6) \]

Based on the iterative equations, for this (1, 1)-out-of-(3, 2) PNPNPN hexacopter system, the probability that the system is in different operational states \((l_1, l_2, \ldots, l_n)\) is obtained in Equations (4.7)-(4.12):

\[ P_{(1,0,0)}(t) = R_2(t)^{6} \quad (4.7) \]

\[ P_{(1,0,0)}(t) = 6 \int_{\tau_1=0}^{t} f_2(\tau_1) d\tau R_2(t)^{5} = 6 F_2(t) R_2(t)^{5} \quad (4.8) \]
The reliability of the system is the sum of these probabilities as shown in Equation (4.13):

\[
R(t) = P_{(0,0,0)}(t) + P_{(1,0,0)}(t) + ... + P_{(2,5,1)}(t)
\]  

(4.13)

4.2.2 Reliability Estimation of \((k_1,k_2)\)-out-of-\((n,m)\) Pairs: G Balanced Systems:

Algorithm

When the number of pairs and/or levels increases, enumerating potential operational states becomes computationally expensive. Therefore, we develop an algorithm as follows: Part I obtains all operational states of the system by enumerating the transition paths and Part II obtains the operational states’ probability expressions and system reliability. We briefly describe the two parts below.

Part I: Obtaining Operational States
While enumerating the transition paths, the preceding state and the following state in one transition activity are stored for further probability calculation. The preceding states are denoted as $S_{\text{prec}}$ and are stored in a list $Q_{\text{state}}$ whose size is dynamically changing. $S_{\text{prec}}$ transits to a set of following states: $S_{\text{foll}}$, which is an $n \times n$ matrix. In this matrix, each row represents a following state: $S_{\text{foll}}^i$, $i = 1, 2, \ldots, n$. For each of the following operational state, an index $U$ is used to keep track of the number of forced down pairs during the transition between the preceding and following state. If the following states satisfy the system requirements, they are placed back into $Q_{\text{state}}$ and wait to be withdrawn as preceding states.

At the same time, the preceding state $S_{\text{prec}}$, following state $S_{\text{foll}}$, and index $U$ are stored in a list $C$ for future use of probability calculation. The steps are:

### Part I: Obtain Operational States ()

Set the initial state as a zero vector: $S_{\text{init}} = \vec{0}_{\text{on}}$, and place it in $Q_{\text{state}}$ as an element.

Set an empty list $C$ to keep all of the preceding and following states combinations.

while $Q_{\text{state}}$ is not empty do

withdraw one element $q$ from $Q_{\text{state}}$, set $S_{\text{prep}} = q$, delete $q$ from $Q_{\text{state}}$.

$S_{\text{foll}} = \vec{1}_{\text{next}} \times S_{\text{prec}} + \vec{1}_{n}$, $\vec{1}_{n}$ is an identity matrix of size $n \times n$.

for ($i = 1, i \leq n, i + +$ ) do

$U = 0$, if $S_{\text{fol}}^i (i) = m - k_2 + 1$ then

search for the next element on the left of the $i^{\text{th}}$ element in $S_{\text{foll}}^i$: $S_{\text{fol}}^i (j)$ such that $|j - i|$ is an odd number and $S_{\text{foll}}^i (j) \leq m - k_2$; $j = 1, \ldots, (i - 1), (i + 1), \ldots, n$.

Search for the next element on the right of the $i^{\text{th}}$ element in $S_{\text{foll}}^i$: $S_{\text{foll}}^i (l)$ such that $|l - i|$ is an odd number and $S_{\text{foll}}^i (l) \leq m - k_2$; $l = 1, \ldots, (i - 1), (i + 1), \ldots, n$.

$U = m - \max(S_{\text{foll}}^i (j), S_{\text{foll}}^i (l)) + k_2 - 1$.

$S_{\text{foll}}^i (\text{aug max}(S_{\text{foll}}^i (l), S_{\text{foll}}^i (j))) = M$, $M$ is a large number.

Find the number of elements in $S_{\text{foll}}^i$ that are larger than $m - k_2$ and set it to $h$.

if $h \leq n - k_2$ then
place $[S^i_{\text{fall}}, S_{\text{prec}}, U]$ in $C$ as a new element.

Place $S_{\text{fall}}^i$ in $Q_{\text{state}}$.

end if

end if

if $S^i_{\text{fall}}(i) < m - k + 1$ then

place $[S^i_{\text{fall}}, S_{\text{prec}}, U]$ in $C$ as a new element.

Place $S_{\text{fall}}^i$ in $Q_{\text{state}}$.

end if

end for

end while

In Part I, the operational state $S_{\text{fall}}$, its preceding state $S_{\text{prec}}$, and the number of forced down pairs $U$ during the transition are obtained and kept as elements in list $C$.

**Part II: Obtaining the Probability Expressions**

In Part II, the symbolic expressions for the operational states are obtained. Suppose that there are $g$ elements in list $C$, the probability expressions $p_l, l = 1, 2, ..., g$ for all operational states are obtained and stored. The steps are:

**Part II: Obtain Probability Expressions**

Order the list $C$ based on the number of failed pairs in the preceding states $S_{\text{prec}}$.

for $(l = 1, l \leq g, l + +)$ do

withdraw $l^{th}$ element $C_i = [S_{\text{prev}}, S_{\text{fall}}, U]$ from $C$.

if $S_{\text{prev}} = 0_{\text{vsa}}$ then

$p_l = \int_{\gamma_i = 0}^{\gamma_i} f_2(\tau_1) d\tau_1 R_2(t)^{\text{vsa}-1}$.

end if

if $S_{\text{prev}} \neq 0_{\text{vsa}}$ then

$p_l = 0$.

for $(k = 1, k < l, k + +)$ do

if $S_{\text{prev}} = S_{\text{fall}}$ then

obtain the exponent of $R_i(t)$ in $p_i : b$.
\[ p_i = p_i + \int_{\tau_i=0}^{T} p_k(\tau_1) f_2(\tau_1) R_2(\tau_1)^{U-b} R_2(t)^{b-U-1} d\tau_1. \]

**end if**

**end for**

place \([S_{presh}, S_{fallo}, U_1, p_1]\) back into the original place in \(C\).

**end for**

The reliability function for system is obtained: \( R(t) = \sum_{i=1}^{g} p_i(t) \).

In Part II, probability expressions for all the operational states are obtained. The system reliability expression is obtained by summing all these probability expressions.

**4.2.3 Numerical Examples**

In this section, scenarios and examples for the reliability estimation of \((k_1, k_2)\)-out-of-\((n, m)\) pairs: G balanced systems with different combinations of \(k_1, k_2, n, m\) are provided and compared.

**Scenario 1: Reliability estimation for systems as a function of the ratio \(\frac{k_2}{m}\) or \(\frac{k_1}{n}\)**

Two examples are provided to illustrate the reliability estimation for the \((k_1, k_2)\)-out-of-\((n, m)\) pairs: G balanced systems with a fixed ratio of \(\frac{k_1}{n}\) or \(\frac{k_2}{m}\). Figure 4.4 considers three systems with equal ratio \(\frac{k_1}{n} = \frac{1}{3}\). It is found, for systems with the fixed ratio \(\frac{k_1}{n}\), that increasing the ratio \(\frac{k_2}{m}\) leads to a lower system reliability at all times when compared with the original system. Similarly, in Figure 4.5, the ratio \(\frac{k_2}{m}\) is set to 0.5.
Increasing $\frac{k_1}{n}$ ratio leads to a lower system reliability at all times when compared with the original system. Increasing the ratio $\frac{k_2}{m}$ or $\frac{k_1}{n}$ implies that the total number of required operating units in the system is increased which results in the decrease of the system reliability.

**Figure 4.4** Reliability Function for Systems with Fixed Ratio $\frac{k_1}{n}$
**Figure 4.5** Reliability Function for Systems with Fixed Ratio $\frac{k_2}{m}$

*Scenario 2: Reliability estimation for systems when $k_1$, $k_2$, $n$, $m$ are proportionally increased*

As shown in Figure 4.6, increasing the values of $k_1$, $k_2$, $n$, $m$ proportionally leads to a higher reliability during $t < 30$. The reliability of (2, 2)-out-of-(4, 4) pairs: G balanced system is higher than the reliability of (1, 1)-out-of-(2, 2) pairs: G balanced system during $t < 30$. Then, the reliability of (2, 2)-out-of-(4, 4) pairs: G balanced system decreases and becomes lower than the reliability of (1, 1)-out-of-(2, 2) pairs: G balanced system.
Figure 4.6 Reliability Function for Systems with Proportional Increased $k_1, k_2, n, m$

Figure 4.7 shows reliability functions of systems when the values of $k_1$ and $n$ increase proportionally while the values of $k_2$ and $m$ are fixed. The reliability of (2, 2)-out-of-(4, 4) pairs: G balanced system is higher than the reliability of (1, 2)-out-of-(2, 4) pairs: G balanced system during $t \leq 10$. Then, the two reliability functions intersect and the reliability of (2, 2)-out-of-(4, 4) pairs: G balanced system decreases faster than the other one.
Figure 4.7 Reliability Function for Systems with Proportionally Increased $k_1$ and $n$

Figure 4.8 shows the reliability functions of (1, 1)-out-of-(3, 2) and (1, 2)-out-of-(3, 4) pairs: G balanced systems. The system with greater values of $k_2$ and $m$ ((1, 2)-out-of-(3, 4) pairs: G balanced system) shows higher reliability than the (1, 1)-out-of-(3, 2) pairs: G balanced system before the two reliability functions intersect at $t \approx 10$. 
From the above three examples, it is found that when the ratios of $\frac{k_1}{n}$ and $\frac{k_2}{m}$ are fixed, increasing the values of $k_1$ and $n$ or $k_2$ and $m$ leads to a higher reliability for a limited period of time comparing with the original systems. Then the reliability decreases faster and becomes lower than the original system. The reason for this phenomenon is that the increased number of available units ($n$ or $m$) provides a higher reliability when balance is not required. However, after a period of time, the number of available units decreases due to failure. The system with more units is more likely to be unbalanced and the number of operational states decreases significantly.

**Figure 4.8** Reliability Function for Systems with Proportionally Increased $k_2$ and $m$
**Scenario 3: Design of systems for given \( k_1 \times k_2 \) and \( n \times m \)**

Suppose that at least \( 2 \times k_1 \times k_2 \) units are required to operate properly for the system to provide its function and that the total number of available units is limited to \( 2 \times n \times m \). It is of interest to configure these units (determine the number of pairs and levels) to achieve the system’s highest reliability function. In the first example, 16 units are available, and four out of 16 units are required to operate properly for the system to function properly.

Figure 4.9 shows the reliability functions for the two system configurations: 1) \((1, 2)\)-out-of-(2, 4) pairs: G balanced system and 2) \((2, 1)\)-out-of-(4, 2) pairs: G balanced system.

With the values \( k_2 \), \( k_1 \), \( k_1 \times k_2 \) and \( n \times m \) being fixed, systems with a smaller number of levels \( m \) has a higher reliability than systems with a larger \( m \).

---

**Figure 4.9** Reliability Function for Systems with Fixed \( k_1 \times k_2 \), \( n \times m \) and \( \frac{k_2}{m} \), \( \frac{k_1}{n} \)
Suppose that 20 units are available and 4 out of 20 units are required for the system to function properly. Two system configurations are considered and compared: 1) (1, 2)-out-of-(2, 5) pairs: G balanced system and 2) (2, 1)-out-of-(5, 2) pairs: G balanced system. Figure 4.10 shows that the reliability for (2, 1)-out-of-(5, 2) system is slightly higher than (1, 2)-out-of-(2, 5) system. From these two examples, we conclude that when the number of available units and the minimum required operating units are given and fixed, systems with a small number of levels and a larger number of pairs are more reliable than other systems.

**Figure 4.10** Reliability function for systems with fixed $k_1 \times k_2$ and $n \times m$
4.3 Reliability Estimation of \((k_1, k_2)\)-out-of-\((n, m)\) Pairs: G Balanced Systems

Considering Forced-down Units as Standbys

4.3.1 Reliability Estimation of \((k_1, k_2)\)-out-of-\((n, m)\) Pairs: G Balanced Systems:

**Transition Enumeration**

In section 4.2, the forced-down units and pairs are considered failed. The main purpose of forcing down units and pairs is to maintain the balance of the system and to assure that adjacent operating pairs perform complementary functions when a failure occurs. However, as the system dynamically transit from one operational state to other states, the number and position of failed units may change and hence the balance situation may change too. It is possible that when new failures cause the unbalance of the system, the system returns to its operational state by resuming the previously forced-down units and pairs. Comparing to forcing down units, resuming the previously forced-down units and pairs is a better approach to bring the unbalanced system back to balance. In this section, the forced-down units and pairs are considered as standbys. When operating units in the system fail and cause the unbalance of the system, we first explore the possibility of resuming one or more units and/or pairs to resume the system balance.

There are two possible resuming activities of the standby units (pairs): 1) *unit resumption*: when one unit fails, one of the forced-down units on the same side of the same pair is resumed to maintain the balance instead of forcing down one of the opposite units in the same pair; 2) *pair resumption*: when a pair fails, instead of forcing down another adjacent operating pair to make sure that any adjacent operating pairs perform complementary functions, we check if a forced-down pair can be resumed. If there are more than one pair
of units that can be resumed, the pair with the greatest number of working units is resumed to provide the highest reliability. The unit resumption is explored in Chapter 3 and is shown in Figure 3.9. Different from Chapter 3, where all pairs in a system perform the same function, the systems discussed in this chapter have adjacent operating pairs performing complementary functions. The pair resuming activity is illustrated in Figure 4.11. For an (2, 1)-out-of-(4, 2) pairs: G balanced system, at least 1-out-of-2 unit is required to operate on both sides of a pair for this pair to be functioning. At least 2-out-of-4 pairs need to be operational and perform complementary functions for the system to be operational. In Figure 4.11-a, 4 pairs of operating units are evenly distributed on a circle. In Figure 4.11-b, the first pair failed (black color) and the second pair is forced down (grey color). In Figure 4.11-c, when the fourth pair fails, instead of forcing down the third pair and cause system failure, the second pair is resumed back to work.

![Diagram](image)

**Figure 4.11** Pair Resumption of the (2, 1)-out-of-(4, 2) Pairs: G Balanced System
Same as section 3.3.1, we use \((l_1, l_2)\) \((l_1, l_2)_{1,2}, \ldots (l_1, l_2)_n\) to represent the state of the system. In this vector, \(l_1\) and \(l_2\) are the numbers of failed units on left and right sides in \(i^{th}\) pair respectively, \(i = 1, 2, \ldots n\). There are \(n\) sub-vectors in this vector, the sequence of these sub-vectors represents the sequence of the \(n\) pairs in the system. For each pair, when a new failure occurs, if there are forced-down units on the same side of the same pair, one of the forced-down units is resumed to maintain the balance. However, if there are no forced-down units on the same side, one of the operating units on the opposite side of the same pair is forced down.

When the number of failed units is larger than \(m - k_2 + 1\) on either side of the \(i^{th}\) pair, this pair is considered failed. Considering adjacent pairs are required to perform complementary functions in this system, when a pair fails, we explore the possibility of resuming one of the forced-down pairs that perform the same function so that any adjacent pairs perform different functions. Otherwise, one of operating pairs that perform complementary function is forced down.

In this section, the (1, 1)-out-of-(3, 2) pairs: G balanced PNPNPN hexacopter system is introduced as an example to illustrate transitions between the operational states of the system and reliability calculation. Part of the transition diagram is shown in Figure 4.12. In this figure, transitions for some operational states are ignored. Because transitions for these ignored operational states are duplicated with other operational states and are represented using the same color circles. For example, both states \((2,0)\) \((0,0)\) \((0,0)\) and
The vectors \( \langle 2, 0, 0 \rangle \), \( \langle 2, 0, 1 \rangle \), \( \langle 2, 1, 0 \rangle \) are in red color. In every vector, there are three sub-vectors that represent the number of failed units for the three pairs in the system. The crossed-out sub-vector means that this pair is considered failed. The vector with a line in the middle means this pair is forced down. Even though the numbers of failed units in the first sub-vector of the two states: \( \langle 2, 0 \rangle \) and \( \langle 2, 1 \rangle \) are different, the two states are considered as duplicate states since the forced down pair (second sub-vector) and the working pair (third sub-vector) are the same. Therefore, only one state’s transition is shown in the figure. Similar situations include: state \( \langle 2, 0, 1, 0 \rangle \) and state \( \langle 2, 1, 0, 0 \rangle \); state \( \langle 2, 0, 1, 1 \rangle \) and state \( \langle 2, 1, 1, 0 \rangle \) and so on. In this way, the scale of transition diagram is largely reduced. Even so, comparing with the transition diagram in Figure 4.2 where there are only 7 states in total, this transition diagram has 37 states.
After obtaining the transition diagram for (1, 1)-out-of-(3, 2) pairs: G balanced PNP-NPN hexacopter system, the next step is to calculate the probability that the system is in one of the operational states \( ((l_1, l_2), (l_1, l_2), \ldots, (l_1, l_2)) \): \( P_{(l_1, l_2), (l_1, l_2), \ldots, (l_1, l_2)} (t) \) and the reliability of the entire system at arbitrary time \( t \): \( R_{sys} (t) \). \( P_{(l_1, l_2), (l_1, l_2), \ldots, (l_1, l_2)} (t) \) is a product of two parts: 1) number of possible transitions from the initial state \( ((0, 0), (0, 0), \ldots, (0, 0)) \) to the current state; 2) the probability that the initial state \( ((0, 0), (0, 0), \ldots, (0, 0)) \) transits to current state. The first part is discussed in section 4.2.2. In Figure 4.12, the number of possible transitions between any two states is marked on the arrows.
Similar to section 4.2.1, closed form expressions for these probabilities cannot be obtained. However, iterative calculations are provided based on the transitions between states. There are four types of transitions: 1) *unit resumption*: when one unit fails, we resume one of the previously forced-down units on the same side (if any exists); 2) *unit forcing down*: when one unit fails and there is no forced-down unit that can be resumed, then we force down one of the opposite units in the same pair; 3) *pair resumption*: when one unit’s failure causes the failure of the pair, then the remaining units of the same pair are forced down and one of the adjacent previously forced-down pairs that perform the same function is resumed (if any exists); 4) *pair forcing down*: when one unit’s failure causes the failure of the pair, then the remaining units of the same pair and one of the adjacent pairs that perform complementary function are forced down. The second and forth transition types are already discussed in section 4.2.1.

The pdf, CDF and reliability function for a single unit’s life time are \( f_1(t) \), \( F_1(t) \) and \( R_1(t) \) respectively. The probability that the system is in certain operational state at time \( t \): \( P_{\text{fol}}(t) \), is obtained by iterative calculations based on the probabilities that the system is in its preceding states, \( P_{\text{pre}}(t) \) as well as the transition between them. For unit resumption transition, the probability calculation is shown in Equation (4.14):

\[
P_{\text{fol}}(t) = \eta \sum_{\tau_1=0}^{t} P_{\text{pre}}(\tau_1) f_1(\tau_1) R_1(t-\tau_1) R_1(\tau_1)^{-b} d\tau R_1(t)^{b-1} \tag{4.14}
\]

where \( \eta \) is the number of transition path from preceding state to the following state, \( b \) is the exponent of \( R_1(t) \) in \( P_{\text{pre}}(t) \) which is the number of operating units in the preceding state.
For unit forcing down transition, the probability calculation is shown in Equation (4.15):

\[ P_{\text{full}} (t) = \int_{\tau_{i}=0}^{t} P_{\text{prec}} (\tau_{i}) f_{1} (\tau_{i}) R_{i} (\tau_{i})^{b} d\tau_{i} R_{i} (t)^{b-2} \quad (4.15) \]

For pair resumption transition, the probability calculation is shown in Equation (4.16):

\[ P_{\text{full}} (t) = \int_{\tau_{i}=0}^{t} P_{\text{prec}} (\tau_{i}) f_{1} (\tau_{i}) R_{i} (\tau_{i})^{U-b} R_{i} (t-\tau_{i})^{U} d\tau_{i} R_{i} (t)^{b-(U+1)} \quad (4.16) \]

where \( r \) is the number of resumed units in the resumed adjacent pair.

For pair forcing down transition, the probability calculation is shown in Equation (4.17):

\[ P_{\text{full}} (t) = \int_{\tau_{i}=0}^{t} P_{\text{prec}} (\tau_{i}) f_{1} (\tau_{i}) R_{i} (\tau_{i})^{U-b} d\tau_{i} R_{i} (t)^{b-(U+1)} \quad (4.17) \]

where \( U \) is the number of forced-down units during the transition.

The probability of the initial state \( ((0,0)(0,0)(0,0)) \) is shown in Equation (4.18):

\[ P_{((0,0)(0,0)(0,0))} (t) = R_{i} (t)^{2mn} \quad (4.18) \]

If the current state has more than one preceding states, the probability for this state should include all the probabilities that the preceding states transits to current state.

For the (1, 1)-out-of-(3, 2) pairs: G balanced PNPNP hexacopter system, the probability calculations for operational states \( ((0,0)(0,0)(0,0)) \), \( ((1,1)(0,0)(0,0)) \), \( ((0,0),(0,0),(0,0)) \) and \( ((0,0),(0,0),(0,0)) \) are shown in Equations (4.19)-(4.22) as
examples for unit forcing down, unit resumption, pair forcing down and pair resumption transitions respectively.

\[
P_{((1,0),(0,0),(0,0))}(t) = 12 \int_{\tau_1 = 0}^{t} f_1(\tau_1) R_1(\tau_1) d\tau_1 R_1(t)^10 \quad (4.19)
\]

\[
P_{((1,1),(0,0),(0,0))}(t) = 12 \int_{\tau_2 = 0}^{t} \int_{\tau_1 = 0}^{\tau_2} f_1(\tau_1) R_1(\tau_1) f_1(\tau_2) R_1(\tau_2) (t - \tau_2) d\tau_1 d\tau_2 R_1(t)^9 \quad (4.20)
\]

\[
P_{([3,0],[3,0]],[3,0])}(t) = 12 \int_{\tau_2 = 0}^{t} \int_{\tau_1 = 0}^{\tau_2} f_1(\tau_1) R_1(\tau_1) f_1(\tau_2) R_1(\tau_2)^5 d\tau_1 d\tau_2 R_1(t)^4 \quad (4.21)
\]

\[
P_{([3,0],[3,0]],[3,0])}(t) = 12 \int_{\tau_2 = 0}^{t} \int_{\tau_1 = 0}^{\tau_2} \int_{\tau_3 = 0}^{\tau_1} \int_{\tau_4 = 0}^{\tau_2} f_1(\tau_1) R_1(\tau_1) f_1(\tau_2) R_1(\tau_2)^5 \times f_1(\tau_3) R_1(\tau_3) f_1(\tau_4) R_1(\tau_4) R_1(t - \tau_4)^4 d\tau_1 d\tau_2 d\tau_3 d\tau_4 \quad (4.22)
\]

After obtaining all of the probabilities for these operational states, the reliability is obtained in Equation (4.23):

\[
R_{sys}(t) = P_{((0,0),(0,0),(0,0))}(t) + P_{((1,0),(0,0),(0,0))}(t) + ... + P_{([3,0],[3,0],[3,0])}(t) \quad (4.23)
\]

### 4.3.2 Reliability Estimation of \((k_1,k_2)\)-out-of-\((n,m)\) Pairs: \(G,\) Balanced Systems: Algorithm

In section 4.2.2, an efficient algorithm is developed to enumerate the transition paths and obtain operational states and their probability expressions. Similarly, in this section, an algorithm is developed to obtain the operational states as well as their probability expressions. Different with previous algorithm, in this section, vector \(w\) is created to keep track of working status for each pair and therefore is a \(1 \times n\) vector. \(w(i) = 0/1/2, i = 1,2,...n\) denotes that the \(i^{th}\) pair is failed, working or forced down.
Vector $U$ is a $1 \times 2$ vector with the first element denoting the number of forced-down units and the second element denoting the number of resumed units. The steps of Part I are:

**Part I: Obtain Operational States ()**

Set initial state as a zero vector: $S_{\text{init}} = \hat{0}_{1\times2n}$. 

Set two vectors: $w = \hat{0}_{1\times n}$, $U = \hat{0}_{1\times 2}$ as index vectors, and place $[S_{\text{init}}, w, U]$ in $Q_{\text{state}}$ as an element.

Set an empty list $C$ to store all the preceding and following states combinations.

**while $Q_{\text{state}}$ is not empty do**

withdraw one element $q$ from $Q_{\text{state}}$, set $S_{\text{prec}} = q(1:2n)$, $w = q(2n+1, 3n)$,

$U = [3n+1: \text{end}]$, then delete $q$ from $Q_{\text{state}}$.

$S_{\text{foll}} = \hat{1}_{2n\times1} \times S_{\text{prec}} + \hat{1}_{2n}$, $\hat{1}_{2n}$ is an identity matrix of size $2n \times 2n$.

**for ($i=1, i \leq n, i++$) do**

if $w\left([\frac{i}{2}]\right) \neq 0$ then

break
endif

if $S_{\text{foll}}^{i}(i) = m - k_2 + 1$ then

set $w\left([\frac{i}{2}]\right) = 1$.

find the nearest left and right elements for $[\frac{i}{2}]$ element in $w$: $w(j)$ and $w(l)$ such that $w(j) = w(l) = 0$.

search elements $w(h)$ between $w(j)$ and $w(l)$ in vector $w$ such that $h - [\frac{i}{2}]$ is an even number and $w(h) = 2$; place $h$ in $\tilde{h}$.

if $\tilde{h}$ is not empty then

find $g = \text{aug min}_{g} \{ \max (S_{\text{foll}}^{i}(2g - 1), S_{\text{foll}}^{i}(2g)), g \in \tilde{h} \}$.

$w(g) = 0$, $U(1) = 2k_2 + 1$, $U(2) = 2(m - \max (S_{\text{prec}}^{i}(2g - 1), S_{\text{prec}}^{i}(2g)))$.

endif

if $\tilde{h}$ is empty then

$g = \text{aug max} \{ \max (S_{\text{prec}}^{i}(2j), S_{\text{prec}}^{i}(2j - 1)), \max (S_{\text{prec}}^{i}(2l), S_{\text{prec}}^{i}(2l - 1)) \}$.

$w(g) = 2$, $U(1) = 2k_2 - 1 + 2(m - \max (S_{\text{prec}}^{i}(2g), S_{\text{prec}}^{i}(2g - 1)))$, $U(2) = 0$.

endif

endif

if $S_{\text{foll}}^{i}(i) < m - k_2 + 1$ then
if \( \text{mod}(i, 2) = 0 \) and \( S'_{\text{full}}(i-1) < S'_{\text{full}}(i) \) or \( \text{mod}(i, 2) = 0 \) and
\( S'_{\text{full}}(i+1) > S'_{\text{full}}(i) \)

then
\[
U(1) = 1, \quad U(2) = 0.
\]
end if

if \( \text{mod}(i, 2) = 1 \) and \( S'_{\text{full}}(i-1) < S'_{\text{full}}(i) \) or \( \text{mod}(i, 2) = 1 \) and
\( S'_{\text{full}}(i+1) > S'_{\text{full}}(i) \)

then
\[
U(1) = 0, \quad U(2) = 1.
\]
end if

end if

find the number of elements in \( S'_{\text{full}} \) that are larger than \( m-k_2 \), set the number to \( j \).

if \( j \leq n-k_2 \) then

place \( [S'_{\text{full}}, S'_\text{prec}, U] \) in \( C \) as a new element.

place \( [S'_{\text{full}}, r, U] \) in \( Q_{\text{state}} \).

end if

end for
end while

In Part I, the operational state, its preceding state and the number of forced-down units as well as resumed units during transition are obtained and saved in the list \( C \). Suppose that there are \( g \) elements in list \( C \), the probability expressions \( p_l, l = 1, 2, ..., g \) for all operational states are obtained and stored.

Part II obtains the probability expressions for all the operational states. The reliability expression for the system is obtained by summing these probability expressions. The steps of Part II are:

**Part II: Probability Estimation()**

Order the list \( C \) based on the number of failed units in preceding states \( S'_{\text{prec}} \).

for \( (l = 1, l \leq g, l++) \) do

withdraw \( l^{th} \) element \( C_l = [S'_{\text{prec}}, S'_{\text{full}}, U_l] \) from \( C \)
if $S_{prec} = \bar{0}_{1 \times 2}$ then
$$p_i = \int_{\tau_1=0}^{t} f_i(\tau_1) d\tau_1 R(t) \tau^{2m-1}$$
end if

if $S_{prec} \neq \bar{0}_{1 \times 2}$ then
$$p_i = 0$$
for $i = 1, k < l, k++$ do
if $S_{prec} = S_{foll}$ then
obtain the exponent of $R_i(t)$ in $p_k$: $b$
$$p_i = p_i + \int_{\tau_2=0}^{t} p_k(\tau_2) f_i(\tau_1) R_i(\tau_1) \tau_i^{b(U(1)-b)} R_i(t-\tau_1)^{U(2)} R_i(t)^{b-(U(1)+1)} d\tau_1$$
end if
end for
place $[S_{prec}, S_{foll}, U_i, p_i]$ back into the original place in $C$
end for

The reliability function for system can be obtained: $R(t) = \sum_{i=1}^{n} p_i(t)$

4.3.3 Numerical Example

In this section, an example is introduced to illustrate the reliability estimation of $(k_1, k_2)$-out-of-$(n, m)$ pairs: G balanced system when the forced-down units are considered as standbys. For an $(1, 2)$-out-of-$(2, 4)$ pairs: G balanced system, the system reliability is obtained when: 1) forced-down units are considered failed and 2) forced-down units are considered as standbys. The reliability of this system is calculated based on the algorithms introduced in section 4.2.2 and 4.3.2. From Figure 4.13, it is shown that when forced-down units are considered as standbys, the system reliability is higher at all times than system reliability when forced-down units are considered as failed.
4.4 Conclusions

In this chapter, we estimate the reliability of \((k_1, k_2)\)-out-of-\((n, m)\) pairs: G balanced system when adjacent units performing complementary functions. Similar to Chapter 3, system balance is required at all the times. In addition, any adjacent pairs should perform complementary functions. To satisfy these two requirements, when new failure occurs, applicable operational control includes 1) forcing down units and/or whole pair; 2) resuming standby units and/or pair. Thus, in this chapter, two scenarios are considered when estimating the reliability for such systems: 1) forced-down units are considered as failed; 2) forced-down units are considered as standbys and are resumed when needed.
Under both scenarios, the system’s transitions from the initial state to all the other operational states are enumerated. The probability that the system is in one of the operational states is obtained by iterative calculations. The iterative calculations are given based on the transition activity history from the initial state. The reliability for the system at time \( t \) is the summation of the probabilities of all the operational states. The challenges and difficulties of reliability estimation for such systems include the enumeration of transitions between operational states and the multiple integral calculations. As the numbers of \( n \) and \( m \) increase, the reliability estimation becomes computational expensive. Two algorithms are developed to simplify the calculation. Based on the analysis of numerical examples, it is found that when the number of available units are fixed, the system is more reliable when a smaller number of levels and a greater number of pairs are used. It is also found that the reliability for systems when forced-down units can be resumed is higher than reliability for systems when forced-down units are considered as failed at all times.
CHAPTER 5

RELIABILITY ESTIMATION OF SPHERICALLY BALANCED SYSTEMS

Reliability estimation models are developed for multi-dimensional balanced systems with units arranged in a multi-level circular configuration in Chapter 3 and 4. These systems are generalized as \((k_1, k_2)\)-out-of-\((n, m)\) pairs: \(G\) balanced systems. In this chapter, we expand the work to the spherically balanced systems. Balanced systems with spherically distributed units are widely used in many applications such as spherical unmanned vehicles (SUV) for oceanography and nuclear detection where rotors are located on multiple planes of a sphere. These vehicles may fail to perform its function due to insufficient and/or unbalanced thrust. These SUV systems have different configurations depending on system designs. Figure 5.1 shows an SUV named Flying Eye developed by the University of Manchester where six rotors are located on three planes of a sphere. The proliferation of spherically balanced systems has heightened the importance of the investigation of their reliability. In this chapter, we develop methods and algorithms for reliability assessment of such systems.

![Figure 5.1 Spherical Unmanned Vehicles](image-url)
5.1 Problem Definition and Assumptions

There are several configurations of spherically balanced systems. In this chapter, we generalize these systems as spherical $k$-$n$-$i$: $G$ balanced systems where $k$, $n$ and $i$ are the minimum number of units required for the system to function properly, the number of units on each plane and the number of planes respectively as described in section 5.1.1.

5.1.1 System Description

For balanced systems with units distributed on a spherical surface, the locations of units have a profound effect on system balance as well as its reliability estimation. Inspired by the SUV in Figure 5.1, we assume that all of the units are located on $i$ planes of the sphere. These $i$ planes intersect with each other at the sphere’s top and bottom points and evenly divide the sphere into equal parts. Assuming that for each plane, there are $n$ units evenly distributed on the circumference of the plane. Two units are located on the planes’ intersection points which are the sphere’s top and bottom points and hence are shared by all of the $i$ planes. The system has a total of $i \times (n - 2) + 2$ units when $i \geq 2$. A total of $k$ balanced units located on any plane are required to work properly for the system to function properly. These minimal $k$ operating units can be located on any plane.

For example, for a system with one plane ($i = 1$) as shown in Figure 5.2-a, the sphere is divided into two hemispheres. There are eight units ($n = 8$) located on the circumference of the plane which is also the equator of the sphere. Since all of the eight units are located on one plane, it becomes a two-dimensional system with units configured circularly as given by Hua and Elsayed (2016). For the system with units located on three planes ($i = 3$)
as shown in Figure 5.2-b, the sphere is divided into six equal parts. There are 16 units \((n=16)\) evenly located on the circumference of each plane shown in red, blue and green colors. The three planes intersect at the sphere’s top and bottom points and the two units located on these two points are shared by the three planes. There are 44 \((i \times (n-2) + 2 = 3 \times 14 + 2 = 44)\) units in total. We refer to such systems as spherical \(k-n-i: G\) balanced systems with \(i = 3m, m = 1, 2, ..., l\).

![Figure 5.2 Spherically Balanced Systems with i Planes](image)

Similar to the multi-level configurations presented in chapters 3 and 4, we investigate the operational balance of the spherical \(k-n-i: G\) balanced system in this chapter. When failures occur in the spherical configuration, they may cause system imbalance, we propose to force down units on certain positions to maintain the system balance. In this chapter, system balance has two aspects: the first is to avoid the rotational motion by ensuring the torques are zeros in three axes (yaw, pitch and roll) and the second is to ensure the system stability by achieving a symmetrical balance (explained later). The two aspects of balance are
independent, and both are required for the spherical system to be considered as balanced-operational system.

Generally, for systems with spherically distributed units, there are six degrees of freedoms (dof) to be considered: surge, sway, heave, roll, pitch and yaw as shown in Figure 5.3. Some of the existing spherical systems only utilize part of them during actual movements. The micro Underwater Unmanned Vehicles (UUV) shown in Figure 2.3 is ballasted so that the UUV will not pitch nor roll. However, this may not be applicable for all the spherically balanced systems. For example, the SUV with rotary wings in Figure 5.1 has six rotors providing thrusts in six different directions and there is no specific mechanism to avoid its motion in any of the six dof. We categorize the motions of a spherically balanced system into two categories: 1) rotational motion including roll, pitch and yaw and 2) translational motion including surge, sway and heave. In this chapter, the system is rotationally balanced when the moments about each of the three axes (yaw, pitch and roll) equal to zero at all the times. We obtain the rotational balance by forcing down units on other locations on the sphere when imbalance occurs.
In 2016, Federal Aviation Administration (FAA) published a study guide for small unmanned aircraft systems in which it defines the stability of an UAV as the ability to correct for conditions that may disturb its equilibrium and to return to or to continue on the original flight path (Federal Aviation Administration, 2016). When a failure or a fault occurs and disturbs the system’s equilibrium, the system loses its rotational balance in pitch, roll and yaw axes. Even though the rotational balance can be recovered by forcing down other operational units, there are two concerns that needs to be addressed: 1) how large are the variations in terms of moments of pitch, yaw, and roll and 2) how fast the system recovers to the original flight status. If the variations are too large or the recovering time is too long, the system is considered unstable. Thus, among the forcing down choices that bring the system to rotational balance, it is important to choose one that provides the most stability.
It is shown that when the operating units in a multi-dimensional system are symmetric with respect to a set of axes, the variations of rotational moments and duration of recovery time are relatively small and hence brings immediate system stability. For example, Marks et al. (2012) investigate four failure scenarios for a Vertical Take-off and Landing (VTOL) octorotor. It is found that when failures occur, if the system is rotationally balanced and the operational units are symmetric with respect to two axes, the variations in pitch, yaw and roll moments are negligible. Saied et al. (2017) propose a control scheme to force down one motor on the opposite side of the failed one and ensures the symmetry of operational motors for a coaxial octorotor. It takes less than 1 second for the octorotor to recover to its original flight path whether the octorotor is in a hovering position or follows a circular trajectory. In this chapter, for the stability of spherical k-n-i: G balanced system, it is required that the operational units to be in a symmetrical pattern. The details of both rotational balance and symmetrical balance are discussed in the next two sections beginning with rotational balance.

### 5.1.2 Rotational Balance

To maintain a spherical system’s rotational balance, the locations and the directions of the propulsions of the unit located on the spherical system are equally important. The locations of the propulsions (units’ configuration) are discussed in the previous section. In this section, the directions of the propulsions are assumed to be tangent to the spherical surface and are either in the clockwise or counterclockwise except the two units located on the top and bottom of the sphere since they are shared by all of the planes. The directions of their propulsions of these units are along the z axis in either the upward or the downward
direction. It is required that the propulsion directions for units on the same plane to be in the same direction; either clockwise or counterclockwise.

For a spherically balanced system with \( i \) planes where \( i = 3m \), we assume that the propulsion direction of \((j + m)^{th}\), \( j = 1, \ldots, m \) planes is clockwise or counterclockwise while the rest of \( 2m \) planes are in the opposite direction. Figure 5.4 shows the propulsion directions for spherically balanced systems with three planes and six planes. The numbers in the circles are the indices of units. For system with three planes shown in Figure 5.4-a, the propulsions’ directions are counterclockwise for the first (blue) and third (red) planes. The propulsions’ directions are clockwise for the second (green) plane. For the six-plane system shown in Figure 5.4-b, the propulsions’ directions are counterclockwise for the third and fourth planes (yellow and light blue) and clockwise for the other four planes.

![Figure 5.4 Propulsion of Spherically Balanced Systems with Three and Six Planes](image_url)
For a three-dimensional object, torque is a vector of three elements indicating the rotational moments along the three orthogonal axes (yaw, roll and pitch). It is given by the cross product of the position vector (where the forces are applied) and the force vector (indicates the force’s direction and magnitude). For a spherical \(k-n-i: G\) balanced system, torque associated with each unit is calculated by multiplying the unit’s position vector by its propulsion vector. For example, for the spherical \(20-16-3: G\) balanced system shown in Figure 5.5-a, the torque given by the second unit (filled in blue) on the first plane is calculated by: 
\[
\left[\begin{array}{c} 0.3827 \\ 0 \\ 0.9239 \\
\end{array}\right] \times \left[\begin{array}{c} -0.9239 \\ 0 \\ 0.3827 \\
\end{array}\right] = \left[\begin{array}{c} 0 \\ -1 \\ 0 \\
\end{array}\right].
\]
The vectors are shown in Figure 5.5-b.

![Figure 5.5 Torque Calculation](image)

The torque vectors of the first (blue), the second (green) and the third (red) plane are obtained as shown in Equation (5.1):
The units on the same plane except the top (first) and bottom (ninth) units have the same torque. The torques of the top and bottom units are zero since the units’ position vector and propulsion vector are parallel. The torque for the entire system is the summation of vectors \( \tau_1, \tau_2 \) and \( \tau_3 \) and is a zero vector. This spherical system is hence rotationally balanced.

The proposed spherical system is always rotationally balanced when all of the units are operating. However, if units (except the top and bottom units) on the \( l^{th} \) plane fail, the spherical system starts to rotate towards the opposite direction of the \( l^{th} \) plane’s propulsion.

To avoid this rotational motion, we force down units on other planes to counteract the torque. We enumerate all of the possible forcing down positions and calculate the torque for each forcing down option. When the torque equals to zero, the state is considered as rotationally balanced.

\[
\tau_1 = \begin{bmatrix} 0, & 0, & 0 \\ 0, & -1, & 0 \\ \vdots \\ 0, & -1, & 0 \end{bmatrix} \quad \tau_2 = \begin{bmatrix} 0, & 0, & 0 \\ -0.866, & 0.5, & 0 \\ \vdots \\ -0.866, & 0.5, & 0 \end{bmatrix} \quad \tau_3 = \begin{bmatrix} 0, & 0, & 0 \\ 0.866, & -0.5, & 0 \\ \vdots \\ 0.866, & -0.5, & 0 \end{bmatrix}
\] (5.1)

5.1.3 Symmetrical Balance

As discussed in 5.1.1, to ensure the system’s stability, the operating units in a multi-dimensional balanced system are required to be in a symmetrical pattern. For example, Hua and Elsayed (2016) require systems with circularly arranged units to be symmetric with respect to at least a pair of perpendicular axes. For the same system, Endharta et al. (2018) define the system balance as operating units spreading proportionately. In this new balance
definition, when the distance between the neighboring operating units has a unique pattern, the system is considered as balanced. Unlike the system in the above two studies, the spherical $k$-$n$-$i$: $G$ balanced systems are three-dimensional systems. Thus, instead of calculating the moments around axes to determine the system balance, the operating units in the spherically balanced systems are required to be symmetric with respect to three planes that pass through the sphere’s center.

A spherically balanced system can have infinite planes of symmetry. However, for the spherical $k$-$n$-$i$: $G$ balanced system, because of the specific system configuration of having the top and bottom units shared by the $i$ planes, we focus on finding the planes of symmetry that are orthogonal to the $x$-$y$ plane (horizontal plane) and pass through the center of sphere. For example, Figure 5.6 shows planes of symmetry for the spherical 30-8-3: $G$ balanced system when units on different locations fail (filled in black color). In Figure 5.6-a, it is obvious that the operating units are symmetrical with respect to the green plane which is orthogonal to the $x$-$y$ plane. We are not able to find another plane of symmetry. Thus, this system is not symmetrically balanced. In Figure 5.6-b, since the operating units are symmetrical with respect to the green, red and blue planes, this system is considered as symmetrically balanced.
For simplicity, all units on the spherical system are projected on the $x$-$y$ plane and the symmetrical balance can be determined from the top view of the system. A system is symmetrically balanced when the operating units are symmetric with respect to three axes from the top view. Figure 5.7 shows the top views of four states for a spherical 30-8-6: G balanced system. Except for the units located on the grey circle (sphere’s equator), each small circle represents two units, one from the upper hemisphere and another from the lower hemisphere. The black/grey circles represent that the units on these locations have failed/forced-down. Circles with right half covered in black/grey means the unit from the upper hemisphere have failed/forced-down. Circles with left half covered in black/grey means the unit from the lower hemisphere have failed/forced-down. The system state shown in Figure 5.7-a has three failed units from upper hemisphere and are symmetric according to three axes. However, with three more units failed on the equator, the system in Figure 5.7-b is not symmetric to any axes and hence not symmetrically balanced. For
the system in 5.7-b to maintain balance, three more units are forced down as shown in 5.7-c and 5.7-d. Since the number of forced-down units are the same, the system states in 5.7-c and 5.7-d are equally reliable. As the number of failed units increases, it becomes a challenge to identify and/or select the best force-down options.

Figure 5.7 Spherically Balanced Systems with Six Planes

In this section, we propose a mathematical approach to determine whether the operating units are symmetrical with respect to three axes after projecting all units on the $x$-$y$ plane for a spherical $k$-$n$-$i$: G balanced system. First, the system is represented as a set of line segments from the sphere’s equator to its center. For example, Figure 5.8 is the top view of a state of the spherical 30-8-6: G balanced system, the numbers around the grey circle (sphere’s equator) are the indices of line segments. The first segment is shown in red color. There are 12 ($2 \times i$) segments in total. On each segment, there are three small circles
representing the locations of units. The locations are ordered from the equator (if there is a unit on the equator) to the center of the sphere. The numbers in the small circles are the indices of locations. If the circle is black/grey/white, this unit is failed/forced-down/operating. We use a vector $S_j = [l'_j, \ldots, l''_j]$, $j = 1, \ldots, 12$ to represent the state of each segment. Let $l^g_j$ be the state of $g^{th}$ location on $j^{th}$ segment.

$$l^g_j = \begin{cases} 
0; & \text{no unit operating on } g^{th} \text{ location} \\
1; & \text{one unit from upper hemisphere operating on } g^{th} \text{ location} \\
3; & \text{one unit from lower hemisphere operating on } g^{th} \text{ location} \\
6; & \text{all units operating on } g^{th} \text{ location}
\end{cases}$$

![Figure 5.8 Unbalanced State for a Spherical 30-8-6: G Balanced System](image)

For example, in the state shown in Figure 5.8, $S_1 = S_5 = S_9 = [6, 3, 6]$, $S_2 = S_6 = S_{10} = [0, 6, 6]$ and $S_3 = S_4 = S_7 = S_8 = S_{11} = S_{12} = [6, 6, 6]$. Suppose there are $v$
locations on each segment. The whole system can be represented by a matrix $S$ of size $(2 \times i, v)$ as shown in Equation (5.2).

\[
S = \begin{bmatrix}
S_1 \\
S_2 \\
\vdots \\
S_{2i}
\end{bmatrix} = \begin{bmatrix}
I^1_1, \ldots, I^v_1 \\
I^1_2, \ldots, I^v_2 \\
\vdots \\
I^1_{2i}, \ldots, I^v_{2i}
\end{bmatrix}
\]  

(5.2)

Second, let $D_j$ be the distance matrix between segments $S_{j+1}$ and $S_j$. The matrix is obtained using Equation (5.3).

\[
D_j = \begin{cases}
S_{j+1} - S_j, & j = 1, \ldots, (2 \times i - 1) \\
S_1 - S_j, & j = 2 \times i
\end{cases}
\]  

(5.3)

and we obtain a distance matrix $D$ of the same size as matrix $S$ as shown in Equation (5.4).

\[
D = \begin{bmatrix}
D_1 \\
D_2 \\
\vdots \\
D_{2i}
\end{bmatrix}
\]  

(5.4)

Next, let $R_j$ be a reverse distance vector between segments $S_{j+1}$ and $S_j$. It is obtained by Equation (5.5) and shown in Equation (5.6).

\[
R_j = \begin{cases}
S_{j+1} - S_j, & j = 2, \ldots, (2 \times i) \\
S_{2i} - S_1, & j = 1
\end{cases}
\]  

(5.5)
For convenience of calculations, we repeat the rows of the reverse matrix and obtain matrix \( R' \) of size \((4\times i, v)\) as shown in Equation (5.7).

\[
R' = \begin{bmatrix}
R_1 \\
R_2 \\
\vdots \\
R_{2vi}
\end{bmatrix}
\]  

(5.7)

Next, let \( M^h \) be the reverse matrix for the \( h^{th} \) segment where \( h = 1, 2, \ldots 2i \). \( M^h \) is obtained by taking \( 2\times i \) rows reversely from matrix \( R' \) starting from \((h+2\times i)^{th}\) row. \( M^h \) is a matrix of size \((2\times i, v)\) and shown in Equation (5.8).

\[
M^h = \begin{bmatrix}
R_h \\
R_{h-1} \\
\vdots \\
R_1 \\
R_{2vi} \\
\vdots \\
R_{2vi-h+1}
\end{bmatrix}
\]  

(5.8)

Last, compare the \( M^h \) for \( h = 1, 2, \ldots 2i \), if \( M^h = D \), the system is symmetrical with respect to the axis in the middle of \( h^{th} \) segment and first segment. When there are more than three axes of symmetry, the system is symmetrically balanced.
5.2 Reliability Estimation of Spherical $k$-$n$-$i$: G Balanced Systems Considering Forced-down Units as Failed

5.2.1 Reliability Estimation of Spherical $k$-$n$-$i$: G Balanced Systems: Transition Enumeration

In this section, we present reliability estimation of the spherical $k$-$n$-$i$: G balanced system by enumerating the transitions among the system’s operational states. Suppose that all of the units on the spherical $k$-$n$-$i$: G balanced system are identical with i.i.d. lifetime distributions. A state is operational when at least $k$ units are working properly while the system is balanced. When a failure occurs, the system transits from one operational state to either another operational state (if any) or non-operational state. We obtain all the operational states of a spherical $k$-$n$-$i$: G balanced system by enumerating the one-step transitions for every operational state. For each operational state, the probability of occurrence is calculated based on the number of failed units as well as all the previous transitions. Considering that a working system can be on any operational states at any time, the reliability of the system is obtained by summing the probabilities of occurrence for all of the operational states.

Transitions from one operational state to another occur when the system encounters a new failure. This new failure may cause the failure of the entire system for two reasons: 1) there are less than $k$ working units in the system and 2) the system is unbalanced (rotational or/and symmetrical) due to the new failure and can’t be resumed by forcing down other units. If the new failure does not cause the failure of the entire system, this transition leads
to an operational state. The transition enumeration starts from the initial state when there are no failures in the system and ends when all the following states are not operational.

5.2.1.1 Forcing-down Positions Determination

During the transition enumeration, forcing down units is important in achieving a high system reliability. While looking for the best following state by enumerating the locations of force-down units, we need to: 1) determine if the state is rotationally balanced; 2) determine if the state is symmetrically balanced and 3) the total number of operating units is larger or equal to \( k \). For states that meet the above conditions, we select the one that results in the highest number of operating units. Enumerating the forcing-down choices for each transition is computationally expensive. Thus, in this section, we introduce some forcing down rules that can efficiently rebalance the spherical \( k-n-i: G \) balanced system when failure occurs.

Figure 5.9 shows the one-step transitions of a spherical \( k-n-i: G \) balanced system from the initial operational state. This spherical system has six planes \( (i = 6) \) and eight units on each plane \( (n = 8) \). There is a total of 38 units. It is required that at least 30 \( (k = 30) \) out of 38 units to be working for the system to be considered operating properly. In this section, we assume that the forced-down units are considered as failed and cannot be resumed. The numbers around the large grey circle (equator) are the indices of the line segments. In Figure 5.9-b and 5.9-c, the unit on the upper and lower hemisphere of the first segment fails which affects the spherical system in two aspects: 1) the torque causes the sphere to roll around the \( y \) axis; 2) the system is not symmetrical with respect to any three axes on \( x- \)
y plane. The “best” forcing down option is to force down the two units on the ninth and fifth segments shown in a grey color. The torque is counteracted, and the operating units are symmetrical to three axes shown in the red color. In Figure 5.9-d, the unit of first segment on the equator (black color) fails which affects the sphere in two aspects: 1) the torque causes the sphere to roll around the y axis and 2) the system is not symmetrical on x-y plane. Similarly, we force down the units on the ninth and fifth segments shown in grey. In Figure 5.9-e and 5.9-f, the top and bottom unit fails, these two failures do not affect the system’s balance. Thus, no units are forced down. The states in Figure 5.9-b to 5.9-f are operational.

Figure 5.9 State Transition for a Spherical 30-8-6: G Balanced System
Figure 5.10 shows the possible one-step transitions for the state in Figure 5.9-b in the spherical 30-8-6: G balanced system. For the states in 5.10-b to 5.10-e, the second failure occurs on similar positions as the first failure. Top or bottom units are failed in Figure 5.10-f and 5.10-g. Units from the equator are failed in Figure 5.10-h to 5.10-j. Except for the states in Figure 5.10-c and 5.10-h where no operational state can be obtained, other states are operational.

![Figure 5.10 State Transitions for a Spherical 30-8-6: G Balanced System](image)

From the above examples, it is obvious that the number of state transitions is large. Forcing down units on certain locations is required for almost every transition. In addition, the
increasing number of failed units brings complexity in the determination of forcing-down positions. Thus, we propose some simple rules for forcing down options as special cases where the system balance is easy to find. We first group the $i$ ($i = 3m; m = 1, 2, ...l$) segments into $2m$ groups with three segments in each group. The $j^{th}$ group includes the following segments: \([j^{th}, (j + m)^{th}, (j + 2m)^{th}], j = 1, ..., m\). There are two groups for systems with three planes. The first group includes the \([1^{st}, 3^{rd}, 5^{th}]\) segments. The second group includes the \([2^{nd}, 4^{th}, 6^{th}]\) segments. After obtaining the segments groups, we use the following rules to determine the locations of forced-down units when a failure occurs:

Rule 1: when the top or bottom units fail, no forcing down is required since the propulsions on these two positions have no effect on the system balance.

Rule 2: when a unit on $j^{th}$ segment fails and it is the system’s first failure, one needs to force down the units on the same position of the other two segments in the same group. The axes of symmetry are along the diameters that pass through these three units.

Rule 3: when another unit fails and all existing failed/forced-down units are on the same location and in a consecutive arrangement (see Figure 5.10-b and 5.10-d), one needs to force down the units on the same position of the other two segments in the same group. The axes of symmetry are in the middle of consecutive failed/forced-down units.
Rule 4: when another unit fails and all existing failed and forced-down units are on the segments of same group (see Figure 5.10-e, 5.10-i and 5.10-j), one needs to force down the units on the same position of the other two segments in the same group. The axes of symmetry are along the diameters that pass through these three units.

When the system states do not fall into the rules discussed above, we enumerate all the possible forcing down positions and select the “best” forcing down option meaning having the minimum number of forced-down units. While enumerating the possible positions for forcing down, one needs to ensure that the state is rotationally and symmetrically balanced. Sections 5.1.2 and 5.1.3 introduce the method to determine whether a state is balanced in rotational and symmetrical aspects.

5.2.1.2 Probability Calculation for Operational States

All the operational states are obtained by enumerating the one-step transitions for each of them until all the following states are non-operational. After obtaining all of the operational states, we calculate the probability of occurrence for each state. For each transition, there are two states involved: preceding state and its probability $P_{prec}(t)$ and following state and its probability $P_{ foll}(t)$. Suppose that the probability density function (pdf) and reliability function for single unit’s lifetime are $f(t)$ and $R(t)$ respectively. The number of forced-down units during one transition is $u$. The probability of occurrence of the following state is obtained by Equation (5.9).

$$P_{foll} = \int_{\tau=0}^{\tau} P_{ prem}(\tau) \times R^{(-1)} \times f d\tau R^{(-u)}$$ (5.9)
where $l$ is the exponent of $R(t)$ in $P_{prec}(t)$.

The probability of occurrence for the initial state in which there are no failures is given in Equation (5.10).

$$P_{init} = R(t)^{i\times(n-2)+2}$$  \hfill(5.10)

The system reliability is obtained by summing all these probabilities as given in Equation (5.11).

$$R_{sys}(t) = \sum_{j=1}^{N} P_j(t)$$  \hfill(5.11)

where $N$ is the total number of operational states.

### 5.2.2 Reliability Estimation of Spherical $k$-$n$-$i$: $G$ Balanced Systems: Algorithm

When the number of units on each plane ($n$) and the number of planes ($i$) are large, enumerating the operational states becomes computationally expensive. Therefore, we develop an algorithm: Part I enumerates the transitions among the operational states and obtain all of the operational states, and Part II calculates the probability of occurrence for each state. In the algorithm, we use a matrix $S = \begin{bmatrix} 1^1_1, \ldots, 1^n_1 \\ 1^1_2, \ldots, 1^n_2 \\ \vdots \\ 1^1_i, \ldots, 1^n_i \end{bmatrix}$ of size $i \times n$ to represent a state of the spherical $k$-$n$-$i$: $G$ balanced system. Each row of the matrix represents the state of units on one plane. For example, $\begin{bmatrix} 1^1_j, \ldots, 1^n_j \end{bmatrix}$ is the state of $n$ units on the $j^{th}$ plane. $1^l_j = 1$ if
the $l^{th}$ unit is operating and $l_j' = 0$ when it is failed or forced down. Figure 5.11-a is the top view of a spherical 30-8-6: G balanced system. The numbers beside the large grey circle (sphere’s equator) are the indices of each plane. Figure 5.11-b is the front view of the fourth plane. The numbers besides the blue circles are the indices for the units. Thus, the state of fourth plane is $[l_4^1,...l_4^6]=[1,0,1,1,0,1,1]$. The state matrix for the operational state in

![Figure 5.11](image)

Figure 5.11-a is $S = \begin{bmatrix} 0,1,1,1,1,1,1; \\ 1,0,1,1,1,0,1; \\ 1,1,1,1,1,0,1; \\ 1,0,1,1,0,1,1; \\ 0,1,1,1,1,1,1; \\ 1,0,1,1,0,1,1; \end{bmatrix}$.

**Figure 5.11** Spherical 30-8-6: G Balanced System

In Part I, while enumerating the transitions, the preceding state $S_{\text{prec}}$ and the following state $S_{\text{fol}}$ in one transition activity are stored in a list $C$ for further probability calculation.
The list $Q_{state}$ is used to store the preceding state and is dynamically changing. For each of the following operational state, a number $V$ is used to keep track of the number of forced-down units during the transition between the preceding and the following state. To reduce the number of repetitive calculations, a number $U$ is used to record the number of duplicated transitions. If the following states satisfy the system requirements, they are placed back into $Q_{state}$ and wait to be withdrawn as preceding states. At the same time, the preceding state $S_{prec}$, following state $S_{ foll }$, index $V$ and number $U$ are stored in a list $C$ as a successful transition for future use of probability calculation. The steps of Part I are:

**Part I: Obtain Operational States ()**

Set the initial state as a matrix of ones: $S_{init} = \mathbf{1}_{\times n}$, and place it in $Q_{state}$ as an element.

Set an empty list $C$ to keep all the preceding and following states combinations.

while $Q_{state}$ is not empty do

withdraw one element $q$ from $Q_{state}$, set $S_{prec} = q$, delete $q$ from $Q_{state}$.

for $(row = 1, \ row \leq i, \ row ++ )$ do

for $(col = 1, \ col \leq n, \ col ++ )$ do

if $S_{prec}(row, col) \neq 0$ do

$U = 1$.

if $col = n$ or $col = \frac{n}{2}$ do

$S_{prec}(:, col) = 0; S_{foll} = S_{prec}$.

Search for duplicate of $[S_{foll}, q]$ in list $C$ for the first two columns.

if find duplicate in $[S_{foll}, q, U', V']$, where $[S_{foll}, q'] = [S_{foll}, q]$ do

assign $U'$ to $U' + 1$.

else

$W = \sum (S_{prec}(row) - (i-1) \times (S_{prec}(row, n) + S_{prec}(row, \frac{n}{2})); V = 0$.

if $W \geq k$ do

place $[S_{foll}, q, U, V]$ in $C$ as a new element.

if $W \geq k + 1$ do

place $S_{foll}$ in $Q_{state}$.

end if

end if

end if
end if
else
\[ S_{prec}(row,col) = 0. \]
call function balance on \( S_{prec} \).
if function balance returns a balanced state \( S_{fail} \) do
search for duplicate of \([S_{fail}, q]\) in list C based on the first two columns.
if find duplicate in \([S'_{fail}, q', U', V']\) where \([S'_{fail}, q'] = [S_{fail}, q]\) do
assign \( U' \) to \( U' + 1 \).
else
\[ W_{prec} = \text{sum}(S_{prec}) - (i - 1) \times \left( S_{prec}(row,n) + S_{prec}(row,n/2) \right). \]
\[ W_{fail} = \text{sum}(S_{fail}) - (i - 1) \times \left( S_{fail}(row,n) + S_{fail}(row,n/2) \right). \]
\[ V = W_{prec} - W_{fail} - 1. \]
if \( W_{fail} \geq k \) do
place \([S_{fail}, q, U, V]\), in \( C \) as a new element.
if \( W_{fail} \geq k + 1 \) do
place \( S_{fail} \) in \( Q_{state} \).
end all if clauses
end all for loops
end while

In Part I, function \textbf{balance} is called to search for the “best” force-down positions when a new failure occurs on the \( g^{th} \) position of \( l^{th} \) plane. The state waiting to be balanced is represented as: \( S_{prec} \). This function returns the best operational state (if any) by forcing down the minimum number of units. The steps of the function \textbf{balance} are:

\begin{verbatim}
Function: balance()
\end{verbatim}

Place \( S_{prec} \) in a list \( Q \) as an element.
\textbf{while} \( Q \) is not empty \textbf{do}
withdraw one element \( q \) from \( Q \), set \( S_{prec} = q \), delete \( q \) from \( Q \).
Obtain the number of operating units: \( W = \text{sum}(S_{prec}) - (i - 1) \times \left( S_{prec}(1,n) + S_{prec}(1,n/2) \right). \)
\textbf{for} \((d = 1, d \leq (W - k), d++) \textbf{do}
\textbf{for} \((row = 1, row \leq i, row++) \textbf{do}
\end{verbatim}
for \((col = 1, \ col \leq n, \ \ col ++)\) do

if \(S'_{\text{prec}}(\text{row}, \ col) \neq 0\) do

\[
\text{indicator} = 1.
\]

if \(\ col = n\ \text{or} \ \ col = n/2\) do

\[
S' = S'_{\text{prec}}; \ S'(\cdot, \ col) = 0.
\]

else

\[
S' = S'_{\text{prec}}; \ S'(\text{row}, \ col) = 0.
\]

end if

for \((j = 1, \ j \leq \frac{i}{3}, \ j ++)\) do

if \(\text{sum}(S'(j; \cdot)) = \text{sum}(S'(j + \frac{i}{3}; \cdot)) = \text{sum}(S'(j + \frac{2i}{3}; \cdot))\) do

\[
\text{indicator} = \text{indicator} \times 1.
\]

else

\[
\text{indicator} = \text{indicator} \times 0.
\]

end if

end for

if \(\text{indicator} = 1\) do

for \((l = 1, \ l \leq i, \ l ++)\) do

\[
S'(l, \cdot) = \text{flip}\left(S'(l, \cdot; \frac{n}{4}) + \text{flip}\left(S'(l, \cdot; \frac{n}{4}; \frac{n}{2} - 1)\right)\right).
\]

\[
S'(l + i, \cdot) = \text{flip}\left(S'(l, \cdot; \frac{n}{2} + 1) + 3\frac{n}{4} + \text{flip}\left(S'(l, \cdot; \frac{3n}{4}; (n - 1))\right)\right).
\]

end for

\[
S'(1, \cdot) = S'(1, \cdot)/2.
\]

\[
S'(\cdot, 2) = 3.
\]

for \((j = 1, \ j \leq 2i - 1, \ j ++)\) do

\[
d(j, \cdot) = S'(i + 1, \cdot) - S'(i, \cdot).
\]

end for

\[
d(2i, \cdot) = S'(1, \cdot) - S'(2i, \cdot).
\]

\[
doubled(2i, \cdot) = S'; S'.
\]

for \((j = 1, \ j \leq 2i, \ j ++)\) do

\[
rd(\cdot, j) = -\text{flip}(\text{doubled}(j: j + 2i - 1, \cdot)).
\]

end for

for \((j = 1, \ j \leq 2i, \ j ++)\) do

if \(d = rd(\cdot, j)\) do

\[
\text{balanceaxis} = [\text{balanceaxis}; j].
\]

end if

end for
After obtaining all of the operational states, Part II is developed to calculate the probability of occurrence for each operational state. The system reliability is then obtained by summing the probabilities. The steps are:

**Part II: Obtain_Probability_Expressions()**

Order the g elements in list C based on the number of failed units in the preceding states $S_{\text{prec}}$.

for $(l = 1, l \leq g, l++)$ do

withdraw $l^{th}$ element $C_l = [S_{\text{foll}}^l, S_{\text{prec}}^l, U^l, V^l]$ from C.

if $S_{\text{foll}}^l = 1_{\text{cn}}$ do

$$P_{\text{foll}}^l = R(t)^{(u-2)\cdot 2}.$$  

place $C_l = [S_{\text{foll}}^l, S_{\text{prec}}^l, U^l, V^l, P_{\text{foll}}^l]$ back into the original place in C.

else

for $(j = 1, j \leq l, j++)$ do

if $S_{\text{prec}}^i = S_{\text{foll}}^l$ do

$$P_{\text{prec}}(t) = P_{\text{foll}}(t).$$

end if

end for

$$P_{\text{foll}}^l = U^l \int_{\tau_{\text{cn}}}^{\tau} P_{\text{prec}}(\tau) \times f(\tau) \times R(\tau)^{b+y^l} \ d\tau R(t)^{b-(y^l+1)}. $$
b is the exponent of \(R(t)\) in \(P_{\text{prec}}(t)\).

The reliability function for system can be obtained:

\[
R(t) = \sum_{i=1}^{g} P_{\text{fail}}(t).
\]
force down a total of 14 units to maintain the system balance. However, when forced down units are considered as standbys, after resuming three previously forced-down units and forcing down two other units on certain locations, the system is balanced with respect to the three axes as shown in Figure 5.12-d. In this case, only eight units are forced down. Obviously, the state in Figure 5.12-d is preferable considering less units are forced down comparing with the state shown in Figure 5.12-c.

![Figure 5.12 State Transitions of a Spherical 10-8-6: G Balanced System](image)

With forced-down units considered as standbys, the system reliability is expected to be higher since less units are forced down due to imbalance. At the same time, an increase of complexity in transition enumeration between operational states and probability calculation is expected. Because for each operational state, the units in this state can be
previously forced down and resumed for several times depending on the locations and sequence of failed units (see Figure 5.13).

5.3.1.1 Resuming/Forcing-down Positions Determination

As we discussed in section 5.2, in a spherical $k-n-i: G$ balanced system, state transitions occur whenever units fail. For each transition, the choice of resuming and/or forcing down units is important in achieving a high system reliability. While looking for the best following state by enumerating the choices of resuming and force-down units, we need to:

1) determine if the state is rotationally balanced; 2) determine if the state is symmetrically balanced and 3) the total number of operating units is larger or equal to $k$. For states that satisfy the above conditions, we select the one that results in the largest number of operating units.
Figure 5.13 State Transitions of a Spherical 30-8-9: G Balanced System

Enumerating the resuming/forcing-down choices for each transition is computationally expensive. Because the units in each operational state can be previously forced down and resumed for several times depending on the locations and sequence of failed units. For example, Figure 5.13 shows one of the transition paths of a spherical 10-8-9: G balanced system. In this example, we track the forcing-down and resuming activities for the unit with red circle. The unit is operating in Figure 5.13-a, b and c. It is forced down in Figure 5.13-b and stayed as a standby unit in Figure 5.13-e and 5.13-f. In Figure 5.13-g, this unit
is resumed as an operating unit. However, in Figure 5.13-h, this unit is forced down again to balance the system.

In each transition, all of the possible resuming and forcing down positions are enumerated in order to achieve a high system reliability. For efficiency, we introduce some resuming/forcing-down rules that can rebalance the spherical $k$-$n$-$i$: $G$ balanced system when failure occurs. As the same in section 5.2.1, we first group the $i (i = 3m; m = 1, 2, \ldots)$ planes into $2m$ groups with three segments in each group. The $j^{th}$ group includes the following segments: $\left[ j^{th}, (j+m)^{th}, (j+2m)^{th} \right]$, $j = 1, \ldots, m$. After obtaining the segment groups, we use the following rules to determine the locations of resumed/forced-down units when a failure occurs:

Rule 1: when the top or bottom units fail, no resumption or forcing down is required since the units on these two positions have no effect on the system balance.

Rule 2: when a unit on $j^{th}$ segment fails and it is the system’s first failure, one needs to force down the units on the same position of the other two segments in the same group. The axes of symmetry are along the diameters that pass through these three units.

Rule 3: when all the failed units are from the same segment group (Figure 5.12-a, Figure 5.13-b and c), no resumption is required. One needs to force down units in the same
position on the segments from the same group. The axes of symmetry are along the
diameters that pass through these failed units.

Rule 4: when another unit fails and all the failed units are in a consecutive arrangement
(Figure 5.12-d), one needs to resume all the previously forced-down units that are not in
the same group and force down units in the same position of the same group. The axes of
symmetry are in the middle of consecutive failed/forced-down units.

When the states do not fall into any of the four situations above, we need to enumerate all
the possible resuming and forcing down positions and select the “best” state meaning
having the maximum number of working units.

5.3.1.2 Probability Calculation for Operational States

Similar with section 5.2.1.2, all of the operational states of a spherical $k$-$n$-$i$: G balanced
system are obtained by enumerating the one-step transitions until all the following states
are non-operational. The next step is to calculate the probability of occurrence for each
state. For each transition, there are two states involved: preceding state and its probability
$P_{prec}$ and following state and its probability $P_{foll}$. Suppose that the probability density
function (pdf) and reliability function for single unit’s lifetime are $f(t)$ and $R(t)$
respectively. The number of forced-down units during one transition is $u$. The number of
resumed units is $r$. The probability of occurrence of the following state is obtained by
Equation (5.12).

$$P_{foll} = \int_{\tau=0}^{t} P_{prec}(\tau) \times f(\tau) \times R(\tau)^u \times R(t-\tau)^r \times R(t)^{-(l+u+r)} \ d\tau \ R(t)^{(l-(u+r+l))} \quad (5.12)$$
where \( l \) is the exponent of \( R(t) \) in \( P_{prec}(t) \).

The probability of occurrence for the initial state in which there are no failures is given in Equation (5.13).

\[
P_{init} = R(t)^{i(n-2)+2}
\]  

(5.13)

The system reliability is obtained by summing all these probabilities as given in Equation (5.14).

\[
R_{sys}(t) = \sum_{j=1}^{N} P_j(t)
\]  

(5.14)

where \( N \) is the total number of operational states.

### 5.3.2 Reliability Estimation of Spherical \( k-n-i: G \) Balanced Systems: Algorithm

In this section, we develop an algorithm to enumerate the operational states and obtain the system reliability when the number of units on each plane \( (n) \) and the number of planes \( (i) \) are large. In Part I: all of the operational states are obtained by enumerating the transitions among them. In Part II: the probability of occurrence for each operational state is calculated.

Same as section 5.2.2, we use a matrix \( S = \begin{bmatrix} l_1^n & \cdots & l_1^n \\ l_2^n & \cdots & l_2^n \\ \vdots & \ddots & \vdots \\ l_i^n & \cdots & l_i^n \end{bmatrix} \) of size \( i \times n \) to represent a state of the spherical \( k-n-i: G \) balanced system. Each row of the matrix represents the state of units on one plane. For example, \( [l_j^1, \ldots, l_j^n] \) is the state of \( n \) units on \( j^{th} \) plane. In this algorithm, since the forced-down units are standbys, we distinguish them from the failed
units mathematically. Thus, \( l'_j = 1 \) if the \( j^{th} \) unit on \( f^{th} \) plane is operating; \( l'_j = 0 \) when it is failed and \( l'_j = 3 \) when it is forced down.

In Part I, while enumerating the transitions, the preceding state \( S_{\text{prec}} \) and the following state \( S_{\text{ foll}} \) in one transition activity are stored in a list \( C \) for further probability calculation.

The list \( Q_{\text{state}} \) is used to track the preceding state and is dynamically changing. For each of the following operational state, numbers \( V \) and \( R \) are the numbers of forced-down and resumed units during the transition. If the following states are balanced and having more than \( k + 1 \) units, they are placed back into \( Q_{\text{state}} \) and to be withdrawn as preceding states.

At the same time, the preceding state \( S_{\text{prec}} \), following state \( S_{\text{ foll}} \), index \( V \) and \( R \) are stored in a list \( C \) as a successful transition for the future use. The steps of Part I are:

---

**Part I: Obtain Operational States**

Set the initial state as a matrix of ones: \( S_{\text{ini}} = 1_{i 	imes n} \), and place it in \( Q_{\text{state}} \) as an element.

Set an empty list \( C \) to keep all the preceding and following states combinations.

while \( Q_{\text{state}} \) is not empty do

withdraw one element \( q \) from \( Q_{\text{state}} \), set \( S_{\text{prec}} = q \), delete \( q \) from \( Q_{\text{state}} \).

for \( (\text{row}=1, \text{row} \leq i, \text{row}++) \) do

for \( (\text{col}=1, \text{col} \leq n, \text{col}++) \) do

if \( S_{\text{prec}}(\text{row}, \text{col}) \neq 0 \) do

if \( \text{row}=1 \) and \( (\text{col} = \frac{n}{2} \) or \( \text{col} = n) \) do

\[ S_{\text{prec}}(:, \text{col}) = 0; \quad S_{\text{ foll}} = S_{\text{prec}}. \]

\[ W = \text{sum}(S_{\text{prec}} == 1) - (i - 1) \times \left( (S_{\text{prec}}(\text{row}, n) == 1) + (S_{\text{prec}}(\text{row}, n/2) == 1) \right) \]

\[ V = 0; \quad R = 0. \]

if \( W \geq k \) do

place \( [S_{\text{ foll}}, q, V, R] \) in \( C \) as a new element.

if \( W \geq k + 1 \) do

---


place \( S_{\text{fail}} \) in \( Q_{\text{state}} \).

end if

end if

end if

else

\( S_{\text{prec}}(\text{row, col}) = 0 \).

call function balance on \( S_{\text{prec}} \).

if function balance returns a balanced state \( S_{\text{fail}} \) do

\( W_{\text{diff}} = S_{\text{prec}} - S_{\text{fail}} \).

\( V = \text{sum}(W_{\text{diff}} = -2) \).

\( R = \text{sum}(W_{\text{diff}} = 2) \).

\( W = \text{sum}(S_{\text{prec}} = 1) - (i - 1) \times \left( \left( S_{\text{prec}}(\text{row, n}) = 1 \right) + \left( S_{\text{prec}}(\text{row, } n/2) = 1 \right) \right) \)

if \( W \geq k \) do

place \( [S_{\text{fail}}, q, V, R] \), in \( C \) as a new element.

if \( W_{\text{fail}} \geq k + 1 \) do

place \( S_{\text{fail}} \) in \( Q_{\text{state}} \).

end all if clauses

end all for loops

end while

In Part I, function \textbf{balance} is called to search for the “best” force-down positions when a new failure occurs on the \( g^{th} \) position of \( l^{th} \) plane. The state loses its balance and is represented as: \( S_{\text{prec}} \). This function returns the best operational state (if any). The steps of the function \textbf{balance} are:

\textbf{Function: balance()}

Place \( S_{\text{prec}} \) in an empty list \( Q \) as an element.

while \( Q \) is not empty do

withdraw one element \( q \) from \( Q \), set \( S_{\text{prec}} = q \), delete \( q \) from \( Q \).

Obtain the number of operating units for \( q \):

\( W = \text{sum}(S_{\text{prec}} = 1) - (i - 1) \times \left( \left( S_{\text{prec}}(1, n/2) = 1 \right) + \left( S_{\text{prec}}(1, n) = 1 \right) \right) \)

Surrogate state: \( S_{\text{sur}}(1:r,:) = S_{\text{prec}}(1:r,1:n/2); \)
\[ S_{\text{sur}} \left( r + 1: 2r, 1: \frac{n}{2} - 1 \right) = \text{flip} \left( S_{\text{prec}} \left( 1: r, \frac{n}{2} + 1: \frac{n}{2} - 1 \right) \right); \]

\[ S_{\text{sur}} \left( r + 1: 2r, \frac{n}{2} \right) = S_{\text{prec}} \left( 1: r, n \right); \]

\[ S_{\text{sur}} \left( S_{\text{sur}} = 3 \right) = 1. \]

\begin{verbatim}
for (row = 1, row \leq 2i, row++) do
    for (col = 1, col \leq \frac{n}{2}, col++) do
        if \( S_{\text{prec}} \left( \text{row}, \text{col} \right) = 0 \) do
            if \( \text{row} = 1 \) and \( \text{col} = \frac{n}{2} \) do
                \( S_{\text{sur}} \left( :, \text{col} \right) = 0. \)
            else if \( \text{row} \in \left[ 1, \frac{2i}{3} \right] \) do
                \( S_{\text{sur}} \left( \text{row} + \frac{2i}{3}, \text{col} \right) = 3; \)
                \( S_{\text{sur}} \left( \text{row} + \frac{4i}{3}, \text{col} \right) = 3. \)
            else if \( \text{row} \in \left[ \frac{2i}{3} + 1, \frac{4i}{3} \right] \) do
                \( S_{\text{sur}} \left( \text{row} - \frac{2i}{3}, \text{col} \right) = 3; \)
                \( S_{\text{sur}} \left( \text{row} + \frac{2i}{3}, \text{col} \right) = 3. \)
            else if \( \text{row} \in \left[ \frac{4i}{3} + 1, 2i \right] \) do
                \( S_{\text{sur}} \left( \text{row} - \frac{2i}{3}, \text{col} \right) = 3; \)
                \( S_{\text{sur}} \left( \text{row} - \frac{4i}{3}, \text{col} \right) = 3. \)
        end if
    end if
end for

\( S' = S_{\text{sur}}. \)

\( S' \left( S' = 3 \right) = 0. \)

\begin{verbatim}
for (l = 1, l \leq 2i, l++) do
    \( S' \left( l, .\right) = \text{flip} \left( S_{\text{sur}} \left( l, 1: \frac{n}{4} \right) + \text{flip} \left( S_{\text{sur}} \left( l, \frac{n}{4}: \frac{n}{2} - 1 \right) \right) \right). \)
end for

\( S' \left( :, 1 \right) = S' \left( :, 1 \right) / 2. \)

\( S' \left( S' = 2 \right) = 3. \)

\begin{verbatim}
for (j = 1, j \leq 2i - 1, j++) do
    \( d \left( j, .\right) = S' \left( i + 1, .\right) - S' \left( i, .\right). \)
end for

\( d \left( 2i, .\right) = S' \left( 1, .\right) - S' \left( 2i, .\right). \)
\end{verbatim}
\[\text{doubled}(2i:) = [S'; S'].\]

for \(( j = 1, j \leq 2i, j++ \) do

\[rd(\cdot,: j) = \text{flip}(\text{doubled}( j : j + 1, :)).\]
end for

for \(( j = 1, j \leq 2i, j++ \) do

\[\text{balanceaxis} = [\text{balanceaxis}; j].\]
end if
end for

if \(\text{count(balanceaxis)} \geq 3\) do

\(\text{return } S'\)

end if

if \(W' \geq k + 1\) do

place \(S'\) in \(Q\).
end if
end while

After obtaining all of the operational states, in Part II, the probability of occurrence for each operational state is obtained. The system reliability is then obtained by summing these probabilities. The steps are:

**Part II: Obtain Probability Expressions()**

Order the \(g\) elements in list \(C\) based on the number of failed units in the preceding states \(S_{\text{prec}}\).

for \(( l = 1, l \leq g, l++ \) do

withdraw \(l^{th}\) element \(C_l = [S_{\text{folll}}, S_{\text{precl}}, V', R']\) from \(C\).

if \(S_{\text{folll}} = S_{\text{precl}}\) do

\[p_{\text{folll}} = R(t)^{(n-2)y^2}.\]

place \(C_l = [S_{\text{folll}}, S_{\text{precl}}, V', R', P_{\text{folll}}]\) back into the original place in \(C\).

else

for \(( j = 1, j \leq l, j++ \) do

if \(S_{\text{precl}} = S_{\text{folll}}\) do

\[P_{\text{precl}}(t) = P_{\text{folll}}(t).\]

end if
end for

\[P_{\text{folll}} = \int_{t=0}^{t} P_{\text{precl}}(\tau) \times f(\tau) \times R(\tau)^{-b+y'+R'} R(t-\tau)^{R'} d\tau R(t)^{-b-(y'+R'+1)} .\]
b is the exponent of $R(t)$ in $P_{prec}(t)$.

```markdown
end if

place $C_i = [S_{foll}^i, S_{prev}^i, V^i, R^i, P_{foll}^i]$ in the original place of list C.
```

```markdown
end for

The reliability function for system can be obtained: $R(t) = \sum_{i=1}^{g} P_{foll}^i(t)$.
```

### 5.4 Numerical Examples

#### 5.4.1 Numerical Examples: Spherical k-n-i: G Balanced Systems with Forced-down Units as Failed

In this section, examples for reliability estimation of the spherical $k$-n-i: $G$ balanced system with different values of $k$, $n$ and $i$ are provided and compared. We focus on analyzing the system reliability as a function the ratio of $k$ over total number of units as well as a function of $k$.

**Scenario 1: Reliability estimation for systems as a function of ratio** $\frac{k}{i(n-2)+2}$

An example is provided to compare the system reliability with different ratios of $\frac{k}{i(n-2)+2}$ which are the ratios of the minimum number of required operating units to the total number of units in the system. For a spherical system with three planes ($i = 3$) and eight units ($n = 8$) on each plane, there are $i \times (n - 2) + 2 = 20$ units in total. Figure 5.14 considers the reliability of this system with three different $k$: 10, 15, 19. The ratios of $\frac{k}{i(n-2)+2}$ are 0.5, 0.75 and 0.95 respectively. It is found that systems with a higher ratio tend to have a lower system reliability comparing with systems with a lower ratio.
Using a fixed ratio of 0.5 we show the reliability plots for three system configurations: spherical 10-8-3: G balanced system, spherical 19-8-6: G balanced system and spherical 16-12-3: G balanced system in Figure 5.15. As observed, the system with a smaller number of planes and/or a smaller number of units on each plane has better performance in system reliability. The reason is that with a more complex configuration, the system is more likely to fail due to imbalance.
Scenario 2: Reliability estimation for systems with fixed $k$

Two examples are provided to illustrate the reliability for the spherical $k$-$n$-$i$: G balanced system when the minimum number of required operating units ($k$) is fixed. In the first example, the minimum number of required operating units is 10. The reliability plots for three systems are shown in Figure 5.16. It is shown that the reliability for spherical 10-8-3: G balanced system has the highest reliability at all times. It may appear illogical as it shows that the most redundant system has the lowest reliability in a $k$-out-of-$n$ setting. However, since the system of interest is a balanced $k$-out-of-$n$ system with a three-
dimensional configuration, redundancy brings the system more units as well as a higher probability of non-working states due to imbalance. Table 5.1 compares the number of good states in the spherical $k$-$n$-$i$: G balanced systems as well as in the corresponding $k$-out-of-$n$ systems when both $k$ and the total number of units are the same. For example, for the spherical 10-8-3: G balanced system, the number of units is 20 and the minimum number of operating units is 10. Thus, the corresponding $k$-out-of-$n$ system is a 10-out-of-20 system. For a $k$-out-of-$n$ system, the number of good states is obtained by summing the permutations of $k$ or more units working, as shown in Equation (5.15). The number of good states for the corresponding spherical $k$-$n$-$i$: G balanced systems is obtained by selecting the balanced states among all the good states in the $k$-out-of-$n$ system. After obtaining the number of good states for both systems, we calculate the ratio of the number of balanced states to the number of good states. It is obvious that for systems with more redundancy, the ratio drops dramatically. This explains why systems with more redundancy tend to have a lower reliability comparing with systems with less redundancy.

\[
\text{number of good states} = \sum_{k}^{n} \binom{n}{k} = \sum_{k}^{n} \frac{n!}{k!(n-k)!} \tag{5.15}
\]
Figure 5.16 Reliability Estimation of Spherical $k$-$n$-$i$: G Balanced Systems with Fixed $k$

Table 5.1 Comparison between $k$-out-of-$n$ and Spherical $k$-$n$-$i$: G Balanced Systems

<table>
<thead>
<tr>
<th></th>
<th>Spherical 10-8-3: G balanced system</th>
<th>Spherical 10-12-3: G balanced system</th>
<th>Spherical 10-8-6: G balanced system</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total number of units</td>
<td>20</td>
<td>32</td>
<td>38</td>
</tr>
<tr>
<td>Ratio of $k$ to total number of units</td>
<td>0.5</td>
<td>0.31</td>
<td>0.26</td>
</tr>
<tr>
<td>Number of good states in $k$-out-of-$n$ ($k=10$)</td>
<td>$6.2 \times 10^5$</td>
<td>$4.3 \times 10^9$</td>
<td>$2.7 \times 10^{11}$</td>
</tr>
<tr>
<td>Number of balanced states in spherical $k$-$n$-$i$: G balanced system</td>
<td>2721</td>
<td>$2.9 \times 10^5$</td>
<td>$4 \times 10^5$</td>
</tr>
<tr>
<td>Ratio of balanced states to good states</td>
<td>$4.4 \times 10^{-3}$</td>
<td>$6.9 \times 10^{-5}$</td>
<td>$1.5 \times 10^{-6}$</td>
</tr>
</tbody>
</table>

In the second example, the minimum number of units that are required to work is 18 ($k=18$).

As shown in Figure 5.17, since majority (90%) of the units in the spherical 18-8-3: G balanced system are required to work, the reliability (blue line) drops fast and is lower than
the other two systems’ reliability at first. As more units fail and cause the imbalance for the other two systems, the reliability plots for the spherical 18-8-3 and 18-8-6: G balanced systems intersect with each other.

![Reliability plots for spherical 18-8-3, 18-8-6, and 18-12-3 systems](image)

**Figure 5.17** Reliability Estimation of Spherical $k$-$n$-$i$: G Balanced Systems with Fixed $k$

**Table 5.2** Comparison between $k$-out-of-$n$ and Spherical $k$-$n$-$i$: G Balanced Systems

<table>
<thead>
<tr>
<th></th>
<th>Spherical 18-8-3: G balanced system</th>
<th>Spherical 18-12-3: G balanced system</th>
<th>Spherical 18-8-6: G balanced system</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total number of units</td>
<td>20</td>
<td>32</td>
<td>38</td>
</tr>
<tr>
<td>Ratio of $k$ to total number of units</td>
<td>0.9</td>
<td>0.56</td>
<td>0.47</td>
</tr>
<tr>
<td>Number of good states in $k$-out-of-$n$ ($k=18$)</td>
<td>211</td>
<td>$1.3 \times 10^9$</td>
<td>$1.9 \times 10^{11}$</td>
</tr>
<tr>
<td>Number of balanced states in spherical $k$-$n$-$i$: G balanced system</td>
<td>5</td>
<td>61565</td>
<td>85305</td>
</tr>
<tr>
<td>Ratio of balanced states to good states</td>
<td>$2.4 \times 10^{-2}$</td>
<td>$4.8 \times 10^{-5}$</td>
<td>$4.5 \times 10^{-7}$</td>
</tr>
</tbody>
</table>
5.4.2 Numerical Examples: Spherical k-n-i: G Balanced Systems with Forced-down Units as Standbys

In this section, examples of different spherical k-n-i: G balanced systems with forced-down units considered as standbys are illustrated. First, we compare the system reliability for the same spherical k-n-i: G balanced system configuration when forced-down units are considered as standbys and can be resumed when needed.

Scenario 1: Reliability Comparison of spherical k-n-i: G balanced system with forced-down units considered as standbys or as failed

Figure 5.18 shows the reliability comparison of a spherical 30-8-6: G balanced system. In this system, the units’ lifetimes are i.i.d. and follow an exponential distribution with mean equals to 500. From the figure, it is obvious that the reliability is higher for a spherical balanced system when forced-down units are resumed to work (blue line) comparing with forced-down units when they are considered as failed (orange line).
Figure 5.18 Reliability of Spherical 30-8-6: G Balanced Systems

**Scenario 2: Reliability comparison between the preceding state and the following state in a transition**

In this example, we show that in a transition, the system on the following state can have a higher reliability comparing with the system on the preceding state. In section 5.3.1, Figure 5.13 illustrates one of the transition paths for the spherical 30-8-9: G balanced system. We observe that when force-down units are considered as standbys, it is possible that in a transition, the following state has more operating units than the preceding state. For example, during the transition shown in Figure 5.19, nine units are resumed, and two units are forced down. The following state (Figure 5.19-b) has six more operating units
comparing with the preceding state (Figure 5.19-a). The system reliability plots for the two states are shown in Figure 5.20. The following state has a higher reliability than the preceding state at all the time. Thus, we conclude that when forced-down units are considered as standbys, it is possible that the following state provides a higher system reliability comparing with the preceding state as shown by the red line and blue lines respectively.

![Figure 5.19 A Transition of a Spherical 30-8-9: G Balanced System](image)
Scenario 3: Reliability comparison of spherical k-n-i: G balanced systems as a function of k

Based on the example from Scenario 1, we observe that the difference between the reliabilities for a system when forced-down units are considered as failed or standbys is large for the spherical 30-8-6: G balanced system. In this example, we show that with k increased, the difference between reliabilities of the same system (considering forced-down units as standbys or failed) is decreased. In Figure 5.21, the blue lines are reliability curves for spherical 26-8-6: G balanced system. The red lines are reliability curves for spherical
30-8-6: G balanced system. The solid lines are reliability curves when forced-down units are considered as standbys. The dashed lines are reliability curves when forced-down units are considered as failed. It is obvious that, the difference between the two blue curves (spherical 26-8-3: G balanced system) is much larger than the difference between the two red curves (spherical 30-8-3: G balanced system). The reason for this difference is that for system with increased $k$, the number of states with resuming activity is largely decreased. As shown in Table 5.3, the ratio of states with resuming activities over total number of states is 0.21 for spherical 26-8-6: G balanced system. However, this ratio is 0.007 for the spherical 30-8-6: G balanced system.

Figure 5.21 Reliability Comparison with Different $k$
**Table 5.3** Comparison of Number of States with Resuming Activities as a Function of \( k \)

<table>
<thead>
<tr>
<th></th>
<th>Spherical 26-8-6: G balanced system</th>
<th>Spherical 30-8-6: G balanced system</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total number of states (A)</td>
<td>109823</td>
<td>5267</td>
</tr>
<tr>
<td>Number of states with resuming activities (B)</td>
<td>22847</td>
<td>38</td>
</tr>
<tr>
<td>Ratio of B/A</td>
<td>0.21</td>
<td>0.007</td>
</tr>
</tbody>
</table>

**Scenario 4: Reliability comparison for spherical \( k\)-n-i: G balanced system as a function of \( n \) and \( i \)**

In this scenario, we show that with the ratio \( \frac{k}{i(n-2)+2} \) fixed, when \( n \) or \( i \) increase, the difference between reliabilities of the spherical \( k\)-n-i: G balanced system considering forced-down units as standbys or as failed is increases. In Figure 5.22, we first analyze the system reliability with increased number \( i \). The blue curves are reliabilities of spherical 26-8-6: G balanced system. The red curves are reliabilities of the spherical 16-8-3: G balanced system. The ratio \( \frac{k}{i(n-2)+2} \) for these two systems is around 0.8. We observe the differences between the two blue lines to be much larger than the red lines.
In Figure 5.23, we observe similar phenomenon that the difference in reliability for systems with a larger number $n$ is increased. The blue curves are reliabilities of the spherical 20-12-3: $G$ balanced system. The red curves are reliabilities of the spherical 16-8-3: $G$ balanced system. The difference between the two blue curves is much larger than the difference between the red curves. Table 5.4 shows the ratio of states with resuming activities over the total number of states for the spherical 16-8-3/26-8-6/20-12-3: $G$ balanced system. We observe that this ratio for the spherical 16-8-3: $G$ balanced system is much smaller than the ratios of the other two systems.
Figure 5.23 Reliability Comparison with Different $n$

Table 5.4 Comparison of Number of States

<table>
<thead>
<tr>
<th></th>
<th>Spherical 26-8-6: G balanced system</th>
<th>Spherical 16-8-3: G balanced system</th>
<th>Spherical 20-12-3: G balanced system</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total number of states (A)</td>
<td>109823</td>
<td>195</td>
<td>374025</td>
</tr>
<tr>
<td>Number of states with resuming activities (B)</td>
<td>22847</td>
<td>2</td>
<td>94078</td>
</tr>
<tr>
<td>Ratio of B/A</td>
<td>0.21</td>
<td>0.01</td>
<td>0.25</td>
</tr>
</tbody>
</table>
5.5 Conclusions

In this chapter, we first introduce the system configuration for the spherical $k$-$n$-$i$: $G$ balanced system and discuss the system balance in two aspects: rotational balance and symmetrical balance. For the system to operate properly, we require the system to maintain rotational balance (no rotational motion in roll, pitch and yaw axes). Meanwhile, to ensure the system stability, the operating units are required to be symmetrical with respect to three planes that are orthogonal to the $x$-$y$ plane. Mathematical models are presented to determine the balance of a state.

Then, we introduce the reliability estimation of the system considering two scenarios: 1) forced-down units are considered as failed and 2) forced-down units are considered as standbys. We first obtain all the operational states by enumerating the one-step transitions for each state. The reliability of the system is estimated by summing the probabilities of the operational states. In more complex system configurations, the transition enumeration becomes computational expensive and error prone. Thus, an algorithm and proper data structures are proposed to avoid the repetitive calculations.

Finally, reliability analysis and comparisons are provided using numerical examples. For the same spherical $k$-$n$-$i$: $G$ balanced system, a smaller $k$ always leads to a higher reliability. When the ratio of $k$ and the total number of units in the system is fixed, systems with less planes and/or units on each plane tend to have higher reliability comparing with systems with more complex configurations. For different systems with the same $k$, systems with more units do not necessarily provide a higher reliability. The reliability depends on two
ratios: 1) the ratio of $k$ to the total number of units in the system and 2) the ratio of balanced states to the total number of good states in a $k$-out-of-$n$ system. In the design of system configuration, it is preferable to have a low ratio of $k$ to the total number of units and a high ratio of balanced states comparing with the total number of good states in a $k$-out-of-$n$ system.

When forced-down units are considered as standbys, the system has a higher reliability, as expected. At the same time, due to resuming activity, the following state in a transition can provide a higher reliability for the system. We observe that the difference of system reliability when forced-down units are considered as standbys or failed depends on $k$, $n$, and $i$. Systems with a small $k$, a large $n$ and/or $i$ have more resuming activities and hence a larger reliability difference comparing with others.
CHAPTER 6

DEGRADATION MODELING OF

\((k_1,k_2)\)-OUT-OF-\((n,m)\) PAIRS: G BALANCED SYSTEMS

Chapters 3 and 4 present the reliability estimation for two types of \((k_1,k_2)\)-out-of-\((n,m)\) pairs: G balanced systems: 1) all units of the pairs perform the same functions and 2) units in adjacent pairs perform complementary functions. In case of failures, system balance is maintained by forcing down operating units and/or resuming previously forced-down units.

In practice, reliability modeling based on failure-time data requires a large sample size of failure data. As an alternative, degradation modeling captures units’ underlying failure process based on degradation growths analysis. When the accumulated degradation exceeds a certain threshold, the unit is considered failed. This approach requires a smaller sample size for reliability modeling. In this chapter, we develop a degradation model to estimate the reliability of \((k_1,k_2)\)-out-of-\((n,m)\) pairs: G balanced systems. For a balanced system with multiple units, when certain units degrade faster than others and reach the failure threshold, operating units are forced down to ensure the system balance. Thus, it is of great interest for all the units to maintain a similar degradation level so that the number of forced-down units due to imbalance is minimized. In section 6.2, we present two degradation models (stationary and non-stationary) for individual units. In section 6.3, we propose a strategy of ensuring balanced degradations of the units by forcing down the units with the largest amount of degradation at any time \(t\) and provide the reliability estimation for the \((k_1,k_2)\)-out-of-\((n,m)\) pairs: G balanced systems. Sections 6.4 and 6.5 provide
numerical examples and a case study based on battery capacity degradation in which we validate the proposed models and analyze system reliability under different scenarios.

6.1 Problem Definition and Assumptions

6.1.1 System Description

Similar to the system definition in Chapters 3 and 4, in this chapter, a balanced system with multi-level of units distributed circularly is generalized as \((k_i,k_2)\)-out-of-\((n,m)\) pairs: 

\[
G \text{ balanced system } s.
\]

In such systems, \(n\) pairs of units are evenly distributed in a circular arrangement in ‘\(m\)’ levels. On each side of a pair, there are \(m\) units arranged vertically and in parallel with the opposite side of the pair as shown in Figure 3.5. There are \(2m\) units for each pair and a total of \(2nm\) units for the entire system. For each pair to be functioning, at least \(k_2\) out of \(m\) units need to operate properly on both sides of the pair. Otherwise, the pair is considered failed and the remaining functioning units of the same pair are forced down. The system is properly functioning when at least \(k_1\) out of the \(n\) pairs work properly.

We analyze the system reliability for two types of systems. In Type I system, all units perform the same function. In Type II system, any horizontally adjacent pairs must perform complementary functions. This means, when a pair fails, one of the adjacent operating pairs that perform the complementary function is forced down.

Degradation modeling is generally explored in three processes: Gamma process, Wiener process and inverse Gaussian (IG) process. Among them, the Gamma process and the IG process are well known for their monotonicity. As a limiting compound Poisson process, the IG process is more suitable in degradation modeling in applications such as fatigue
crack growth and corrosion. In this chapter, we assume that the degradation paths of individual units in the \((k_1, k_2)\)-out-of-\((n, m)\) pairs: G balanced system follow the IG stochastic process. It is a stationary process which means the mean and variance do not change with time. Additionally, we propose an improved IG process in which the degradation at any time \(t\) is dependent on the amount of the degradation at time \(t - \Delta t\) and is therefore a non-stationary process.

6.1.2 Balance Requirement

In these \((k_1, k_2)\)-out-of-\((n, m)\) pairs: G balanced systems, assuming that all units, regardless of their functions, follow the same degradation model. A unit fails when its degradation value exceeds a predetermined threshold. Each operating unit is assigned a weight of 1 and a failed unit is assigned a weight of 0 regardless of their degradation value. Similar to Chapters 3 and 4, a system is balanced when its center of gravity is in the center of the circle and unbalanced system is considered failed. This balance requirement applies to both Type I and Type II systems.

In this \(n\)-pair-\(m\)-level system, if one unit’s degradation value exceeds the threshold, the unit fails and the entire system loses its balance. Thus, when one unit fails, one of the opposite units with the largest degradation value is forced down immediately to maintain the balance of the system. Meanwhile, for the Type II system when any adjacent operating pairs performing complementary functions, if one pair of units is not operational (failed or forced down), the adjacent pair with the largest number of failed units is forced down so that any
two adjacent units perform complementary functions. Forced-down units are considered as standbys and can resume their operations when needed.

### 6.2 Degradation Modeling of Individual Units

For any individual unit in a $(k_1,k_2)$-out-of-$(n,m)$ pairs: G balanced system, two degradation scenarios are discussed: 1) the degradation processes for individual units are stationary (discussed in section 6.2.1) and 2) the degradation processes for individual units are non-stationary (discussed in section 6.2.2). The degradation paths follow IG stochastic processes. When a unit’s degradation path exceeds a threshold $d_s$, the unit is considered failed. A unit’s state, *i.e.* operating, standby and failure, depends on its degradation amount as well as the states of other units in the system. This is because a unit can be forced down due to a failure in the same pair or a failure of its adjacent pair (Type II system). When units/pairs are forced down, their degradation values remain unchanged during the standby period.

#### 6.2.1 Stationary Degradation Process

In $(k_1,k_2)$-out-of-$(n,m)$ pairs: G balanced systems, we suppose that a unit $i$’s degradation process $\{r_i(t), \ t \geq 0\}$ follows an IG process with a scale parameter $\lambda$ and a shape function $\Lambda(t)$. The degradation process has the following properties:
1. $r_i(t)$ has independent increments: degradation increments

$$\Delta r_i(t_j) = r_i(t_j) - r_i(t_{j-1})$$
and $\Delta r_i(t) = r_i(t_j) - r_i(t_{j-1})$ are independent with each other for $\forall j \neq l$.

2. Degradation increments follow an inverse Gaussian distribution:

$$\Delta r_i(t_j) \sim IG \left( \Lambda(t_j) - \Lambda(t_{j-1}), \lambda \left( \Lambda(t_j) - \Lambda(t_{j-1}) \right)^2 \right).$$

Usually, the shape function is assumed to be a function of time: $\Lambda(t) = \mu t$. The probability density function (pdf) and the cumulative density function (CDF) of the degradation increment $\Delta r_i(t_j) = r_i(t_j) - r_i(t_{j-1})$ are given respectively by Equations (6.1) and (6.2).

$$f \left( (r_i(t_j) - r_i(t_{j-1})) \mid \Lambda(t_j) - \Lambda(t_{j-1}), \lambda \left( \Lambda(t_j) - \Lambda(t_{j-1}) \right)^2 \right)$$

$$= \frac{\lambda \left( \Lambda(t_j) - \Lambda(t_{j-1}) \right)^2}{2\pi \left( r_i(t_j) - r_i(t_{j-1}) \right)^3} \exp \left[ -\frac{\lambda \left( \left( r_i(t_j) - r_i(t_{j-1}) \right) - \left( \Lambda(t_j) - \Lambda(t_{j-1}) \right) \right)^2}{2 \left( r_i(t_j) - r_i(t_{j-1}) \right)} \right]$$

$$= \frac{\lambda \mu^2}{2\pi \left( r_i(t_j) - r_i(t_{j-1}) \right)^3} \exp \left[ -\frac{\lambda \left( \left( r_i(t_j) - r_i(t_{j-1}) \right) - \mu \right)^2}{2 \left( r_i(t_j) - r_i(t_{j-1}) \right)} \right]$$

$$= \Phi \left[ \frac{\lambda \mu^2}{\sqrt{r_i(t_j) - r_i(t_{j-1})}} \left( \frac{r_i(t_j) - r_i(t_{j-1})}{\mu} - 1 \right) \right]$$

$$+ e^{2\lambda \mu} \Phi \left[ -\frac{\lambda \mu^2}{\sqrt{r_i(t_j) - r_i(t_{j-1})}} \left( \frac{r_i(t_j) - r_i(t_{j-1})}{\mu} + 1 \right) \right]$$

(6.1)

The expectation and variance of the degradation process are given in Equations (6.3) and (6.4) respectively:
Because the IG process is monotonic, assuming the degradation threshold is \( d_h \), the CDF of unit \( i \)'s failure time (first passage time) \( T_i \) is obtained by Equation (6.5).

\[
P(T_i < t) = P(r_i(t) > d_h) = 1 - F(d_h; \Lambda(t), \lambda \Lambda(t)^2) \\
= \Phi \left[ \frac{\lambda}{d_h} (\Lambda(t) - d_h) \right] - e^{2\lambda \Lambda(t)} \Phi \left[ -\frac{\lambda}{d_h} (\Lambda(t) + d_h) \right] \\
= \Phi \left[ \frac{\lambda}{d_h} (\mu t - d_h) \right] - e^{2\lambda \mu t} \Phi \left[ -\frac{\lambda}{d_h} (\mu t + d_h) \right]
\]  

(6.5)

Suppose that at time \( t_j \), the unit \( i \)'s degradation value is \( r_i(t_j) \). The probability that the unit fails in the next time increment given it is properly working at time \( t_j \) is given by Equation (6.6).

\[
P(t_j < T_i < t_{j+1}) = P(r_i(t_j) + \Delta r_i(t_{j+1}) > d_h | r_i(t_j) < d_h) \\
= P(\Delta r_i(t_{j+1}) > d_h - r_i(t_j)) = 1 - F(d_h - r_i(t_j)) \\
= \Phi \left[ \frac{\lambda}{d_h - r_i(t_j)} (\mu - d_h + r_i(t_j)) \right] - e^{2\lambda \mu t} \Phi \left[ \frac{\lambda}{d_h - r_i(t_j)} (r_i(t_j) - \mu - d_h) \right]
\]  

(6.6)

Note that, the degradation process of unit \( i \) is observed at every discrete unit of time. The point estimation of parameters \( \lambda \) and \( \mu \) can be obtained based on these observations. Let \( r_i(t) = [r_i(t_1), r_i(t_2), ... , r_i(t_n)] \) be the observed degradation values for unit \( i \) during time
The initial degradation is assumed to be zero: \( r_i(t_0) = 0 \). Then, the degradation increments during time \([t_0, t_n]\) is obtained by
\[
\left[ (r_i(t_1) - r_i(t_0)), (r_i(t_2) - r_i(t_1)), \ldots, (r_i(t_n) - r_i(t_{n-1})) \right].
\]
The log-likelihood function is given by Equation (6.7):
\[
\ln l(\mu, \lambda) = \sum_{j=1}^{n} \ln f (r_i(t_j) - r_i(t_{j-1}))
\]
\[
= \sum_{j=1}^{n} \ln \left( \frac{\lambda \mu^2}{2\pi (r_i(t_j) - r_i(t_{j-1}))^3} \exp \left[ \frac{-\frac{\lambda^2 ((r_i(t_j) - r_i(t_{j-1})) + \mu)}{2(r_i(t_j) - r_i(t_{j-1}))} \right] \right)
\]
(6.7)
\[
= \sum_{j=1}^{n} \left( \frac{1}{2} \ln \frac{\lambda \mu^2}{2\pi (r_i(t_j) - r_i(t_{j-1}))^3} - \frac{\lambda (r_i(t_j) - r_i(t_{j-1}) + \mu)^2}{2(r_i(t_j) - r_i(t_{j-1}))} \right)
\]
The MLE estimators are obtained by solving the following equations numerically:
\[
\frac{\partial \ln l}{\partial \mu} = 0, \quad \frac{\partial \ln l}{\partial \lambda} = 0
\]

6.2.2 Non-stationary Degradation Process

In section 6.2.1, we present the degradation modeling for individual units in the \((k_1, k_2)\)-out-of-\((n, m)\) pairs: G balanced system based on the IG process. In this model, the degradation amount is a function of time and is predictable at any time. However, in many cases, degradation is also affected by immediately preceding degradation state. For example, the degradation growth of fatigue crack depends not only on material and environmental factors, but also on the previous crack length. A large initial crack length usually leads to a large future crack growth. Thus, in this section, we present an IG based...
non-stationary degradation process for the individual unit: improved inverse Gaussian (IIG) process (Guo et al. 2018). In the IIG degradation process, the degradation increments for an individual unit are functions of the unit’s preceding degradation status.

Suppose that unit i’s degradation process \( \{ r_i(t), \ t \geq 0 \} \) is observed at discrete units of time. The degradation increment \( \Delta r_i(t_j) = r_i(t_j) - r_i(t_{j-1}) \) during time \([t_{j-1}, t_j]\) is a function of the preceding degradation state \( r_i(t_{j-1}) \) and follows an inverse Gaussian distribution with a scale parameter \( \lambda \) and a shape function \( \Lambda(r_i(t_j)) = \mu_0(r_i(t_{j-1})) \). The pdf and CDF of \( \Delta r_i(t_j) \) is given by Equations (6.8) and (6.9) (Guo et al. 2018).

\[
\begin{align*}
  f\left( \Delta r_i(t_j) \middle| \Lambda(r_i(t_j)) - \Lambda(r_i(t_{j-1})), \lambda \left( \Lambda(r_i(t_j)) - \Lambda(r_i(t_{j-1})) \right) \right) &= \sqrt{\frac{\lambda \left( \Lambda(r_i(t_j)) - \Lambda(r_i(t_{j-1})) \right)}{2\pi \Delta r_i^3(t_j)}} \\
  &= \frac{\lambda \left( \Lambda(r_i(t_j)) - \Lambda(r_i(t_{j-1})) \right)}{2\pi \Delta r_i^3(t_j)} \exp \left[ -\frac{\lambda \left( \Delta r_i(t_j) - \left( \Lambda(r_i(t_j)) - \Lambda(r_i(t_{j-1})) \right) \right)^2}{2 \Delta r_i(t_j)} \right] \\
  &= \sqrt{\frac{\lambda \left( \mu_0 \Delta r_i(t_{j-1}) \right)^2}{2\pi \Delta r_i^3(t_j)}} \exp \left[ -\frac{\lambda \left( \Delta r_i(t_j) - \mu_0 \Delta r_i(t_{j-1}) \right)^2}{2 \Delta r_i(t_j)} \right] \tag{6.8}
\end{align*}
\]
\[ F \left( \Delta r_i(t_j) \left| \Lambda \left( r_i(t_j) \right) - \Lambda \left( r_i(t_{j-1}) \right) \right) \right) = \Phi \left[ \frac{\lambda \left( \Lambda \left( r_i(t_j) \right) - \Lambda \left( r_i(t_{j-1}) \right) \right)^2}{\Delta r_i(t_j)} \left( \frac{\Delta r_i(t_j)}{\Lambda \left( r_i(t_j) \right) - \Lambda \left( r_i(t_{j-1}) \right)} - 1 \right) \right] \]

\[ + e^{2 \lambda_i \Delta r_i(t_{j-1})} \Phi \left[ \sqrt{\frac{\lambda \left( \Lambda \left( r_i(t_j) \right) - \Lambda \left( r_i(t_{j-1}) \right) \right)^2}{\Delta r_i(t_j)}} \left( \frac{-\Delta r_i(t_j)}{\Lambda \left( r_i(t_j) \right) - \Lambda \left( r_i(t_{j-1}) \right)} - 1 \right) \right] \]

\[ = \Phi \left[ \frac{\lambda \left( \mu_0 \Delta r_i(t_{j-1}) \right)^2}{\Delta r_i(t_j)} \left( \frac{\Delta r_i(t_j)}{\Delta r_i(t_{j-1})} - 1 \right) \right] \]

\[ + e^{2 \lambda_i \Delta r_i(t_{j-1})} \Phi \left[ \sqrt{\frac{\lambda \left( \Lambda \left( r_i(t_j) \right) - \Lambda \left( r_i(t_{j-1}) \right) \right)^2}{\Delta r_i(t_j)}} \left( \frac{-\Delta r_i(t_j)}{\Lambda \left( r_i(t_j) \right) - \Lambda \left( r_i(t_{j-1}) \right)} - 1 \right) \right] \quad (6.9) \]

The expectation and variance are given respectively by Equations (6.10) and (6.11):

\[ E \left( \Delta r_i(t_j) \right) = \Lambda \left( r_i(t_j) \right) - \Lambda \left( r_i(t_{j-1}) \right) = \mu_0 \Delta r_i(t_{j-1}) \quad (6.10) \]

\[ Var \left( r_i(t_j) \right) = \frac{\mu_0 \Delta r_i(t_{j-1})}{\lambda} \quad (6.11) \]

Suppose that the degradation threshold is \( d_h \). The probability that unit \( i \) fails in the next time increment given it works at time \( t_j \) with the degradation value \( r_i(t_j) \) is given by Equation (6.12).

\[ P \left( t_j < T < t_{j+1} \right) = P \left( r_i(t_j) + \Delta r_i(t_j) > d_h \mid r_i(t_j) < d_h \right) = P \left( \Delta r_i(t_j) > d_h - r_i(t_j) \right) \]

\[ = 1 - F \left( d_h - r_i(t_j) \right) = \Phi \left[ \sqrt{\frac{\lambda}{d_h - r_i(t_j)}} \left( \mu_0 \Delta r_i(t_j) - d_h + r_i(t_j) \right) \right] \]

\[ - e^{2 \lambda_i \Delta r_i(t_j)} \Phi \left[ \sqrt{\frac{\lambda}{d_h - r_i(t_j)}} \left( r_i(t_j) - \mu_0 \Delta r_i(t_j) - d_h \right) \right] \quad (6.12) \]
Similar to the IG process, the IIG process is also monotonic. We estimate the lifetime of unit \( i \) by iterative calculations. Suppose that the degradation process is observed until time \( t_n \) and the degradation at that time is \( r_i(t_n) \), the lifetime can be obtained by the following procedure:

1. Starting from \( j = 1 \).
2. Generate a random number \( \Delta r_i(t_{n+j}) = r_i(t_{n+j}) - r_i(t_{n+j-1}) \) based on the inverse Gaussian distribution \( IG\left(\Lambda(r_i(t_{n+j})), \lambda \Lambda^2(r_i(t_{n+j}))\right) \).
3. Obtain the degradation value at time \( t_{n+j} : r_i(t_{n+j}) = r_i(t_{n+j-1}) + \Delta r_i(t_{n+j}) \).
4. Compare \( r_i(t_{n+j}) \) to the threshold \( d_h \). If \( r_i(t_{n+j}) < d_h \), set \( i \) to \( i + 1 \) and proceed to step 2. Else if \( r_i(t_{n+j}) \geq d_h \), the predicted failure time is \( t_{n+j} \).
5. Repeat steps 1-4 \( N \) times, the lifetime of unit \( i \) can be obtained by taking the average of the \( N \) predicted lifetimes.

In this proposed model, the parameters include \( \{\mu_0, \lambda\} \). Similarly, we estimate these parameters using MLE. Let \( r_i(t) = \begin{bmatrix} r_i(t_1), r_i(t_2), \ldots, r_i(t_n) \end{bmatrix} \) be the observed degradation values of unit \( i \) until time \( t_n \). Suppose that the initial degradation is \( r_i(t_0) = 0 \). The degradation increments during time \( (t_0, t_n) \) is represented by the vector \( \begin{bmatrix} (r_i(t_1) - r_i(t_0)), (r_i(t_2) - r_i(t_1)), \ldots, (r_i(t_n) - r_i(t_{n-1})) \end{bmatrix} \). The log-likelihood function is given by Equation (6.13):
\[ \ln l(\mu, \lambda) = \sum_{j=2}^{n} \ln f(r_i(t_j) - r_i(t_{j-1})) \]
\[ = \sum_{j=2}^{n} \ln \left( \frac{\lambda \left( \mu_0 \Delta r_i(t_{j-1}) \right)^2}{2\pi \left( r_i(t_j) - r_i(t_{j-1}) \right)^3} \exp \left[ -\frac{\lambda \left( \Delta r_i(t_j) - \mu_0 \Delta r_i(t_{j-1}) \right)^2}{2 \left( r_i(t_j) - r_i(t_{j-1}) \right)} \right] \right) \]
\[ = \sum_{j=2}^{n} \left( \frac{1}{2} \ln \frac{\lambda \left( \mu_0 \Delta r_i(t_{j-1}) \right)^2}{2\pi \left( r_i(t_j) - r_i(t_{j-1}) \right)^3} - \frac{\lambda \left( \Delta r_i(t_j) - \mu_0 \Delta r_i(t_{j-1}) \right)^2}{2 \left( r_i(t_j) - r_i(t_{j-1}) \right)} \right) \]

The MLE estimators are obtained by solving the following equations numerically:
\[ \frac{\partial \ln l}{\partial \mu_0} = 0, \quad \frac{\partial \ln l}{\partial \lambda} = 0 \]

6.3 Degradation Modeling of \((k_1, k_2)\)-out-of-(\(n, m\)) Pairs: G Balanced Systems

In a \((k_1, k_2)\)-out-of-(\(n, m\)) pairs: G balanced system, units’ degradation processes are random processes depending on random events and environmental factors. In this section, we assume that all units’ degradation paths follow a stationary process. More specifically, the units’ degradation processes \(\{r_i(t), \quad t \geq 0\}; \quad i = 1, 2, \ldots, 2nm\) follow the same degradation model (inverse Gaussian or improved inverse Gaussian). During the degradation, when certain units degrade faster than others and reach the degradation threshold first, to maintain the system balance, operating units on other locations are forced down even though their degradation values have not reached the threshold. Thus, it is preferable to have all units’ degradation values maintained at a similar level to minimize the number of forced-down units due to imbalance.
In this chapter, we present a degradation balance mechanism by dynamically forcing down and resuming the “most” degraded units while maintaining the minimum number of required working units and system balance. Suppose that all units’ degradation paths in the \((k_1, k_2)\)-out-of-\((n,m)\) pairs: G balanced system are observed at every discrete time interval and the degradation threshold is predetermined as \(d_h\). Any unit with a degradation value larger or equal to \(d_h\) is considered as failed. In addition, we introduce \(d_c\) as the threshold of difference between the most and the least degraded units in the system at any time. If the degradation difference between the most and the least degraded units in the system is larger or equal to \(d_c\), then the most degraded unit is forced down for the next unit of time.

To maintain the system balance, one needs to force down one of the operating units on the opposite side of the same pair or resume one of the previously forced-down units on the same side whenever is possible. For Type II systems, further actions such as forcing down or resuming pairs are required. In the next observation, if this previously forced-down unit is no longer the most degraded unit and/or the difference between the most and least degraded unit does not exceed threshold \(d_c\), this unit resumes operation. If the difference between the most and least degraded units is smaller than \(d_c\) in one observation, no forcing-downs are needed. During the entire degradation process, units with largest degradations are dynamically forced-down so that all units’ degradation values are at a similar level and the number of forced-down units due to imbalance is minimized.

For example, Figure 6.1 shows the degradation process of a pair in a Type I (2, 2)-out-of-(4, 3) pairs: G balanced system. In this pair, there are three units stacked vertically on both
sides. At least two units are required to be operating on both sides for a pair to be considered as operating properly. In the Figure, circles represent the units in the pair. Circles with white/black/grey color represent operating/failed/forced-down units respectively. The numbers next to the circles are degradation values at that observation time. The failure threshold is $d_h = 1$. The threshold $d_c$ is 0.3.

In Column A of Figure 6.1, the degradation balance mechanism is not applied which means the most degraded units are not forced down when the difference between this unit and the least degraded unit is larger than $d_c$. At time $t_0$, all the six units are operating with degradation values equal to zero. At time $t_1$, unit four fails because the degradation value reaches the threshold $d_h$. Unit one from the opposite side is forced down to balance the system simultaneously. At time $t_2$, unit five fails and unit one is resumed. At time $t_3$, unit one fails and the pair fails since there is only one operating units on the left side which is less than $k_2$ requirement for system’s operation. Other operating units (units two, three and six) in this pair are forced down while their degradation values have not reached the failure threshold.

In Column B, the most degraded units are forced down once the difference in degradation is larger than $d_c$. At time $t_0$, all of the six units are operating with degradation values equal to zero. At time $t_1$, unit four is forced down since the difference between its degradation value and the least degraded unit’s value is equal to the threshold $d_c$. To maintain the
system balance, unit one is forced down as it has the largest degradation value among the units on the left side of the pair. At time $t_2$, unit five is the most degraded unit and the difference between its value and the least degraded unit is larger than the threshold $d_c$. Thus, unit five is forced down. Unit one is resumed to operate to maintain the system balance. At time $t_3$, the difference between the most and least degraded units are less than the threshold $d_c$, units four and five are resumed. The degradation process shown in Column B is preferred compared with the process shown in Column A. This is because when the degradation process is continuously monitored, and most degraded units are forced down and resumed “back” to operation based on their degradation conditions. Hence, the degradation levels among the units are approximately close to each other and the likelihood of system’s unbalance is decreased. In sections 6.3.1 and 6.3.2, we discuss the reliability estimation when the degradation balance mechanism is applied or not.
6.3.1 Lower Bound Reliability Based on Transition Enumeration

In this section, we investigate the reliability for the \((k_1, k_2)\)-out-of\((n, m)\) pairs: G balanced system when units with large degradation values are not forced down during observations. Under this scenario, a closed-form expression can be obtained by enumerating the transitions among operational states and calculating the probabilities for these operational states. This is considered as a lower bound reliability for such systems.

Suppose that the degradation paths of the \(2nm\) units in the \((k_1, k_2)\)-out-of\((n, m)\) pairs: G system are observed until time \(T\). At time \(T\), the system can be on any operational state \(S_T\).
Similar to Chapter 3, starting from state $S_1$, we first enumerate all the following operational states $S_j$, $j = 1, 2, \ldots$. The system reliability at any time $t \ (> T)$ is obtained by summing the probabilities of occurrence for these operational states. The operational states enumeration is as same as in Chapter 3 and is not discussed in this chapter. However, the probability calculation for each operational state is different from Chapter 3. Because in this chapter, units’ failure time predictions are based on the degradation growths instead of failure time distributions.

**Scenario 1: Units’ Degradation Paths Follow Inverse Gaussian Process**

We assume that all units’ degradation paths $\{r_i(t), t \geq 0\}; \ i = 1, 2, \ldots nm$ follow the same inverse Gaussian process with a scale parameter $\lambda$ and a shape function $\Lambda(t) = \mu t$. The CDF of unit $i$’s degradation increments $F(\Delta r_i(t))$ is given in Equation (6.2). Suppose that at time $T$, unit $i$ with a degradation value of $r_i(T)$ is operating ($r_i(T) \leq d_h$). Unit $i$’s first passage time (which is unit $i$’s lifetime) is $t_i$. The CDF of unit $i$’s lifetime $F_i(t); t > T, \ i = 1, 2, \ldots nm$ is obtained by Equation (6.14).
\[
P(t_i < t \mid r_i(T) \leq d_h) = P(r_i(t) - r_i(T) > d_h - r_i(T)) = P(r_i(t-T) > d_h - r_i(T))
\]
\[
= 1 - P(r_i(t-T) \leq d_h - r_i(T)) = 1 - F(d_h - r_i(T); \Lambda(t-T), \lambda \Lambda(t-T)^2)
\]
\[
= \Phi \left[ \frac{\lambda}{\sqrt{d_h - r_i(T)}} (\Lambda(t-T) - d_h + r_i(T)) \right] - e^{2\lambda(t-T)} \Phi \left[ - \frac{\lambda}{\sqrt{d_h - r_i(T)}} (\Lambda(t-T) + d_h - r_i(T)) \right]
\]
\[
= \Phi \left[ \frac{\lambda}{\sqrt{d_h - r_i(T)}} (\mu(t-T) - d_h + \mu T) \right] - e^{2\lambda(t-T)} \Phi \left[ - \frac{\lambda}{\sqrt{d_h - r_i(T)}} (\mu(t-T) + d_h - \mu T) \right]
\]

(6.14)

Using \( F_i(t) \), the pdf \( f_i(t) \) and the reliability function \( R_i(t) \) are immediately obtained.

**Scenario 2: Units’ Degradation Paths Follow Improved Inverse Gaussian Process**

We assume that all units’ degradation paths \( \{r_i(t), t \geq 0\}; i = 1, 2, \ldots, 2nm \) follow the same improved inverse Gaussian process with a scale parameter \( \lambda \) and a shape function \( \Lambda(t) = \mu_0 (r_i(t-1)) \). The CDF of unit \( i \)’s degradation increments during \( (t-1, t] \) is \( F(\Delta r_i(t)) \) (shown in Equation (6.9)). Suppose that the unit \( i \)’s degradation process is observed until \( T \). At time \( T \), unit \( i \) with a degradation value of \( r_i(T) \) is operating properly \( (r_i(T) \leq d_h) \). Unit \( i \)’s first passage time (which is unit \( i \)’s lifetime) is \( t_i, t_i > T \). As discussed in section 6.2.2, the improved inverse Gaussian process is not a stationary process. Degradation growths in each time interval is a function of the previous degradation states. The CDF of unit \( i \)’s lifetime distribution \( F_i(t); t > T, i = 1, 2, \ldots, 2nm \) is obtained by Equation (6.15).
\[ F_i(t) = P \left( t < t r (1), r_i (2), \ldots, r_i (T) \right) = P \left( r_i (t) - r_i (T) > d_h - r_t (T) \right) \]
\[ = 1 - P \left( r_i (t) - r_i (T) \leq d_h - r_t (T) \right) \]
\[ = 1 - P \left( r_i (t) - r_i (t - 1) \leq d_h - r_t (t - 1) \right) \]
\[ = 1 - F \left( d_h - r_t (t - 1); \Lambda \left( r_i (t) \right) - \Lambda \left( r_i (t - 1) \right), \lambda \left( \Lambda \left( r_i (t) \right) - \Lambda \left( r_i (t - 1) \right) \right) \right) \]
\[ = 1 - F \left( d_h - r_t (t - 1); \mu_0 \Delta r_t (t - 1), \lambda \left( \mu_0 \Delta r_t (t - 1) \right) \right) \]
\[ = \Phi \left[ \frac{\lambda}{\sqrt{d_h - r_t (t - 1)}} \left( \mu_0 \Delta r_t (t - 1) - d_h + r_t (t - 1) \right) \right] - e^{2 \mu_0 \Delta r_t (t - 1)} \Phi \left[ \frac{\lambda}{\sqrt{d_h - r_t (t - 1)}} \left( \mu_0 \Delta r_t (t - 1) + d_h - r_t (t - 1) \right) \right] \]  
(6.15)

In this equation, the probability of unit \( i \) failing before time \( t \) equals the probability of unit \( i \)’s degradation at time \( t \) being larger than \( d_h \). This probability is the CDF of unit \( i \)’s lifetime distribution and is dependent on the previous degradation increment and state at time \( t - 1 \): \( \Delta r_t (t - 1) \) and \( r_t (t - 1) \). These two values are unknown and estimated by Equations (6.16) and (6.17).

\[ E \left( \Delta r_i (t - 1) \right) = \Lambda \left( r_i (t - 1) \right) - \Lambda \left( r_i (t - 2) \right) = \mu_0 \Delta r_i (t - 2) \]
\[ = \mu_0 \lambda^{-1-T} \Delta r_i (T) = \mu_0 \lambda^{-1-T} \left( r_i (T) - r_i (T - 1) \right) \]  
(6.16)

\[ E \left( r_i (t - 1) \right) = r_i (T) + \sum_{j=1}^{t-1} E \left( \Delta r_i (T + j) \right) = r_i (T) + \sum_{j=1}^{t-1} \mu_0 \lambda^j \left( r_i (T) - r_i (T - 1) \right) \]  
(6.17)

Using \( F_i(t) \), the pdf \( f_i(t) \) and the reliability function \( R_i(t) \) are immediately obtained. So far, we obtained the lifetime distribution functions for units for the inverse Gaussian degradation model and the improved inverse Gaussian degradation model. The next step is
to calculate the probability of occurrence for each operational state. During each transition, there is a previous state $S_{\text{prev}}$ and a following state $S_{\text{foll}}$. Suppose that the probability of occurrence for those two states are: $P_{\text{prev}}(t)$ and $P_{\text{foll}}(t)$. The probability of occurrence for the following state $P_{foll}(t)$ can be calculated iteratively based on the $P_{\text{prev}}(t)$ and the transition activity between these two states. We consider the following five possible transition activities:

Case 1: Unit $i$ fails at time $\tau$, $\tau > T$. One of the opposite operating units (unit $i'$) in the same pair is forced down at the same time. The $P_{\text{foll}}(t)$ is obtained by Equation (6.18).

$$P_{\text{foll}}(t) = \int_{\tau}^{t} f_i(\tau) R_{\text{prev}}(\tau) \prod_{j \in \{i',i\}} R_j^{-1}(t-T) d\tau$$  \hspace{1cm} (6.18)

Case 2: Unit $i$ fails at time $\tau$, $\tau > T$. One of the operating units (unit $i'$) on the same side of the same pair is resumed at the same time. The $P_{\text{foll}}(t)$ is obtained by Equation (6.19).

$$P_{\text{foll}}(t) = \int_{\tau}^{t} f_i(\tau) R_{\text{prev}}(t-\tau) \prod_{j \in \{i,i'\}} R_j^{-1}(t-T) d\tau$$  \hspace{1cm} (6.19)

Case 3: Unit $i$ in pair $j$ fails at time $\tau$, $\tau > T$. The number of failed units in at least one side of this pair is larger than $m - k_2 + 1$, which causes the failure of the entire pair. All the operating units (represented by $j^-$) in pair $j$ is forced down at time $\tau$. The $P_{\text{foll}}(t)$ is obtained in Equation (6.20).

$$P_{\text{foll}}(t) = \int_{\tau}^{t} f_i(\tau) \prod_{j \in \{j^-\}} R_j(\tau) P_{\text{prev}}(t) \prod_{k \in \{i,j^-\}} R_k^{-1}(t-T) d\tau$$  \hspace{1cm} (6.20)
Case 4: For Type II systems, unit $i$ in pair $j$ fails at time $\tau$, $\tau > T$. The number of failed units in at least one side of pair $j$ is larger than $m - k_2 + 1$, which causes the failure of the entire pair. All the units in pair $j$ and one of adjacent pair (pair $j'$) are forced down. The $P_{\text{fail}}(t)$ is obtained by Equation (6.21).

$$P_{\text{fail}}(t) = \int_T^t f_i(\tau) \prod_{k \in \{j', j''\}} R_k(\tau) P_{\text{prev}}(t) \prod_{h \in \{j', j''\}} R_h^{-1}(t-T)d\tau$$ (6.21)

Case 5: For Type II systems, unit $i$ in pair $j$ fails at time $\tau$, $\tau > T$. The number of failed units reaches exceeds the maximum number that causes the system’s failure and the pair $j$ is considered failed. One of previously forced-down pair (pair $j'$) is resumed. The $P_{\text{fail}}(t)$ is obtained by Equation (6.22).

$$P_{\text{fail}}(t) = \int_T^t f_i(\tau) \prod_{k \in \{j', j''\}} R_k(\tau) \prod_{h \in \{j', j''\}} R_h(t-\tau) P_{\text{prev}}(t) \prod_{h \in \{j', j''\}} R_h^{-1}(t-T)d\tau$$ (6.22)

The above cases include all the possible transition activities during operational states enumeration. Thus, the probability of occurrence for each operational state can be obtained by iterative calculations based on its transition history. The probability of the initial state $S_T$ is obtained by multiplying the reliability functions of all the working units (represented by $sys^+$) at time $T$ as shown in Equation (6.23). The system reliability is obtained by summing the probabilities for each operational state as shown in Equation (6.24).

$$P_{\text{init}}(t) = \prod_{i \in \{sys^+\}} R_i(t-T)$$ (6.23)

$$R_{\text{system}}(t) = \sum_{i=1}^l P_i(t)$$ (6.24)
In this section, the system reliability is obtained when the most degraded units are not forced-down if the degradation difference between the most and least degraded units is larger than the threshold: $d_c$ during the observation. When degradation balance mechanisms are not applied, this reliability is regarded as the lower bound of the system reliability.

### 6.3.2 Simulation of $(k_1,k_2)$-out-of-$(n,m)$ Pairs: $G$ Balanced Systems.

In section 6.3.1, a lower bound system reliability is obtained as a closed-form expression. However, when degradation mechanisms are applied at each observation, system reliability can only be obtained by simulation. In this section, we discuss a simulation model for reliability calculation for the $(k_1,k_2)$-out-of-$(n,m)$ pairs: G balanced systems. With the degradation data from observed degradation paths, we can estimate the parameters based on MLE for both inverse Gaussian and improved inverse Gaussian models. Based on the estimated parameters, we create $N=15000$ degradation paths for the inverse Gaussian or improved inverse Gaussian process. Each path is observed until its degradation value reaches the threshold $d_h$ or $t$ (a large value to ensure that the degradation process fails before $t$). The simulation process is described in two parts of an algorithm: Part I: $N$ degradation paths are generated based on degradation models (inverse Gaussian or improved inverse Gaussian) for each unit in the system. The failure times for the $N$ paths are archived in a vector `failure_time`. Part II: Based on $N$ degradation paths’ failure time, obtain the system reliability. In Part I of the algorithm, function `check_failure` is called to determine the systems’ operation status and whether the unit with highest amount of
degradation should be forced down. Function unit_status is called to change units’ status accordingly. These two parts are described below.

**Part I: Simulate N degradation paths**

```plaintext
While i ≤ N do
  Create a 3-dimensional zero matrix D of size \( n \times 2m \times (t + 1) \) to track the degradation history.
  Create a 3-dimensional zero matrix I of size \( n \times 2m \times t \) to save increments observations.
  Create a 3-dimensional all-ones matrix U of size \( n \times 2m \times t \) to mark units’ status (operating: 1; failed: 0; forced-down: 3) during each observation.
  Create a vector failure_time of size N to archive the failure time of each degradation path.
  for (j = 1, j ≤ t, j++) do
    Generate a random number matrix of size \( n \times 2m \) based on degradation model and assign the matrix to \( I(:,:,j) \) as the first increments observation for all units.
    Update the matrix I: \( I(:,:,j) = I(:,:,j) \odot (U(:,:,j) == 1) \).
    Update the degradation matrix: \( D(:,j+1) = I(:,:,j) + D(:,:,j) \).
    Check if the system is failed or need to be balanced using function check_failure:
    \[
    \text{[index_fail, index_adjust, max_row, max_col]} = \text{check_failure}(D(:,: + 1), U(:,:)).
    \]
    if index_fail = 0 do
      Update unit’s status matrix using function unit_status:
      \( U(:,j+1) = \text{unit_status}(\text{max_row, max_col, D(:,j+1), U(:,j), index_adjust}) \).
    else
      Update the vector failure_time: failure_time(i) = j.
    break
  end if
end for
i++.
end while
```

**Part II: Obtain system reliability**

Find the unique values of vector failure_time and sort them in ascending order and save in vector unique_values_sorted.
Count the number of occurrences for each element in unique_values_sorted and divide them by \( N \) to obtain the frequency. Save the frequency in vector frequency.
Find the accumulated sum for vector frequency and save in vector cumsum.
Reliability is obtained by \( R(t) = 1 - \text{cumsum} \).

**Function: check_failure**

Assign 0 to num_operating_pair.
```plaintext
for (i = 1, i ≤ n, i++) do
```
if \( \sum (D(i, 1; m, j + 1) \leq d_k) \geq k_2 \) and \( \sum (D(i, m + 1; 2m, j + 1) \leq d_k) \geq k_2 \) do
\[ \text{num-operating-pair} += 1. \]
end if
end for
if \( \text{num-operating-pair} \geq k_1 \) do
index_fail = 0.
else
index_fail = 1.
return
end if

Update units’ status matrix by:
\[ U = (U(\cdot, j) == 1) + (U(\cdot, j) == 3). \]
Update degradation matrix by:
\[ D = D(\cdot, j + 1) \odot U. \]
Find the maximum value in matrix \( D \) and save it as \( \text{max} \).
Find the row and col of \( \text{max} \) in \( D \) and save it in \( \text{max_row} \) and \( \text{max_col} \).
Delete the failed units in matrix \( D \): \( D(D == 0) = 1500 \).
Find the minimum value in matrix \( D \) and save it as \( \text{min} \).
Calculate the range between \( \text{max} \) and \( \text{min} \): \( \text{diff} = \text{max} - \text{min} \).
if \( \text{diff} \geq d_c \) do
\[ \text{index-adjust} = 1. \]
else
\[ \text{index-adjust} = 0. \]
end if

**Function: unit_status**
for \( i = 1, i \leq n, i + + \) do
for \( l = 1, l \leq 2m, i + + \) do
if \( U(i, l, j) == 1 \) and \( D(i, l, j + 1) \geq d_k \) do
\[ U(i, l, j) = 0. \]
if any \( U(i, k, j) = 2 \) (unit at \( k^\text{th} \) column are from the same side of unit at \( l^\text{th} \) column)
\[ U(i, k, j) = 1. \]
else
\[ U(i, h, j) = 2 \] (unit at \( h^\text{th} \) column are from the opposite side of unit at \( l^\text{th} \) column).
end if
if \( i == \text{max-row} \) and \( l == \text{max-col} \) do
\[ \text{indicator} = 1. \]
end if
end if
end for
end for
if \( \text{indicator} = 1 \) do
return
else
Reliability functions for the \((k_1, k_2)\)-out-of-\((n, m)\) pairs: G balanced systems are obtained using the above algorithm.

### 6.4 Numerical Examples

In this section, we provide reliability analysis and comparison for \((k_1, k_2)\)-out-of-\((n, m)\) pairs: G balanced systems.

**Scenario 1: Reliability functions for the \((2,2)\)-out-of-\((3,3)\) pairs: G balanced system when units follow same degradation process with different scale and shape parameter**

Suppose that all units in the \((2,2)\)-out-of-\((3,3)\) pairs: G balanced system follow inverse Gaussian process with the same mean \(\mu\) but different scale parameter \(\lambda\). The mean \(\mu\) is predetermined as 0.03 which means that the average degradation increment is 0.03 degradation units per time (the unit scale is arbitrary). The scale parameters \(\lambda\) are 1.2, 0.5 and 0.05 for the three systems. Figure 6.2 shows that with a smaller \(\lambda\), the system reliability is improved. At the same time, the distance between the reliability and the lower bound reliability is increased for the system with a smaller \(\lambda\). This means that the strategy of forcing down the most degraded units ‘temporarily’ is essential when \(\lambda\) is small.
Similarly, Figure 6.3 shows the reliability function for the (2,2)-out-of-(3,3) pairs: G balanced system when units follow inverse Gaussian process with same scale parameter and different means. The scale parameters $\lambda$ is preset as 1.2. The means (average degradation growth per unit time) are 0.01 and 0.03 respectively for the two systems. As expected, the system reliability with a larger mean fails first. The distance between reliability and lower bound reliability is larger for the system with a smaller mean. This is because that with a smaller average degradation increment, there are more opportunities to balance the degradation and hence a much higher reliability comparing with the lower bound reliability.
Figure 6.3 Reliability of (2, 2)-out-of-(3, 3) Pairs: G Balanced System with Different Mean

Figure 6.4 shows the reliability function for the (3,3)-out-of-(4,4) pairs: G balanced system when all units follow improved inverse Gaussian process with same scale parameter $\lambda$ and different shape functions. The shape function for the improved inverse Gaussian process is

$$\Lambda(t_i(t_j)) = \mu_0 + \mu_i t_j(t_{j-1})$$.

For the systems in Figure 6.4, the parameters in the shape function are: $\mu_0 = 0.03$; $\mu_i = 0.05$. The scale parameters are set as 0.5 and 1.5 respectively. The reliability functions shown in Figure 6.4 share the same conclusion as in Figure 6.2.
Scenario 2: Reliability functions for the \((k_1, k_2)\)-out-of-(\(n, m\)) pairs: G balanced systems with different configurations

In this scenario, we discuss the reliability for different \((k_1, k_2)\)-out-of-(\(n, m\)) pairs: G balanced systems with units following the same degradation process. For the two systems in Figure 6.5, units’ degradation processes follow the same inverse Gaussian process with mean \(\mu = 0.03\) and scale parameter \(\lambda = 0.5\). Similarly, units in the two systems of Figure 6.6 follow an improved inverse Gaussian process with the same parameters: \(\mu_0 = 0.03; \mu_1 = 0.05; \lambda = 0.5\). It is obvious that systems with higher \(\frac{k_1}{n}\) and/or \(\frac{k_2}{m}\) ratios have a lower reliability compared with systems with lower ratios. This is because when more units are required to work, the system reliability decreases.
Figure 6.5 Reliability of \((k_1, k_2)\)-out-of-\((n, m)\) Pairs: G Balanced System
When Units Follow Inverse Gaussian Process

Figure 6.6 Reliability of \((k_1, k_2)\)-out-of-\((n, m)\) Pairs: G Balanced System
When Units Follow Improved Inverse Gaussian Process
6.5 Case Study

In this section, we discuss the application of the proposed degradation model in Lithium Ion battery’s degradation process. Recent years, the battery electric vehicles (BEVs) and the battery unmanned aerial vehicles (UAVs)’ rising popularity leads to a significant research on battery’s degradation process and its lifespan. The constant discharge/charge cycle is considered as the most important reason for battery aging. As the number of cycles of discharge/charge increases, battery capacity decreases. When the battery capacity falls below a certain threshold, the battery is considered failed.

In this case study, capacity degradation data for four batteries are observed during the discharge/charge processes as shown in Table 6.1. Figure 6.7 shows that the degradation processes are linear before the capacity reaches the level of two and after that, the battery capacity decreases exponentially. Therefore, we divide the degradation process into two parts: Part I includes the degradation process before the capacity reaches a value of two and Part II includes the rest of degradation process until it reaches the failure threshold. We fit each part with different degradation models including Brownian motion, inverse Gaussian process, Gamma process and improved inverse Gaussian process. We observe that Brownian motion with a constant drift $\mu$ and a diffusion coefficient $\sigma$ is the best fit for the degradation data when the degradation is less than two. While as the degradation is greater than two, an improved inverse Gaussian process with shape function $\Lambda(r_i(t_j)) = \mu(r_i(t_{j-1}))$ and a scale parameter $\lambda$ is the best fit. The estimated parameters for the two parts are shown in Table 6.2.
Table 6.1 Battery Capacity Degradation vs. Number of Cycles

<table>
<thead>
<tr>
<th>Cycles</th>
<th>Battery 1</th>
<th>Battery 2</th>
<th>Battery 3</th>
<th>Battery 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>2.54</td>
<td>2.53</td>
<td>2.52</td>
<td>2.50</td>
</tr>
<tr>
<td>150</td>
<td>2.50</td>
<td>2.50</td>
<td>2.49</td>
<td>2.47</td>
</tr>
<tr>
<td>300</td>
<td>2.47</td>
<td>2.47</td>
<td>2.46</td>
<td>2.44</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>4500</td>
<td>1.71</td>
<td>1.65</td>
<td>1.75</td>
<td>1.73</td>
</tr>
<tr>
<td>4600</td>
<td>1.67</td>
<td>1.60</td>
<td>1.73</td>
<td>1.70</td>
</tr>
</tbody>
</table>

Figure 6.7 Battery Capacity Degradation Process

Table 6.2 Estimated Parameters

<table>
<thead>
<tr>
<th></th>
<th>Brownian motion for Part I</th>
<th>Improved inverse Gaussian for Part II</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\mu$</td>
<td>0.02</td>
<td>1.095</td>
</tr>
<tr>
<td>$\sigma$</td>
<td>0.012</td>
<td>N/A</td>
</tr>
<tr>
<td>$\lambda$</td>
<td>N/A</td>
<td>196.32</td>
</tr>
</tbody>
</table>
We simulate the battery degradation process using the estimated parameters as shown in Figure 6.8. In this simulation, we assume that the initial battery capacity is 2.6. The battery capacity degradation follows Brownian motion using the estimated parameters initially. When the battery capacity reaches two and starts to decrease exponentially, the degradation process follows improved inverse Gaussian. After comparing the simulated degradation paths in Figure 6.8 with the observed degradation paths in Figure 6.7, we conclude that the proposed model is appropriate for the battery capacity degradation modeling.

**Figure 6.8** Simulated Battery Capacity Degradation Process

*Scenario 1: Reliability of (3,3)-out-of-(4,4) and (2,2)-out-of-(4,4) pairs: G balanced systems*
In this scenario, we suppose that the system has four pairs of batteries with each pair having four batteries on both sides. Reliabilities and its lower bounds for the (2,2)-out-of-(4,4) and (3,3)-out-of-(4,4) pairs: G balanced system are shown in Figure 6.9. When the degradation balance mechanism is applied, the reliability for the (3,3)-out-of-(4,4) pairs: G balanced system is found to be higher than the lower bound reliability for the (2,2)-out-of-(4,4) pairs: G balanced system. This observation suggests that degradation balance mechanism largely improves the reliability of system with a larger number of required working units.

**Figure 6.9** Reliability of (2,2)-out-of-(4,4) and (3,3)-out-of-(4,4) Pairs: G Balanced Systems

**Scenario 2:** Reliability of \((k_1, k_2)\)-out-of-\((n,m)\) pairs: G balanced systems with different thresholds \(d_c\)
In this scenario, we compare the reliability for \((k_1, k_2)\)-out-of-\((n, m)\) pairs: G balanced systems when the threshold values \(d_c\) are different. In Figures 6.10 and 6.11, the reliability functions for the \((2,2)\)-out-of-(4,4) and \((3,2)\)-out-of-(4,4) pairs: G balanced system are improved when \(d_c\) is small. This is because when the degradation among units are balanced more frequently, the reliability improvement is more significant. We also observe that the improvement is larger for the \((2,2)\)-out-of-(4,4) system in comparison with the \((3,2)\)-out-of-(4,4) system. This is due to the fact that when the system requires more units to be working, there are less opportunities to balance the degradation unevenness among units.

![Figure 6.10 Reliability of (2,2)-out-of-(4,4) Pairs: G Balanced System](image)

**Figure 6.10** Reliability of \((2,2)\)-out-of-(4,4) Pairs: G Balanced System
6.6 Conclusions

In this chapter, we revisit the system description and balance requirement for the Type I and Type II \((k_1, k_2)\)-out-of-(\(n, m\)) pairs: G balanced systems. We estimate the system reliability based on units’ degradation processes. Among the different degradation models, we focus on the inverse Gaussian class due to its monotonicity and practicality. Besides the regular inverse Gaussian process, an improved inverse Gaussian process is introduced to capture the degradation increments’ dependency on previous degradation states.
When degradation differences among the units in the \((k_1, k_2)\)-out-of-\((n, m)\) pairs: G balanced systems are substantial, certain units reach the failure threshold first and other units are forced down to ensure the system balance. Thus, we propose a strategy to balance the unevenness among units’ degradation by forcing down the most degraded units temporarily during the entire degradation process. System reliability is obtained through simulation when this mechanism is applied. Moreover, a closed-form lower bound reliability is obtained through state enumeration and iterative calculations when it is not applied.

Finally, reliability analysis and comparisons are provided in numerical examples and a case study. In the numerical examples, we found that when the values of shape and scale parameters decrease, the system reliability improves for both the inverse Gaussian and the improved inverse Gaussian processes. The difference between the reliability (when degradation balance mechanism is applied) and the lower bound reliability is more substantial when the system has a higher reliability. This is because when system reliability is high, there are more opportunities to balance the degradation differences among units and hence difference between reliability and lower bound reliability becomes larger. We also found that higher ratios of \(\frac{k_1}{n}\) and or \(\frac{k_2}{m}\) result in a decrease in the system’s reliability which echoes the conclusions in Chapters 3 and 4.
In the case study, we fit a degradation model combining the Brownian motion and improved inverse Gaussian process to the battery capacity degradation data. Parameters for these two models are estimated. The model is validated by comparing the original degradation data and the simulated degradation path. Then, we assume that the batteries are units in the \((k_1, k_2)\)-out-of-\((n, m)\) pairs: G balanced systems and compare the system reliability for different scenarios. It is shown that for systems with higher ratios of \(\frac{k_1}{n}\) and \(\frac{k_2}{m}\), when degradation balance mechanism is applied, the system reliability may exceed the lower bound reliability of systems with lower ratios of \(\frac{k_1}{n}\) and \(\frac{k_2}{m}\). It also shown that when the threshold \(d_c\) is large, which means the degradation process is less balanced, the system reliability decreases.
CHAPTER 7
CONCLUSIONS AND FUTURE RESEARCH

7.1 Conclusions

The major contributions of this dissertation include the investigation of methodologies for reliability estimation of different balanced systems with multi-dimensional distributed units. We focus on two system configurations: 1. balanced systems with multi-level circular configurations referred to as the \((k_1,k_2)\)-out-of-\((n,m)\) pairs: G balanced system; 2. balanced systems with spherical configurations referred to as the spherical \(k-n-i\): G balanced system. The reliability estimation of balanced systems with spatially allocated units depends not only on the individual units’ operational status but also on operating units’ spatial locations due to the balance requirement. We investigate the functions of systems with spatially distributed units and define balance requirements for the multi-level circular system and the spherical system. We develop analytical and algorithmic solutions for reliability estimation of such systems as described below.

First, we develop reliability models for two types of the \((k_1,k_2)\)-out-of-\((n,m)\) pairs: G balanced systems. An unbalanced system is considered failed and thus is rebalanced by forcing down units while the system has more than the minimum number of working units. We consider two scenarios: forced-down units are considered as failed units and forced-down units are considered as standbys. For the first scenario, a double loop \(k\)-out-of-\(n\) model is proposed when all of the units in the system are identical. For the second scenario,
a continuous-time discrete-state stochastic process is developed when the units’ lifetimes are exponentially distributed. When units are not identical nor exponentially distributed, we develop procedures to enumerate system’s operational states and calculate probabilities for these operational states. This procedure is time consuming and error prone for large scale systems. We thus establish an algorithm for efficient computation. Optimal system design is given through numerical examples.

Second, we develop reliability models for spherical $k$-$n$-$i$: G balanced systems. We define the rotational and symmetrical balance for spherically balanced systems. Mathematical approaches are established for the determination of both balances. Rebalance rules are given for some of the simple imbalance situations with forced-down units considered as failed and standbys. However, for complex imbalance situations, we need to enumerate all possible rebalance options. Thus, an efficient algorithm is developed for large scale systems. Similarly, reliability is obtained by enumerating operational states and calculating probabilities of these states. Guidance for the optimal system design is proposed through numerical examples.

Finally, we propose degradation models for the $(k_1,k_2)$-out-of-$(n,m)$ pairs: G balanced system. A degradation balance mechanism is proposed to balance degradation values among units and hence reduce the probability of the system imbalance. When this mechanism is applied, all units’ degradation processes are observed in time intervals. The ‘most’ degraded unit is forced down temporarily during an observation when the
degradation difference between the ‘most’ and ‘least’ degraded units is larger than a certain threshold. We focus on two degradation process: inverse Gaussian process (stationary process) and improved inverse Gaussian process (non-stationary process). The improved inverse Gaussian process is developed to capture the non-linearity in degradation growth. A Monte Carlo simulation is proposed for the reliability estimation. A closed-form solution is introduced as the lower bound system reliability when the balance mechanism is not applied. A case study based on battery degradation shows that the balance mechanism improves the system reliability significantly.

7.2 Future Research

The research of this dissertation can be extended in several directions that have applications and potential implementations. We present two potential problems, they are:

Problem 1: In Chapter 6, we develop degradation models for the \((k_1,k_2)\)-out-of-\((n,m)\) pairs: G balanced systems. We show that the system reliability is largely improved when the degradation balance mechanism is applied. We also observe that the balance mechanism works best when units follow stationary degradation process. Similar approaches are needed for degradation models of the spherical k-n-i: G balanced systems. These are challenging issues since the two balance conditions (symmetrical and rotational) need to be considered while applying the degradation balancing mechanism for different degradation processes.
Problem 2: In Chapters 3, 4 and 5, optimal system design is given through numerical examples for both the balanced systems with multi-level circular configuration and spherical configuration. We observe that additional redundancy can reduce the system reliability by bringing more imbalance situations. Future research may analyze the optimal system design quantitively in order to achieve the highest system reliability.
REFERENCES


Gharib, M., Elsayed, E. M. and Nashwan, I. I. H. (2011). 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*, 3 (1): 47-56.


Habib, A., Yuge, T., Al-Seedy, R. and Ammar, S. J. A. M. M. (2010). Reliability of a consecutive $(r, s)$-out-of-$(m, n)$: F lattice system with conditions on the number of


