Reliability Engineering and System Safety 173 (2018) 23–33

Contents lists available at ScienceDirect

Reliability Engineering and System Safety

journal homepage: www.elsevier.com/locate/ress

Simplification of inclusion–exclusion on intersections of unions with application to network systems reliability

Lukas Schäfer[a,*], Sergio García[a], Vassili Srithammavanh[b]

[a] School of Mathematics, The University of Edinburgh, Edinburgh, United Kingdom
[b] AIRBUS Group Innovation, Paris, France

ARTICLE INFO

Keywords:
Structural reliability
Combinatorics
Non series-parallel systems
Inclusion–exclusion

ABSTRACT

Reliability of safety-critical systems is a paramount issue in system engineering because in most practical situations the reliability of a non series-parallel network system has to be calculated. Some methods for calculating reliability use the probability principle of inclusion–exclusion. When dealing with complex networks, this leads to very long mathematical expressions which are usually computationally very expensive to calculate. In this paper, we provide a new expression to simplify the probability principle of inclusion–exclusion formula for intersections of unions which appear when calculating reliability on non series-parallel network systems. This new expression exploits the presence of many repeated events and has many fewer terms, which significantly reduces the computational cost. We also show that the general form of the probability principle of inclusion–exclusion formula has a double exponential complexity, whereas the simplified form has only an exponential complexity with a linear exponent. Finally, we compare its computational efficiency against the sum of disjoint products method KDH88 for a simple artificial example and for a door management system, which is a safety-critical system in aircraft engineering.

1. Introduction

Reliability of a network system is the probability of the system not failing. It is a critical issue in different fields such as computer networks, information networks or gas networks. In particular, reliability of safety-critical network systems [19,21] is an important topic in system engineering. For example, aircraft architecture has safety-critical network systems such as fly-by-wire, actuation, fire warning and door management systems. In most practical situations, the reliability of a complex network system (e.g., a system that is not series-parallel) has to be calculated exactly [8]. There are several methods to calculate or simulate the reliability of a complex system which have been developed in recent decades. Some classical static modelling techniques, including reliability block diagram models [10], fault tree models, and binary decision diagram models, have been widely used to model static systems. A general introduction to these methods can be found in [21]. For time-dependent systems, modeling techniques such as Markov models [12], dynamic fault tree models [3] and Petri net models [30] have been used. Reliability system calculation can also be divided into systems or multi-state components, where the components of a system operate in any of several intermediate states with various effects on the entire system performance [14–17,20,25] or with binary-state components, where either a component works perfectly or not at all. In this paper, we consider binary-state components. Furthermore, reliability of complex systems with specific graph structures, like systems with a hypercube structure [11,13], got attention over the last years. But whereas most of these methods consider systems with two-terminal nodes or k-terminal nodes where all k-nodes have to be connected, we consider a different type of complex systems and a specific structure that has multiple functions with multiple start and end nodes.

In this paper, we propose a new method to calculate the reliability of such a complex system with a new way of writing the classic probability principle of inclusion–exclusion formula. The classical probability principle of inclusion–exclusion formula is

$$P\left(\bigcup_{i=1}^{n} A_i\right) = \sum_{i=1}^{n}\left((-1)^{i+1} \sum_{\substack{J\subseteq\{1,\dots,n\},\\ |J|=i}} P\left(\bigcap_{j\in J} A_j\right)\right). \tag{1}$$

The new method detects which combination of events leads to the same event when simplified and has, therefore, many fewer summands than the classical formula for intersections of unions.

Practical reliability calculations often involve very long expressions when the probability principle of inclusion–exclusion formula (1) is used. Therefore, there are many approaches in the literature on gen-

* Corresponding author.
E-mail addresses: sergio.garcia-quiles@ed.ac.uk (L. Schäfer), vassili.srithammavanh@airbus.com (V. Srithammavanh).

https://doi.org/10.1016/j.ress.2018.01.003
Received 26 May 2017; Received in revised form 8 December 2017; Accepted 4 January 2018
Available online 4 January 2018
0951-8320/

eral network reliability calculations to simplify the probability principle of inclusion–exclusion such as, for example, partitioning techniques [6] and the sum of disjoint products method [1,2,7,18,22,29]. The sum of disjoint products method is the most often used approach, with recent results in [5,23,26–28].

All these methods need the exact system structure to simplify the reliability calculation. Therefore, in this paper, we propose a new approach to simplify the probability principle of inclusion–exclusion without needing the exact system structure, and to apply it to the calculation of the reliability of complex network systems in system engineering. In the following, we introduce what kind of complex network systems we consider and why we consider a method that does not need the exact system structure.

In system engineering, most network systems have multiple functions that have to be performed and these are not always independent (e.g., they share components). Reliability can be increased if different sets of components in the network can perform the same function. Therefore, these functions are implemented multiple times in the network system through different sets of components, and calculation of the reliability of the network system becomes a very complex task.

In our paper, we assume that all failure probabilities of the components are known exactly. We do not consider the case when these probabilities are known only approximately (e.g., either by estimation or a confidence interval). If the different components of the network are independent of each other, then we can easily calculate the reliability of a set of components. Through this we can calculate the reliability of one implementation of a function, which is defined as the probability of the event that one implementation of the function does not fail. Finally, the probability of an intersection of such events can be calculated easily. However, if full independence cannot be assumed, then the calculation becomes very expensive, usually prohibitively so.

Our main motivation lies in dealing with optimization problems with reliability constraints where this calculation means that costs for models even for very small networks make the problems intractable. The main reason for this is the large number of variables and non-linear constraints involved in the reliability calculation within the optimization model. Furthermore, the approaches to simplify the probability principle of inclusion–exclusion formula mentioned before (e.g., sum of disjoint products) are not suitable for use within an optimization formulation, because the exact structure of the system has to be known before constructing the reliability constraints. Also, approximation and the use of lower or upper bounds for the principal of inclusion–exclusion are not suitable for exact optimization because they are not in general monotone increasing or decreasing with regard to the reliability. In practice, optimization models that involve reliability are usually either solved through heuristics or by assuming series-parallel systems [4,9,24].

In this paper, we show how to calculate the reliability of a network system in which components are not necessarily independent in a way that requires considerably fewer operations (and, thus, it is much cheaper computationally) than the direct use of the probability inclusion–exclusion principle. The key is to exploit the fact that, when dealing with a network system, the probability inclusion–exclusion principle has many repeated terms when applied to intersections of unions. Furthermore, we compare the computational efficiency and number of summands of our new method with the sum of disjoint products method KDH88 from [7]. KDH88 is a sum of disjoint products method with multiple-variable inversion that can be easily applied to a network system with subsystems that are subgraphs with multiple start and end nodes and not just paths with one start and end node. Therefore, it is applicable for the complex systems we consider and is suitable to make comparisons against.

The rest of the paper is organised as follows. In Section 2, we provide our motivation by showing why it can be expensive to calculate the reliability of a non series-parallel system. We also state the main result (Proposition 1) which provides a formula to calculate the reliability for a network system in an exact way with a much lower number of operations. To be able to use the formula from Proposition 1, we provide an algorithm in Appendix C which can be easily implemented. In Section 3, we compare our method with KDH88 and the classic probability principle of inclusion–exclusion (3) for simple artificial examples and a door management system application. Finally, we provide some conclusions and discuss future perspectives in Section 4.

Table 1
Number of summands in the probability principle of inclusion–exclusion formula.

$\lvert\mathcal{F}\rvert$	$\lvert\mathcal{F}_i\rvert$	Summands
2	2	15
2	3	5.11×10^{2}
2	4	6.55×10^{4}
3	2	2.55×10^{2}
3	3	1.34×10^{8}
3	4	1.84×10^{17}
4	2	6.55×10^{4}
4	3	2.41×10^{24}
5	2	4.29×10^{9}
5	3	1.41×10^{73}

2. Motivation and main result

We start by showing that, if independence cannot be assumed, then it can be very expensive to calculate the probability of a non series-parallel network system with multiple functions and implementations. Afterwards, we introduce a result (Proposition 1) that reduces the number of calculations involved. Let n be the number of functions in the system and t_i be the number of implementations of function i in the system. Let $F_i,\ i \in \{1,\dots,n\}$, be the event that function i of a system does not fail in a specific period of time and $F_{ij},\ j \in \{1,\dots,t_i\}$, be the event that implementation j of function i does not fail in a specific period of time. Let $\mathcal{F} = \{F_1,\dots,F_n\}$ be the set of all functions and $\mathcal{F}_i = \{F_{i1},\dots,F_{it_i}\}$ be the set of all implementations of function i. Furthermore, let R be the event that the system does not fail. The reliability of the system, $P(R)$, is the probability that no function in $\mathcal{F}$ fails. A function $F \in \mathcal{F}$ does not fail if at least one of its implementations does not fail. Therefore,

$$P(R) = P\left(\bigcap_{i=1}^{n} F_i\right) = P\left(\bigcap_{i=1}^{n}\left(\bigcup_{j=1}^{t_i} F_{ij}\right)\right).$$

Because the different functions and implementations may not be independent, $P(R)$ is not easily calculable. In order to work on this expression, first we need to establish some notations. Let

$$W = \{1,\dots,t_1\} \times \dots \times \{1,\dots,t_n\}, \text{ and}$$

$$B_w = \bigcap_{i=1}^{n} F_{iw_i} \quad \text{for } w = (w_1,\dots,w_n) \in W,$$

where $w_i \in \{1,\dots,t_i\}$ and represents the implementation index of function i. We then have that

$$P(R) = P\left(\bigcap_{i=1}^{n}\left(\bigcup_{j=1}^{t_i} F_{ij}\right)\right) = P\left(\bigcup_{w\in W}\left(\bigcap_{i=1}^{n} F_{iw_i}\right)\right) = P\left(\bigcup_{w\in W} B_w\right). \tag{2}$$

Now the probability principle of inclusion–exclusion can be used and it follows that

$$P(R) = \sum_{t=1}^{|W|}\left((-1)^{t+1}\sum_{\substack{I\subseteq W,\\ |I|=t}} P\left(\bigcap_{j\in I} B_j\right)\right). \tag{3}$$

The number of summands in (3), which is equal to the number of possible intersections of B_w's, is $\sum_{t=1}^{|W|}\binom{|W|}{t} = 2^{|W|}-1$ with $|W| = \prod_{i=1}^{n}|\mathcal{F}_i|$. We therefore have a doubly exponential computational complexity. Table 1 shows the number of summands for different values of the number of functions and implementations, with the assumption that every function has the same number of implementations.