

Ji-Min LU
Fares INNAL
Xiao-Yue WU
Yiliu LIU
Mary Ann LUNDTEIGEN

TWO-TERMINAL RELIABILITY ANALYSIS FOR MULTI-PHASE COMMUNICATION NETWORKS

ANALIZA NIEZAWODNOŚCI PAR TERMINALI W WIELOFAZOWYCH SIECIACH KOMUNIKACYJNYCH

Most researches of network reliability generally assume that the system structures do not change with time. This paper presents the concept of multi-phase network systems (MPNS) to consider dynamic characteristics of networks, and analyze the reliability of MPNS. MPNS reliability is evaluated through a cross-phase binary decision diagram (BDD). The BDD-based algorithm can act as a platform to consider various components behaviors such as repair and growing pressure. Case study shows that the proposed MPNS concept is an effective description of some practical communication networks, and the cross-phase BDD model is efficient in analyzing MPNS reliability.

Keywords: Binary decision diagram, component-behavior model, multi-phase network systems, system reliability.

Większość badań niezawodności sieci ogólnie przyjąć, że struktury systemu nie zmieniają się w czasie. W artykule przedstawiono koncepcję systemów sieciowych wielofazowych (MPNS) rozpatrywanie dynamicznych właściwości sieci i analizy niezawodności MPNS. MPNS niezawodność jest oceniany przez cross-fazowego schematu decyzyjnego binarny (BDD). Algorytm z siedzibą w BDD może działać jako platforma do rozważenia różnych komponentów zachowań, takich jak naprawy i rosnącej presji. Studium przypadku pokazuje, że proponowana koncepcja MPNS jest skutecznym opis niektórych praktycznych sieci komunikacyjnych, a cross-fazowego modelu BDD jest skuteczny w analizie MPNS niezawodność.

Słowa kluczowe: Binarny schemat decyzji, model zachowania komponentów, systemów sieciowych wielofazowe, niezawodność systemu.

1. Introduction

Network has become a common and effective description for many engineering systems such as transportation, communications, and gas-pipeline systems. In graph theory, the network is usually denoted as $G=(V,E)$, where V is the vertex set representing terminals (bus stop, switchboard, etc.), and E is the edge set representing links (road, cable, etc.). Graph theory further classifies edges into directed and undirected edges. Communication networks, as well as transportation systems, are typical examples of undirected networks. In literature, networks structures normally remain unchanged during the entire mission. However, changing network structures and varying stress on components have become increasingly common in many engineering applications. This kind of systems can be termed as the multi-phase network systems (MPNS) in comparison to traditional single-phase network systems (SPNS). A typical example of MPNS is the satellite telemetry, tracking and control (TT&C) system which is responsible for the data transmission between satellites and ground facilities. In satellite TT&C systems, communications between a certain ground station and a target satellite usually rely on many relay satellites. As different relay satellites fly over the ground station, the communication networks changes its link structures and constitutes a MPNS.

Two-terminal reliability (or terminal-pair reliability) is one of key concern in the design and maintenance of network systems. Take communication systems for instance. The two-terminal reliability can be considered as the probability of successfully transmitting data from

one source to one receiver. Researches on two-terminal reliability of single-phase networks have been extensively conducted since 1970s. Early methods are mainly based on the enumeration of minimal paths/cuts, or based on the factoring theory. In minimal paths/cuts methods [2, 9, 16, 23, 25, 28], the reliability is evaluated by enumerating all minimal paths/cuts and summing the probabilities of their disjoint forms. A *path* is a set of network components (edges/vertices) such that if these components are operational, the system is up. A *path* is minimal if it has no proper subpaths. Conversely a *cut* can be considered as a set of network components such that if these components fail, the system is down. Literature [1, 22] shows that the number of cuts is usually much smaller than the number of paths for many practical systems, meaning that cut-based methods (i.e. to calculate the unreliability) have better performance. However, both paths and cuts based method have difficulties in applying to large networks since the number of paths/cuts may grow exponentially with the network size.

In contrast to paths/cuts based approaches, factoring methods (or decomposition/ topology methods) may exhibit better performance, especially combined with some reduction techniques [27, 30, 31, 38]. In factoring methods, a certain component of the network is chosen and then the network is decomposed into two subnetworks. One subnetwork assumes the component is up and the other subnetwork assumes the component is down. Another version of this method does not choose a single component, but replaces some special substructures by smaller ones [8, 12]. Some experiments [32, 43] show that

factoring methods (with reduction) are more effective than classical minimal paths/cuts methods.

Methods for network reliability analysis are normally based on Boolean algebra [36], fault tree [40], digraph [4, 5, 18, 19], and BDD [37, 45, 49]. Kelly and Bartlett [4, 5, 18, 19] compare the digraph methods with the fault-tree methods in system fault diagnostics. In the 1990s, binary decision diagrams (BDD) based approaches [11, 14, 15, 19, 22, 33, 47] became increasingly popular for network analysis. A strength of BDD-based methods is that the BDD provides a succinct and disjoint description of system success/failure. Another strength is that BDD can act as a platform to consider various engineering problems such as common cause failure (CCF) [47], imperfect vertices [22], k -terminal network reliability [15], etc. The efficiency of BDD methods are measured by the BDD size which depends heavily on the chosen orderings of BDD variables. Hence, a big challenge facing BDD-based methods is the determination of the optimal orderings strategy and the automatic generation of BDD for large systems. Although literature [6] shows that improving the variable ordering of BDD is NP -complete, Friedman [11] proposes an algorithm (of time complexity $O(n^2 3^n)$) to find the optimal variable ordering. For small networks, existing ordering algorithms (such as breadth-first searching [19, 44]) can usually generate a BDD which is succinct enough for reliability computation.

From literature it can be seen that extensive efforts have been expended to evaluate the reliability of single-phase network. Nevertheless, we can hardly find the solution to analyze MPNS reliability. Some approaches [10] may consider changes in network connectivity, but they seldom make allowance for multi-phase characteristics such as the varying system configuration, changing failure rates, and repair activities of network components, etc. Based on the BDD merging technique, this paper proposes a non-simulation approach to analyze the MPNS reliability. Additionally, the proposed method uses “o analyze the ration, changing component-behavior model” to consider phenomena such as the components repair, changes in failure rates, and multi-state components. Our method can be extended to consider unreliable vertices by minor modification to the algorithm.

The remainder of the paper is organized as follows. In Section 2, readers are briefed on the basics of phased-mission systems and BDD. In Section 3, the proposed BDD-based algorithm is presented as the mean to evaluate MPNS reliability. In Section 4, we insert the component-behavior model into the BDD-based method to analyze MPNS with repairable edges/vertices. Section 5 depicts two cases to show that MPNS is effective in analyzing the reliability of communication systems. Finally, we draw the conclusions in Section 6.

2. Preliminary knowledge

2.1. Brief of phased-mission-system reliability

The phased-mission systems (PMS), or multi-phase mission systems, describe the systems in which the components stress and system structure may change from phase to phase. An example of PMS is the rocket launching mission where the boosters, engines, and solar panels are used at different time, resulting in different system structures in different phases. A challenge facing the reliability evaluation of PMS is the dependency problem which means a single component may appear repeatedly across phases. As consequence, the reliability of PMS is not the simple multiplication of the reliability of each phase. Such challenge does not appear in single-phase systems.

Researches in the field of PMS reliability are prevalent. Existing non-simulation approaches are normally based on Boolean algebra [36], fault tree (FT) [40], BDD [37, 45, 49] and Markov chains [3, 18, 34, 35, 50], etc. Markov-chains based approaches are mainly used to analyze PMS with repairable components. In contrast, the BDD, FT,

and Boolean-algebra based algorithms are relatively more efficient in analyzing non-repairable components. A significant challenge facing most existing PMS researches is that they cannot be applied to large PMS (contain up to 100 components & phases). For large systems, most PMS approaches will encounter various explosion problems, such as the state-space explosion for Markov-chain models and the BDD-size explosion for BDD models.

Fortunately, methods analyzing small-scale PMS reliability are relatively mature (especially the BDD and Markov-chain based methods). These methods have been expanded to consider various engineering problems, such as the imperfect coverage [46], combinatorial phase requirements [24], and CCF [45, 46] etc. For PMS reliability, a remarkable algorithm is the BDD-Markov combined method [41] which builds a separate Markov chain for each component. The separate Markov chain and associated matrices manipulation can be termed as the “component behavior model” [24] which is able to depict various components’ failure and repair activities in a given phase. One strength of the component-behavior model lies in its efficiency in analyzing PMS with many repairable components, while traditional Markov-chain models are difficult to deal with such PMS. This paper uses the component-behavior model to allow for various component activities (such as repair and idle) and their effects on network reliability.

A noticeable difference between PMS and MPNS is that some PMS approaches cannot be applied to MPNS. For instance, the reliability block diagram (RBD) based methods and the FT methods are inapplicable to MPNS. This paper uses the BDD-based method to analysis MPNS reliability.

2.2. BDD method for reliability evaluation

BDD or ordered BDD (OBDD) based methods [14, 15, 19, 22, 33, 47] have been widely used in analyzing network reliability since 1990s. BDD is a directed graph based on Shannon decomposition. The Shannon decomposition for a Boolean function f is defined as:

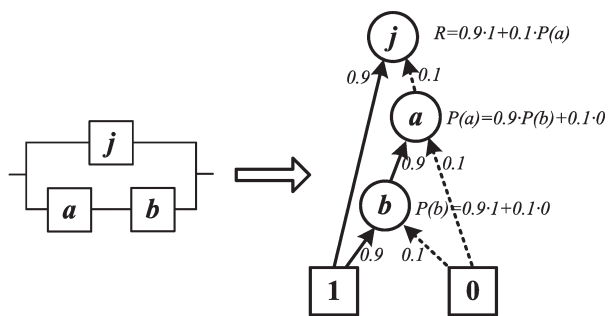
$$f = x \cdot f_{x=1} + \bar{x} \cdot f_{x=0} \quad (1)$$

where x is one of the decision variables, and $f_{x=1}$ is the expression of f at $x=1$ (i.e. x is true). By choosing an ordering over all variables, a given Boolean expression (such as Eq.) can be expressed as a binary tree by recursively applying the Shannon decomposition. Such decomposition may result in a space-consuming BDD if no proper variable ordering is chosen. Basically, reliability evaluation based on BDD contains the following two steps.

Step (1) – Choose an ordering strategy for network components, and generate BDD according to the chosen ordering. The chosen ordering strategy should ensure the resulting BDD as succinct as possible as the size of BDD depends heavily on the ordering. Readers may refer to [7, 17, 29] for an overview about the existing ordering methods.

Step (2) – Calculate the network reliability based on the BDD.

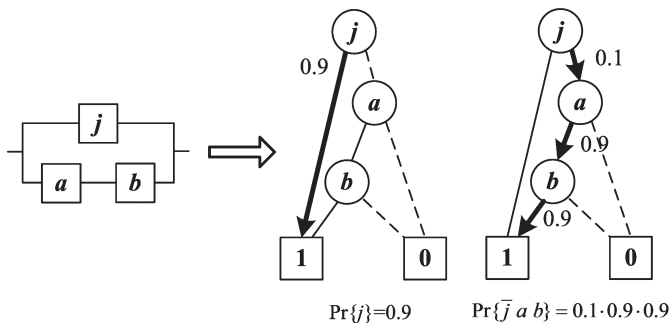
In general, reliability evaluation based on BDD can be categorized into two types: one is the bottom-up algorithm which traverses all BDD nodes; the other one is the top-down algorithm which enumerates all BDD paths. In the bottom-up algorithm, we calculate the probability associated with each node, from the bottom node to the top node using a recursive method based on the Shannon Decomposition of Eq. . The probability of the top node equals to the system reliability. For instance, assume the reliability of each edge (a, b, j) is 0.9 (independent of time). The bottom-up evaluation of BDD is illustrated in Fig. 1.



Reliability block diagram Binary decision diagram

Fig. 1. Bottom-up reliability evaluation based on BDD nodes

On the other hand, the top-down algorithm calculates the probability of each BDD path which connects the top node with the bottom node $\bar{1}$. And then it sums up the probabilities of these paths to obtain the system reliability. For the system in Fig. 2, there are two paths from j to $\bar{1}$ (see Fig. 2), therefore the system reliability is $R = \Pr\{j\} + \Pr\{\bar{j} a b\}$ where \bar{j} represents that j fails during the mission.



Reliability block diagram Binary decision diagrams

Fig. 2. Top-down reliability evaluation based on BDD paths

Compared with the top-down algorithm, the bottom-up algorithm may be more efficient to traditional (single-phase) systems as the number of BDD nodes is normally smaller than the number of BDD paths. However, for PMS, just the opposite is true. That is because the bottom-up algorithm is applicable only to BDD where nodes of the latter phase are placed higher than (or in front of) nodes of the former phase. For PMS, such BDD are almost too complex to generate. Hence the algorithm in the next section uses the top-down algorithm which is applicable to a variety of BDD.

3. BDD analysis for MPNS

In this section, a BDD-based method is proposed to analyze MPNS reliability. Generally it contains two steps: the first step is to produce the cross-phase BDD of MPNS; the second step is to calculate MPNS reliability based on the cross-phase BDD.

Step 1: BDD generation for the entire MPNS

The algorithm begins with the separate generation of BDD for every phase with classical methodologies [44]. And then these BDD are integrated together to form the final BDD of the entire MPNS. To illustrate the algorithm, consider a 3-phase MPNS which is illustrated in Fig. 3.

There are plentiful methods of generating BDD for networks in Fig. 3. Here we use the generation algorithm in [44]. Take the bridge

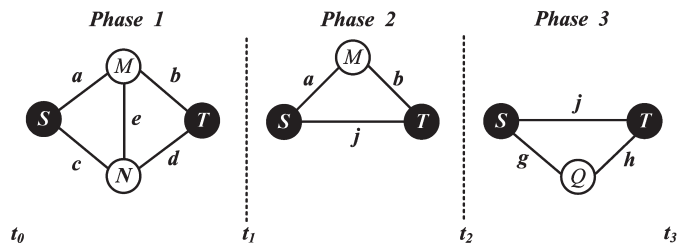


Fig. 3. MPNS example with 3 phases and duplicating components

network in Fig. 3 (first phase) for example, minimal paths are ab , aed , cd , and ceb ; hence the Boolean function of the phase-1 network is:

$$F_{phase1} = ab + aed + cd + ceb \tag{2}$$

In Eq. (2), let $L(x)$ be the length of Boolean variable x . $L(x)$ is defined as the length of the shortest minimal path containing x . For instance, the length of Boolean variable a is $L(a)=2$ because the shortest minimal path containing a is ab (aed is not shortest); similarly $L(b)=L(c)=L(d)=2$, and $L(e)=3$. And then F_{phase1} is decomposed by the Boolean variable whose length is shortest, meaning that a is chosen (b , c , and d are also appropriate here), i.e.:

$$\begin{aligned} F_{a=1} &= b + ed + cd + ceb \\ F_{a=0} &= cd + ceb \end{aligned} \tag{3}$$

When there are more than one Boolean variables whose lengths are shortest, we choose the variable which appears most frequently in Eq. (2). Above decomposition process (Eq. (3)) will repeat to form the BDD of each phase, as shown in Fig. 4. In Fig. 4, BDD node x_i represents the behavior of edge x during phase i . For instance, j_2 represents the behavior of edge j during phase 2. In this section, the assumption that all network vertices are perfectly reliable excludes network vertices from BDD.

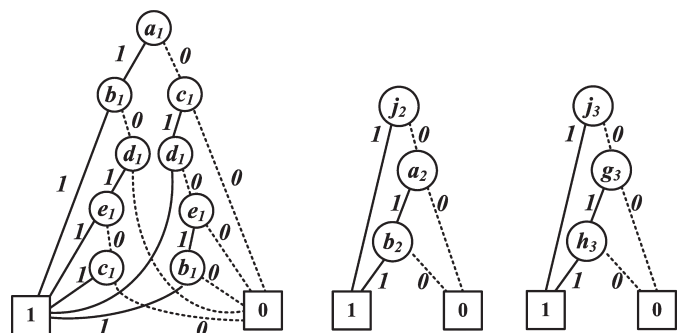


Fig. 4. BDD of phase 1, 2, and 3

Next, we integrate above BDD one-by-one to form a large BDD of the entire MPNS. To do this, we combine the top node in the next-phase BDD with the bottom node $\bar{1}$ in the former-phase BDD (see Fig. 5), based on the assumption of AND logic between phases (i.e., the system can progress into the task of the next phase only when it successfully finishes the previous-phase task.). In case studies readers will find this assumption common in practical MPNS. The new cross-phase BDD represents the system behavior (success/failure) from phase 1 to phase 2. Although the BDD combining solution can

be found in [41], it is applied to network systems for the first time. After the last-phase BDD is merged, we obtain the final BDD of the entire MPNS, which is shown in Fig. 6.

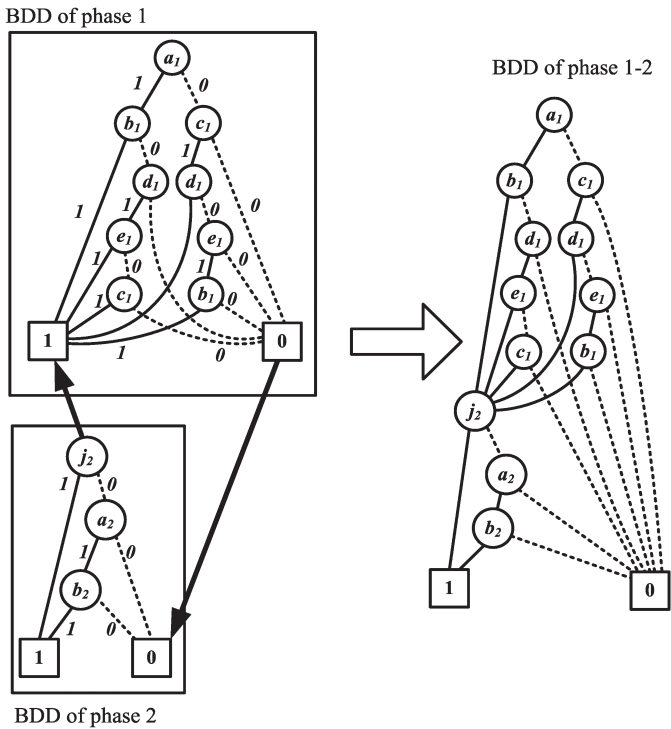


Fig. 5. Generation of BDD for phase 1-2

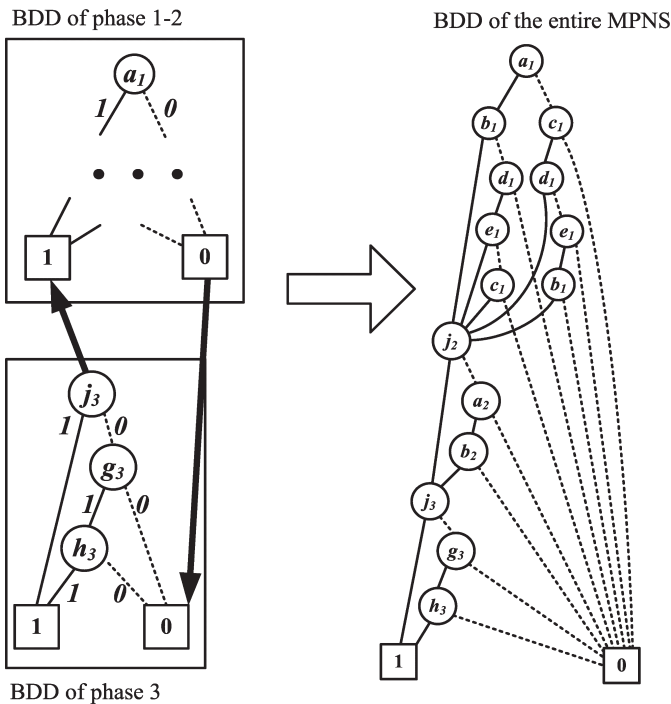


Fig. 6. Generation of the final BDD for the entire MPNS

Step 2: Reliability evaluation after the cross-phase BDD is generated

As mentioned in section 2, there are two algorithms available to the BDD-based evaluation. One is the bottom-up algorithm which traverses all BDD nodes, while the other one is the top-down algorithm which enumerates all BDD paths. Here, we use the top-down algorithm because it is applicable to a wide range of BDD (i.e., it does

not require that latter-phase nodes are placed higher than former-phase nodes.).

In order to evaluate MPNS reliability, we first find 20 paths from node a_1 to $\bar{1}$ ($20 = 5 \times 2 \times 2$; 5 paths for phase 1; and 2 paths for phase 2-3). And then the system reliability is the sum of probabilities of these 20 paths, i.e.:

$$R_{MPNS} = \sum_{m=1}^{20} \Pr\{path_m\} \tag{4}$$

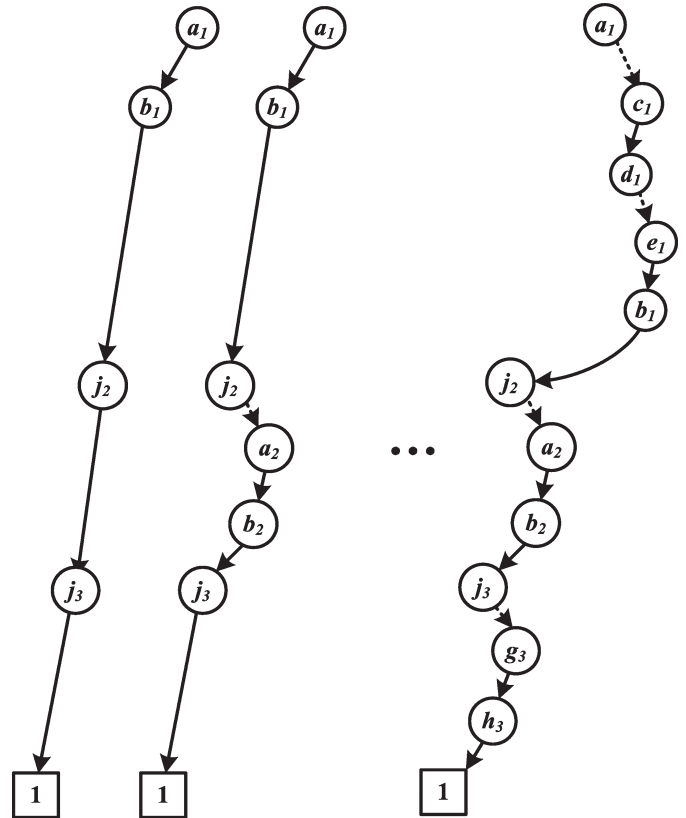


Fig. 7. Twenty paths connecting the top node with the bottom node

In Eq. (4), $\Pr\{path_m\}$ is the probability of the m^{th} BDD path. For instance, the $path_2$ ($a_1 \xrightarrow{1} b_1 \xrightarrow{1} j_2 \xrightarrow{0} a_2 \xrightarrow{1} b_2 \xrightarrow{1} j_3 \xrightarrow{1} \bar{1}$) indicates that edges a and b should keep operational during phase 1 and phase 2; meanwhile j fails and is repaired during phase 2, and then functions successfully during phase 3. By assuming edges are independent we have:

$$\Pr\{path_2\} = \Pr\{a_1 a_2\} \cdot \Pr\{b_1 b_2\} \cdot \Pr\{\bar{j}_2 j_3\} \tag{5}$$

Eq. (5) is easy to calculate when specific probability distribution functions of edges are provided. However, calculation based on Eq. will be complex when the number of phases increases and edges become repairable. In the next section we present the component-behavior model to solve this problem.

4. Method Expansion

In this section, we show that some new techniques can be inserted into the proposed method to consider various engineering problems. The first challenge presented in this section is the multi-phase repair

of network components. The second challenge is the failure occurred in network vertices. Both problems are common in engineering applications.

4.1. Challenge 1: repair in MPNS components

Here, we present a component-behavior model [24] to allow for the complex multi-phase repair/failure activities of components, based on the following assumptions.

(1) Repaired components are reused only in the next phase. (i.e., repaired components cannot be reused immediately until the next phase begins.)

(2) Components' life and repair time are independent variables of exponential distributions.

The component-behavior model is valuable because the traditional approach is too complicated to analyze multi-phase repair. For instance, consider the BDD path $\bar{j}_1\bar{j}_2\bar{j}_3j_4$ which indicates that the state of j is down sometime during phase 1 (the same happens in phase 2 and 3), but the state of j is always up during phase 4. Traditional methods usually need to discuss $\Pr\{\bar{j}_1\bar{j}_2\bar{j}_3j_4\}$ in a way like Eq. (6).

$$\Pr\{\bar{j}_1\bar{j}_2\bar{j}_3j_4\} = \Pr\{X\}\Pr\{Y\}\Pr\{Z\} \cdot \Pr\{\bar{j}_1\bar{j}_2\bar{j}_3j_4|XYZ\} + \Pr\{\bar{X}\}\Pr\{\bar{Y}\}\Pr\{Z\} \cdot \Pr\{\bar{j}_1\bar{j}_2\bar{j}_3j_4|\bar{X}\bar{Y}Z\} + \Pr\{X\}\Pr\{\bar{Y}\}\Pr\{Z\} \cdot \Pr\{\bar{j}_1\bar{j}_2\bar{j}_3j_4|X\bar{Y}Z\} + \Pr\{\bar{X}\}\Pr\{Y\}\Pr\{Z\} \cdot \Pr\{\bar{j}_1\bar{j}_2\bar{j}_3j_4|\bar{X}YZ\} \quad (6)$$

where events X, Y, Z represent that j is repaired during phase 1, 2, 3, respectively. Events $\bar{X}, \bar{Y}, \bar{Z}$ represent that j is not repaired (after failure) during phase 1, 2, 3, respectively. The event $\bar{j}_1\bar{j}_2\bar{j}_3j_4|\bar{X}\bar{Y}Z$ represents that the state of j is never repaired during phase 1 and 2; in phase 3 j is repaired sometime but fails again; finally j stays operational in phase 4. Apparently such decomposition is too cumbersome to implement.

In order to provide a concise approach, we propose the "component-behavior model" [24] which is effective in analyzing various behavior of components. Take the system in Fig. 2 for instance, matrices $U_i^{(k)}$ and $D_i^{(k)}$ (represent k is up and down in phase i , respectively) replace the constant edge reliability (0.9), as shown in Fig. 8.

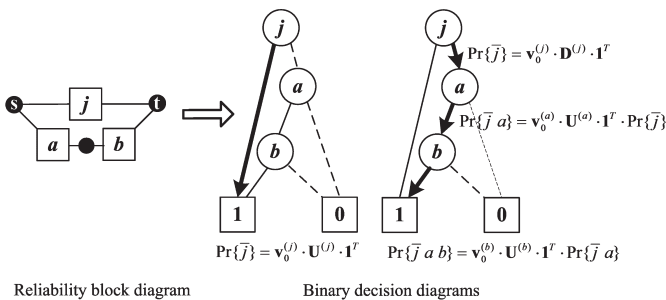


Fig. 8. Component-behavior model integrated with BDD

In Fig. 8, the initial state vector $v_0^{(k)} = (1,0)$ for initially operational binary-state edge k . The column vector $1^T = (1,1)^T$. In Fig. 8 we use a matrix ($C_i^{(k)}$) to represent the behavior of certain network edge k during phase i . i.e.:

$$C_i^{(k)} = \begin{cases} E_i^{(k)}, & \text{if the state of } k \text{ is either operational or down in phase } i \\ U_i^{(k)}, & \text{if component } k \text{ stays operational during phase } i \\ D_i^{(k)}, & \text{if component } k \text{ is down sometime in phase } i \\ R_i^{(k)}, & \text{if component } k \text{ continues repairs in phase } i \\ I, & \text{if the state of } k \text{ remains unchanged in phase } i \end{cases} \quad (7)$$

In Eq., the expression of $E_i^{(k)}, \dots, R_i^{(k)}$ can be found in [24]. For instance, $U_i^{(k)}, E_i^{(k)}$, and $D_i^{(k)}$ are of the form:

$$U_i^{(k)} = \exp\begin{bmatrix} -\lambda_i^k & \lambda_i^k \\ 0 & 0 \end{bmatrix} \cdot T_i, E_i^{(k)} = \exp\begin{bmatrix} -\lambda_i^k & \lambda_i^k \\ \mu_i^k & -\mu_i^k \end{bmatrix} \cdot T_i, D_i^{(k)} = E_i^{(k)} - U_i^{(k)} \quad (8)$$

where λ_i^k is the failure rate of k during phase i (μ_i^k is the repair rate); T_i is the duration of phase i . By applying the evaluation algorithm (Fig. 8) to the cross-phase BDD of MPNS, we can evaluate the reliability of MPNS with repairable edges easily. It should be noted that the method in this section is effective even when components becomes non-repairable (At this case let $\mu_i^k = 0$).

For instance, reconsider the BDD path $a_1b_1\bar{j}_2a_2b_2j_3$ which is mention in Eq. (see Section 3). We have:

$$\begin{aligned} \Pr\{a_1a_2\} &= v_0^{(a)} \cdot U_1^{(a)} \cdot U_2^{(a)} \cdot E_3^{(a)} \cdot 1^T \\ \Pr\{b_1b_2\} &= v_0^{(b)} \cdot U_1^{(b)} \cdot U_2^{(b)} \cdot E_3^{(b)} \cdot 1^T \\ \Pr\{\bar{j}_2j_3\} &= v_0^{(j)} \cdot E_1^{(j)} \cdot D_2^{(j)} \cdot U_3^{(j)} \cdot 1^T \end{aligned} \quad (9)$$

The probability of $a_1b_1\bar{j}_2a_2b_2j_3$ can be expressed as:

$$\Pr\{a_1b_1\bar{j}_2a_2b_2j_3\} = (v_0^{(a)}U_1^{(a)}U_2^{(a)}E_3^{(a)}1^T) \cdot (v_0^{(b)}U_1^{(b)}U_2^{(b)}E_3^{(b)}1^T) \cdot (v_0^{(j)}E_1^{(j)}D_2^{(j)}U_3^{(j)}1^T) \quad (10)$$

$$\begin{aligned} a_1 &\xrightarrow{\text{up}} U_1^{(a)}; a_2 \xrightarrow{\text{up}} U_2^{(a)}; a_3 \xrightarrow{\text{not specified}} E_3^{(a)}; \\ b_1 &\xrightarrow{\text{up}} U_1^{(b)}; b_2 \xrightarrow{\text{up}} U_2^{(b)}; b_3 \xrightarrow{\text{not specified}} E_3^{(b)}; \\ j_1 &\xrightarrow{\text{Not specified}} E_1^{(j)}; j_2 \xrightarrow{\text{down}} D_2^{(j)}; j_3 \xrightarrow{\text{up}} U_3^{(j)}; \end{aligned}$$

Fig. 9. BDD path interpretation according to the component-behavior model

In fact, $a_1b_1\bar{j}_2a_2b_2j_3$ does not clarify the behavior of the edge j during the first phase. Hence in Eq. the matrix $E_1^{(j)}$ is used to represent that j may fail even if it is not used during phase 1. Conversely we can replace $E_1^{(j)}$ with the identity matrix I based on the assumption that j will maintain its initial state (operational) during phase 1. In this case Eq. (11) becomes:

$$\Pr\{a_1b_1\bar{j}_2a_2b_2j_3\} = (v_0^{(a)}U_1^{(a)}U_2^{(a)}E_3^{(a)}1^T) \cdot (v_0^{(b)}U_1^{(b)}U_2^{(b)}E_3^{(b)}1^T) \cdot (v_0^{(j)} \cdot I \cdot D_2^{(j)}U_3^{(j)}1^T) \quad (11)$$

From Fig. 9 we can see it is easy to interpret BDD paths into corresponding matrices, making it practical to implement in computer

languages. By summing the probabilities of all BDD paths we can obtain the reliability of MPNS, i.e.:

$$R_{MPNS} = \sum_{m=1}^s \Pr\{path_m\} \quad (12)$$

where s is the number of paths linking the top node with the bottom node $\bar{1}$ in the cross-phase BDD. In summary, the algorithm proposed in this section can be illustrated by Fig. 10.

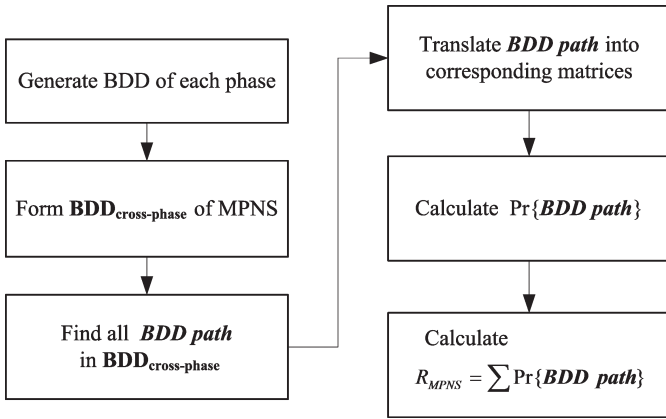


Fig. 10. BDD and component-behavior combined method for MPNS reliability analysis

4.2. Challenge 2: unreliable vertices

In many realistic engineering applications, network vertices may fail as well as edges. These vertices are termed as unreliable vertices (or imperfect vertices). In literature we can find some researches [22, 26, 40, 39] allowing for unreliable vertices. Kuo et. al. [22] replace edges with the entities which combines both edges and vertices, and use the entities to build BDD. This approach is called ‘‘Incident edge substitution’’. In factoring-theorem algorithms, unreliable vertices are taken into consideration by factoring on vertices as well as on edges. In our approach, both vertices and edges are considered as BDD variables, and meanwhile the BDD generation algorithm [44] is applied to both of them.

Consider the MPNS in Fig. 3, the BDD generation technique [22] is applied to both networks edges and vertices, resulting in BDD of Fig. 11. Next, three BDD are combined one-by-one as we did in Section 3.1 to generate a final cross-phase BDD of MPNS. Based on the cross-phase BDD we can evaluate the reliability of MPNS. In the

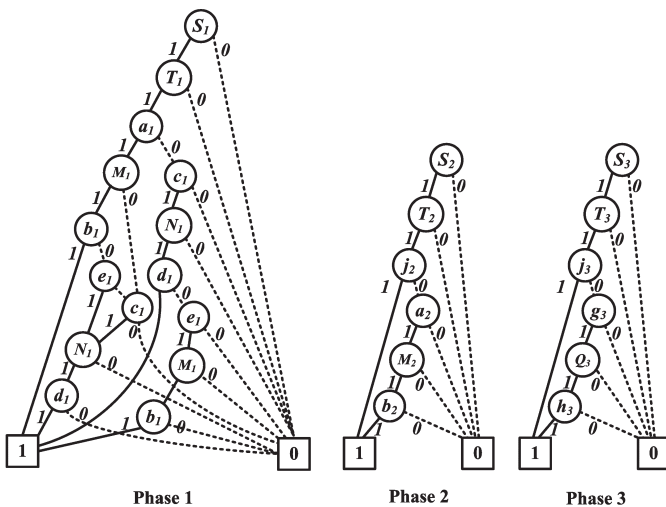


Fig. 11. Network BDD considering vertices & links failure

next section we propose two practical cases which consider repairable components and unreliable vertices, respectively.

5. Case study

5.1. Satellite telemetry, tracking & control (TT&C) mission

In the satellite telemetry, tracking & control (TT&C) mission, high reliability is one of the top concern for communication-system designers. To ensure reliable communications, some on-orbit satellites usually act as the repeater between the source ground station and the target satellite. For instance in Fig. 12, a relay satellite Sat_1 may retransmit signals when the target Sat_2 is not directly visible to the ground-station facility Fac_1 .

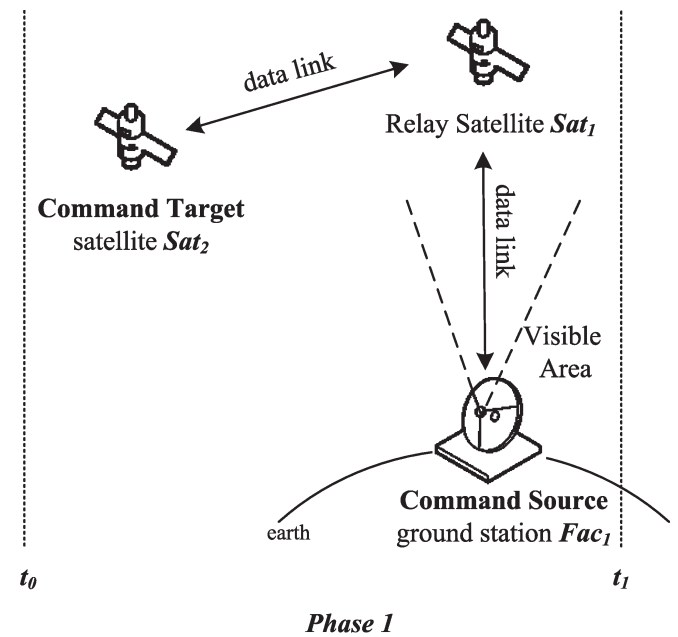


Fig. 12. Relay satellite between the source ground facility and the target satellite

For practical satellite navigation systems, there exist several data-relay alternatives. By considering relay satellites as vertices, the communication scheme can be interpreted as a network which is shown in Fig. 13.

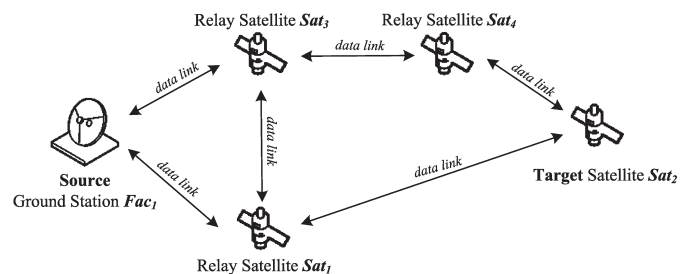


Fig. 13. Relay satellites and combination links in a certain phase

As different non-geostationary satellites fly over the source ground station. Data links in the previous phase may become invalid as the original relay satellites are no longer invisible to the ground facility. Hence it leads to the changing communication structures in different phases, as shown in Fig. 14. This changing system structure can be extracted as a MPNS model (see Fig. 15).

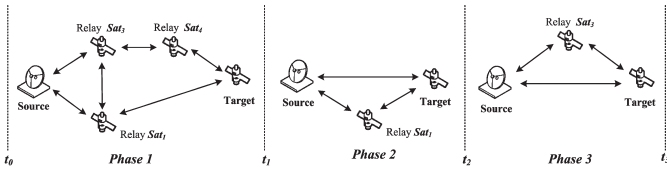


Fig. 14. Changing communication links in the spaceflight TT&C mission

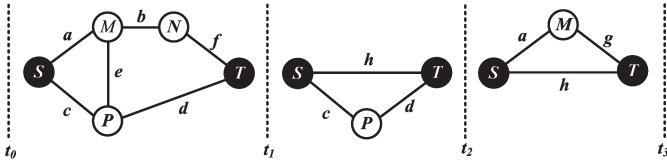


Fig. 15. MPNS of the spaceflight TT&C mission

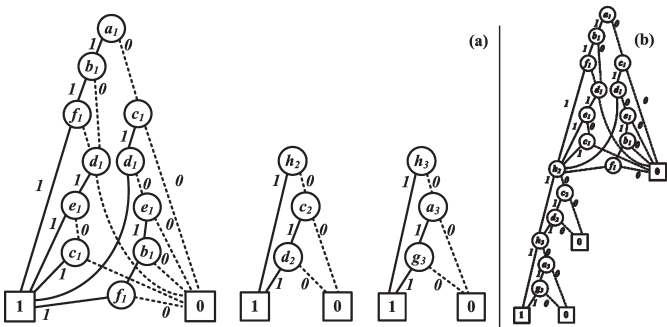


Fig. 16. BDD for the satellite TT&C systems

In order to evaluate the reliability of MPNS, we generate BDD (Fig. 16(a)) for different phases according to the algorithm in Section 3 [44]. These BDD are combined together to form a cross-phase BDD of MPNS (Fig. 16(b)). The reliability of MPNS is evaluated by the top-down algorithm. For the cross-phase BDD, there are 28 vectors which correspond to 28 paths from a_1 to $\bar{1}$. (7 paths for phase 1; 2 paths for phase 2, and 2 paths for phase 3; we have $28 = 7 \times 2 \times 2$ paths for the cross-phase BDD.)

For instance, the probability of $path_1$ ($a_1 \xrightarrow{1} b_1 \xrightarrow{1} f_1 \xrightarrow{1} h_2 \xrightarrow{1} h_3 \xrightarrow{1} \bar{1}$) is:

$$\Pr\{path_1\} = (v_0^{(a)} \cdot U_1^{(a)} \cdot [1,1]) \cdot (v_0^{(b)} \cdot U_1^{(b)} \cdot [1,1]) \cdot (v_0^{(c)} \cdot E_1^{(c)} \cdot [1,1]) \cdot (v_0^{(d)} \cdot E_1^{(d)} \cdot [1,1]) \cdot (v_0^{(e)} \cdot E_1^{(e)} \cdot [1,1]) \cdot (v_0^{(f)} \cdot U_1^{(f)} \cdot [1,1]) \cdot (v_0^{(g)} \cdot E_1^{(g)} \cdot [1,1]) \cdot (v_0^{(h)} \cdot I_1^{(h)} \cdot U_2^{(h)} \cdot U_3^{(h)} \cdot [1,1]) \quad (13)$$

where $U_1^{(a)}$ indicates that edge a keeps operational during phase 1.

$E_1^{(c)}$ indicates that the behavior of c in phase 1 is either up or down.

$I_1^{(h)} U_2^{(h)} U_3^{(h)}$ indicates that the state of h remains unchanged during phase 1, and keeps operational during phase 2 and phase 3. The MPNS reliability is the sum of all $\Pr\{path_i\}$, i.e.:

$$R_{MPNS} = \sum_i \Pr\{path_i\} \quad (14)$$

In order to verify the above algorithm, we compare results of our method and that of the simulation method under two scenarios. The first scenario considers all communication links as nonrepairable and their life are independent variables of exponential distributions, failure rates $[\lambda_a, \lambda_b, \dots, \lambda_h] = [0.2, 0.22, 0.24, \dots, 0.34]$ (unit: probability per hour), and failure rates do not change across phases. Let

$[T_1, T_2, T_3] = [1, 1, 1]$ (hour) be the duration of phase 1-3. System reliability calculated by our approach and the simulation approach are shown in Table. 1.

Figures in Table. 1 show that our result is very close to the simulation result. The Monte Carlo simulation is carried out with the software of Windchill Quality Solutions (formerly Relx) [42]. The number of simulation iteration is 10^6 . Additionally, results of Monte Carlo simulation is very close to the Petri-net simulation results of the GRIF software [13]. From the 5th row data of Table. 1 we can see that the proposed method is able to detect the sudden reliability drop at the phase-transition moment.

Table 1. Reliability of MPNS with nonrepairable links

Time	System Reliability	
	Proposed method	Monte Carlo simulation
0 (Start of phase 1)	1	1
1 (End of phase 1)	0.85400	0.85394
1.001 (Start of phase 2)	0.85399	0.85390
2 (End of phase 2)	0.71389	0.71386
2.001 (Start of phase 3)	0.69466	0.69458
3 (End of phase 3)	0.57323	0.57334

In the second scenario, we suppose all communication links are repairable (exponential distributed repair time) and their repair rates $[\mu_a, \mu_b, \dots, \mu_h] = [0.4, 0.42, 0.44, \dots, 0.54]$ (μ_i do not change across phases, and other parameters equals to that in the scenario 1.). Figures in Table. 2 show that results of our method (1st column data) are slightly different from that of the simulation method (3rd column data).

Table 2. Reliability of MPNS with repairable links

Time	System Reliability		
	Proposed method (Original system)	Proposed method (more accurate) (Divide 1 phase into 4 phases)	Monte Carlo simulation
0 (Start of phase 1)	1	1	1
1 (End of phase 1)	0.8540	0.8656	0.8715
2 (End of phase 2)	0.7213	0.7415	0.7510
3 (End of phase 3)	0.5905	0.6164	0.6231

When the system contains repairable components, the proposed method generates accurate results only on the assumption that repaired units can be reused only in the next phase (cannot be reused immediately until next phase). Without this assumption our method only provides an approximate result (i.e., data differences between 1st column and 3rd column in Table 2). In Monte Carlo simulation, no such assumption exists and thus simulation approach generates a more accurate (higher) result than our method. Fortunately the impact of our assumption can be diminished by dividing one mission phase into multiple ones. We divide one phase into four phases and obtain more accurate results which is shown in the 2nd column of Table. 2. Figures in the 2nd column are close to the simulation results, partly demonstrating the effectiveness of the proposed method in analyzing repairable components.

5.2. Speed-monitoring mission

In railway systems, speed detection has become an indispensable security measure to avoid clashes and delay. A common speed-monitoring equipment is the radar speedometer which uses the Doppler effect to analyze the speed of trains. In addition, laser speedometers are normally installed near railway stations as an alternative to supervise the incoming train. Consider a train P scheduled to move across Station α , β , and γ , as shown in Fig. 17. Radar speedometers M and N measure the speed of the train and directly transmit data to the control center Q . In addition, laser sensors $A \sim F$ measure the speed when P approaches railway stations. Data from $A \sim F$ can be directly transmitted to Q , or gathered at specific sensors to be transmitted together.

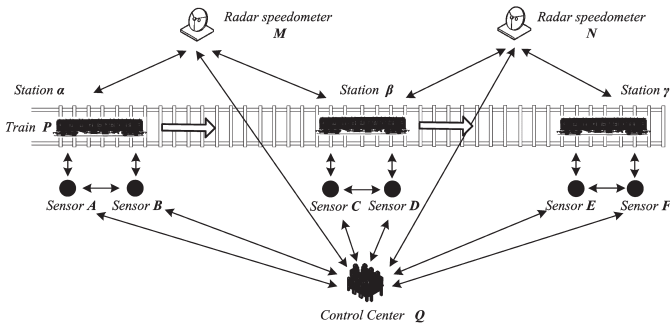


Fig. 17. Train-speed monitoring mission.

Our aim is to analyze the reliability of the speed-monitoring system when the train approaches stations. Apparently MPNS is an effective description for the system. We decompose the mission into three phases to obtain a MPNS which is shown in

Fig. 18. Suppose both edges and vertices are prone to failure (except P and Q), the proposed algorithm generates the BDD for each phase (Fig. 19 considers uMv or sNt as an entity) and combines them together to form a cross-phase BDD. The system reliability is the sum of probabilities of all BDD paths. (There are $392 = 7 \times 8 \times 7$ paths in the cross-phase BDD; 7 paths for phase 1; 8 path for phase 2; and 7 paths for phase 3)

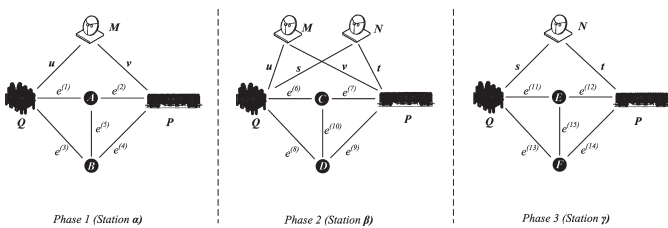


Fig. 18. MPNS model for the train-speed monitoring system.

By setting the following parameters we can compare our results with simulation results (generated by Windchill [42] and GRIF software [13]), as shown in Table. 3.

- (1) Failure rates of all components $\lambda = 0.1$ (failure rates remain unchanged across phases; life of components are exponentially distributed independent variables.);
- (2) All components are non-repairable;
- (3) Phase duration $[T_1, T_2, T_3] = [1, 1, 1]$ hour.

From cases in this section it can be seen that MPNS is a useful tool to model some communication projects. Figures in Table. 3 show that the proposed evaluation method can calculate an accurate reliability of multi-phase networks.

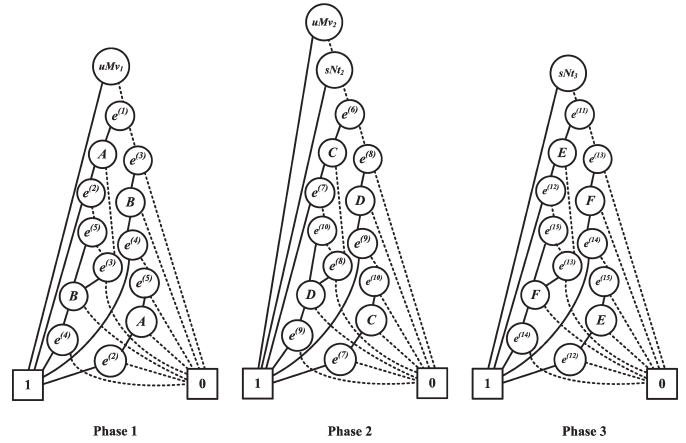


Fig. 19. BDD of each phase in MPNS

Table 3. Reliability of the speed-monitoring system

Time	System Reliability	
	Proposed method	Monte Carlo simulation
0 (Start of phase 1)	1	1
1 (End of phase 1)	0.98543	0.98544
2 (End of phase 2)	0.97908	0.97909
3 (End of phase 3)	0.95445	0.95449

6. Conclusion

The paper proposes an BDD-based algorithm to evaluate the reliability of MPNS. From literatures we find existing methods do not take dynamic characteristics of networks into consideration. Hence this paper introduces the concept of MPNS to consider some dynamic characteristics which include the varying system configuration, changing components' failure rates, and repair activities during the mission. From case study it can be seen that MPNS is the effective description of many communication systems.

The reliability of MPNS is evaluated with the cross-phase BDD which is generated by merging BDD of different phases. When the number of phase in MPNS increases, the merging step is easier to implement in computers compared to the backward ordering in [49]. Another strength of the BDD-based approach is that it can be further expanded to consider repairable edges and failure in vertices.

Future efforts will focus on the k-terminal reliability analysis of MPNS, and the reliability analysis of directed MPNS with CCF. Additionally, truncation of the cross-phase BDD can also be taken into consideration to make our approach applicable to large MPNS.

Acknowledgement

Authors express their gratitude to the reviewer's comment about the digraph (and Signed Directed Graphs) based methods. This research is partly carried out with the Reliability, Availability, Maintainability and Safety (RAMS) group at the Norwegian University of Science and Technology (NTNU), and it is funded by the National Natural Science Foundation of China with the grant no. 71071159.

References

1. Aggarwal KK, Chopra YC, Bajwa JS. Modification of cutsets for reliability evaluation of communication systems. *Microelectronics Reliability* 1982; 22(3): 337-340, [http://dx.doi.org/10.1016/0026-2714\(82\)90005-1](http://dx.doi.org/10.1016/0026-2714(82)90005-1).
2. Ahmad SH. Simple enumeration of minimal cutsets of acyclic directed graph. *IEEE Transactions on Reliability* 1988; 37(5): 484-487, <http://dx.doi.org/10.1109/24.9868>.
3. Alam M, Alsaggaf UM. Quantitative reliability evaluation of repairable phased-mission systems using Markov approach. *IEEE Transactions on Reliability* 1986; 35(5): 498-503, <http://dx.doi.org/10.1109/TR.1986.4335529>.
4. Bartlett LM, Hurdle EE, Kelly EM. Integrated system fault diagnostics utilising digraph and fault tree-based approaches. *Reliability Engineering and System Safety* 2009; 94(6): 1107-1115, <http://dx.doi.org/10.1016/j.res.2008.12.005>.
5. Bartlett LM, Hurdle EE, Kelly EM. Comparison of digraph and fault tree based approaches for system fault diagnostics. *Proceedings of the European Safety and Reliability Conference (ESREL) 2006*; 1: 191-198.
6. Bollig B, Wegener I. Improving the variable ordering of OBDDs is NP-complete. *IEEE Transactions on Computers* 1996; 45(9): 993-1002, <http://dx.doi.org/10.1109/12.537122>.
7. Bouissou M, Bruyere F, Rauzy A. BDD based fault-tree processing: a comparison of variable ordering heuristics. *Proceedings of European Safety and Reliability Association Conference (ESREL) 1997*, <http://dx.doi.org/10.1016/b978-008042835-2/50231-9>.
8. Choi MS, Jun CH. Some variants of polygon-to-chain reductions in evaluating reliability of undirected network. *Microelectronics Reliability* 1995; 35(1): 1-11, [http://dx.doi.org/10.1016/0026-2714\(94\)P1833-X](http://dx.doi.org/10.1016/0026-2714(94)P1833-X).
9. Colbourn CJ, *The combinatorics of network reliability*, Oxford University Press: London, 1987.
10. Cook JL, Ramirez-Marquez JE. Two-terminal reliability analyses for a mobile ad hoc wireless network. *Reliability Engineering and System Safety* 2007; 92(6): 821-829, <http://dx.doi.org/10.1016/j.res.2006.04.021>.
11. Friedman SJ, Supowit KJ. Finding the optimal variable ordering for binary decision diagrams. *Proceedings of the 24th ACM/IEEE Design Automation Conference 1987*: 348-356, <http://dx.doi.org/10.1145/37888.37941>.
12. Gadani J. System effectiveness evaluation using star and delta transformations. *IEEE Transactions on Reliability* 1981; 30(1): 43-47, <http://dx.doi.org/10.1109/TR.1981.5220959>.
13. GRIF-Workshop Software, SATODEV LLC 2014; <http://grif-workshop.com/>.
14. Hardy G, Lucet C, Limnios N. Computing all-terminal reliability of stochastic networks with binary decision diagrams. *11th International Symposium on Applied Stochastic Models and Data Analysis 2005*: 17-20.
15. Hardy G, Lucet C, Limnios N. K-terminal network reliability measures with binary decision diagrams. *IEEE Transactions on Reliability* 2007; 56(3): 506-515, <http://dx.doi.org/10.1109/TR.2007.898572>.
16. Hariri S, Raghavendra CS. SYREL: A symbolic reliability algorithm based on path and cutset methods. *IEEE Transactions on Computers* 1987; 100(10): 1224-1232, <http://dx.doi.org/10.1109/TC.1987.1676862>.
17. Huang HZ, Zhang H, Li Y. A new ordering method of basic events in fault tree analysis. *Quality and Reliability Engineering International* 2012; 28(3): 297-305, <http://dx.doi.org/10.1002/qre.1245>.
18. Kelly EM, Bartlett LM. Enhanced diagnosis of faults using the digraph approach applied to a dynamic aircraft fuel system. *Proceedings of the first international conference on availability, reliability and security (ARES) 2006*.
19. Kelly EM, Bartlett LM. Application of the digraph method in system fault diagnostics. *Proceedings of the first international conference on availability, reliability and security (ARES) 2006*, <http://dx.doi.org/10.1109/ARES.2006.31>.
20. Kim K, Park KS. Phased-mission system reliability under Markov environment. *IEEE Transactions on Reliability* 1994; 43(2): 301-309, <http://dx.doi.org/10.1109/24.295013>.
21. Kuo SY, Lu SK, Yeh FM. Determining terminal-pair reliability based on edge expansion diagrams using OBDD. *IEEE Transactions on Reliability* 1999; 48(3): 234-246, <http://dx.doi.org/10.1109/24.799845>.
22. Kuo SY, Yeh FM, Lin HY. Efficient and exact reliability evaluation for networks with imperfect vertices. *IEEE Transactions on Reliability* 2007; 56(2): 288-300, <http://dx.doi.org/10.1109/TR.2007.896770>.
23. Locks MO. A minimizing algorithm for sum of disjoint products. *IEEE Transactions on Reliability* 1987; 36(4): 445-453, <http://dx.doi.org/10.1109/TR.1987.5222436>.
24. Lu JM, Wu XY. Reliability evaluation of generalized phased-mission systems with repairable components. *Reliability Engineering and System Safety* 2014; 121: 136-145, <http://dx.doi.org/10.1016/j.res.2013.08.005>.
25. Lu JM, Wu XY, Liu Y, Mary Ann Lundteigen. Reliability analysis of large phased-mission systems with repairable components based on success-state sampling. *Reliability Engineering and System Safety* 2015; 142: 123-133, <http://dx.doi.org/10.1016/j.res.2015.05.010>.
26. Netes VA, Filin BP. Consideration of node failures in network-reliability calculation. *IEEE Transactions on Reliability* 1996; 45(1): 127-128, <http://dx.doi.org/10.1109/24.488928>.
27. Page LB, Perry JE. Reliability of directed networks using the factoring theorem. *IEEE Transactions on Reliability* 1989; 38(5): 556-562, <http://dx.doi.org/10.1109/24.46479>.
28. Rai S, Kumar A, Prasad EV. Computing terminal reliability of computer network. *Reliability Engineering* 1986; 16(2): 109-119, [http://dx.doi.org/10.1016/0143-8174\(86\)90079-X](http://dx.doi.org/10.1016/0143-8174(86)90079-X).
29. Rauzy A. Binary decision diagrams for reliability studies. *Handbook of Performability Engineering*. Springer: London, 2008: 381-396.
30. Resende LI. Implementation of a factoring algorithm for reliability evaluation of undirected networks. *IEEE Transactions on Reliability* 1988; 37(5): 462-468, <http://dx.doi.org/10.1109/24.9862>.
31. Resende MG. A program for reliability evaluation of undirected networks via polygon-to-chain reductions. *IEEE Transactions on Reliability*; 1986; 35(1): 24-29, <http://dx.doi.org/10.1109/TR.1986.4335334>.
32. Satyanarayana A, Chang MK. Network reliability and the factoring theorem. *Networks* 1983; 13(1): 107-120, <http://dx.doi.org/10.1002/net.3230130107>.
33. Singh H, Vaithilingam S, Anne RK. Terminal reliability using binary decision diagrams. *Microelectronics Reliability* 1996; 36(3): 363-365, [http://dx.doi.org/10.1016/0026-2714\(95\)00087-9](http://dx.doi.org/10.1016/0026-2714(95)00087-9).

34. Smotherman MK, Geist RM. Phased mission effectiveness using a nonhomogeneous Markov reward model. *Reliability Engineering and System Safety* 1990; 27(2): 241-255, [http://dx.doi.org/10.1016/0951-8320\(90\)90057-T](http://dx.doi.org/10.1016/0951-8320(90)90057-T).
35. Smotherman MK, Zemoudeh K. A non-homogeneous Markov model for phased-mission reliability analysis. *IEEE Transactions on Reliability* 1989; 38(5): 585-590, <http://dx.doi.org/10.1109/24.46486>.
36. Somani AK, Trivedi KS. Phased-mission system analysis using Boolean algebraic methods. *Proceedings of ACM/SIGMETRICS conference on Measurement and modeling of computer systems* 1994: 98-107, <http://dx.doi.org/10.1145/183019.183029>.
37. Tang ZH, Dugan JB. BDD-based reliability analysis of phased-mission systems with multimode failures. *IEEE Transactions on Reliability* 2006; 55(2): 350-360, <http://dx.doi.org/10.1109/TR.2006.874941>.
38. Theologou OR, Carlier JG. Factoring and reductions for networks with imperfect vertices. *IEEE Transactions on Reliability* 1991; 40(2): 210-217, <http://dx.doi.org/10.1109/24.87131>.
39. Torrieri D. Calculation of node-pair reliability in large networks with unreliable nodes. *IEEE Transactions on Reliability* 1994; 43(3): 375-377, <http://dx.doi.org/10.1109/24.326428>.
40. Vaurio JK. Fault tree analysis of phased mission systems with repairable and non-repairable components. *Reliability Engineering and System Safety* 2001; 74(2): 169-180, [http://dx.doi.org/10.1016/S0951-8320\(01\)00075-8](http://dx.doi.org/10.1016/S0951-8320(01)00075-8).
41. Wang D, Trivedi KS. Reliability analysis of phased-mission system with independent component repairs. *IEEE Transactions on Reliability* 2007; 56(3): 540-551, <http://dx.doi.org/10.1109/TR.2007.903268>.
42. Windchill Quality Solutions (formerly Relex) Version 10 Software, PTC Product & Service Advantage LLC 2014; <http://www.ptc.com/product/windchill>.
43. Wood RK. A factoring algorithm using polygon-to-chain reductions for computing K-terminal network reliability. *Networks* 1985; 15(2): 173-190, <http://dx.doi.org/10.1002/net.3230150204>.
44. Wu XY. Reliability modeling and reliability analysis of complex correlative systems. PhD degree thesis of National University of Defense Technology China, 2000.
45. Xing L, Levitin G. BDD-based reliability evaluation of phased-mission systems with internal/ external common-cause failures. *Reliability Engineering and System Safety* 2013; 112: 145-153, <http://dx.doi.org/10.1016/j.ress.2012.12.003>.
46. Xing L. Reliability evaluation of phased-mission systems with imperfect fault coverage and common-cause failures. *IEEE Transactions on Reliability* 2007; 56(1): 58-68, <http://dx.doi.org/10.1109/TR.2006.890900>.
47. Xing L. An efficient binary-decision-diagram-based approach for network reliability and sensitivity analysis. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans* 2008; 38(1): 105-115, <http://dx.doi.org/10.1109/TSMCA.2007.909493>.
48. Xing L, Dugan JB, Morrissette BA. Efficient reliability analysis of systems with functional dependence loops. *Eksploatacja i Niezawodność – Maintenance and Reliability* 2009; 43(3): 65-69.
49. Zang X, Sun H, Trivedi KS. A BDD-based algorithm for reliability analysis of phased-mission systems. *IEEE Transactions on Reliability* 1999; 48(1): 50-60, <http://dx.doi.org/10.1109/24.765927>.
50. Zhang T, Bai GH, Guo B. Success probability model of phased mission systems with limited spares. *Eksploatacja i Niezawodność – Maintenance and Reliability* 2012; 14(1): 24-32.

Ji-Min LU

College of Information System and Management
National University of Defense Technology
De Ya Road 109, Changsha, Hunan 410073, China

Fares INNAL**Yiliu LIU****Mary Ann LUNDTEIGEN**

Department of Production and Quality Engineering
Norwegian University of Science and Technology, Valgrinda, N-7491
Trondheim, Norway

Xiao-Yue WU

College of Information System and Management
National University of Defense Technology
De Ya Road 109, Changsha, Hunan 410073, China

E-mails: jeemanlv@gmail.com, innal.fares@hotmail.fr, yiliu.liu@ntnu.no,
mary.a.lundteigen@ntnu.no, xiaoyuewucn@gmail.com
