

Network Reliability Analysis and Complexity Quantification Using Bayesian Network and Dual Representation

Dongkyu Lee

Graduate Student, Dept. of Civil and Environmental Engineering, Seoul National University, Seoul, S. Korea

Ji-Eun Byun

Lecturer, James Watt School of Engineering, University of Glasgow, Glasgow, United Kingdom

Junho Song

Professor, Dept. of Civil and Environmental Engineering, Seoul National University, Seoul, S. Korea

Kayvan Sadeghi

Associate Professor, Dept. of Statistical Science, University College London, London, United Kingdom

ABSTRACT: Critical roles of lifeline networks in modern societies, e.g., electricity or water distribution, transportation, make it essential to accurately assess their reliability. However, existing network reliability analysis (NRA) methods struggle to handle large-scale networks because the computational complexity increases rapidly with the number of components. To address such scalability issues, we propose a new NRA method that builds Bayesian network (BN) and junction tree models using the dual representation of a given network. The proposed method has three main advantages: (1) accurate and fast evaluation of the connectivity of large-scale networks; (2) straightforward implementation by existing BN algorithms; and (3) quantitative prediction of NRA complexity for various network topologies. We demonstrate the effectiveness and accuracy of the proposed method by quantifying the complexity of typical topologies and evaluating the reliability of a real-world network.

1. INTRODUCTION

Modern society operates on various lifeline networks such as transportation networks, power networks, and gas distribution networks. A mathematical *network* model represents such a system in performance or reliability analyses by sets of vertices and arcs. Vertices represent node-type components such as intersections or substations, while arcs represent line-type components such as roads or pipelines. To secure reliable operation of lifeline networks, it is critical to evaluate their reliability against potential risks, e.g., catastrophic events. While there are various definitions of network performance, this paper focuses on the connectivity of an origin-destination (O-D) pair. In other words, the system failure probability is the probability that there is no path between a given O-D pair.

To calculate the exact failure probability of a network consisting of N arcs, the probabilistic analysis needs to be performed over an N dimensional probability distribution. Since the size of such a distribution grows exponentially with the number of components, N , a complete quantification of these distributions becomes infeasible even for a moderate number of components (typically, $N \geq 30$). To address such limitations, various sampling-based methods have been proposed. However, such an approach is computationally inefficient when the failure probability is low. Another limitation of the sampling-based approach is that a new analysis should be performed to update the failure probabilities based on available information or evidence.

To overcome the limitations discussed above, we propose an efficient reliability analysis

method for network systems where the failure event is defined as the disconnection of an O-D pair of interest. The main idea is that, by employing a Bayesian network (BN), we separate the modeling of structural failures and functional failures of network components (i.e., edges and/or nodes). Thereby, the analysis complexity depends not only on the network size (i.e., the number of component events), but also on the network topology. Once a BN model is established, one can utilize existing BN inference algorithms to carry out reliability analysis.

The advantages of the proposed method are three-fold. First, the method can compute the exact failure probability of large-scale networks that were previously considered to be too large for an exact analysis. Second, existing BN algorithms that are readily available in general-purpose software programs facilitate implementation. Finally, the method enables us to quantify the complexity of network topology from the perspective of reliability analysis, which remains as an unresolved task.

In this paper, only arc failures are considered. It is noted that this is not necessarily a limitation since it is straightforward to modify the proposed method to consider node failures (Ball et al. 1995). Also, the proposed method requires that a given network has no directed cycle, which is a fundamental requirement for BNs.

The paper is organized as follows. Section 2 summarizes the background theories of BN and JT and introduces the concept of dual representation of networks. Section 3 proposes a new NRA method based on BN and dual graphs. In Section 4, the proposed method is applied to the NRA, and the network complexity is quantified in terms of the number of arcs in various network topologies. Then, a large-scale transportation network is analyzed as a numerical example to demonstrate the efficiency and usefulness of the proposed method in Section 5. Finally, the conclusions and future work are presented in Section 6.

2. BACKGROUND

2.1. Bayesian network

A BN is one of the probabilistic graphical models (PGMs) that visualize directional dependence between random variables (r.v.'s). A BN is represented by a directed acyclic graph (DAG), $G(N, E)$, where N and E denote a set of nodes that stand for r.v.'s and a set of directed edges that represent statistical or causal dependencies between a node pair, respectively. When an edge points from node N_i to node N_j , they are called *parent* and *child* nodes, respectively.

Once a BN graph is set up, each node $N_i \in N$ needs to be quantified by a probability distribution being conditioned on its parent nodes $Pa(N_i)$, i.e., $P(N_i|Pa(N_i))$. Then, the joint probability distribution $P(N)$, represented by a BN graph, becomes a product of the conditional probabilities of all nodes, i.e.,

$$P(N) = \prod_{N_i \in N} P(N_i|Pa(N_i)). \quad (1)$$

The equation above shows how a BN factorizes a full joint distribution $P(N)$ into lower-dimensional distributions $P(N_i|Pa(N_i))$, which can significantly reduce the memory required to store distributions. In other words, a BN enables efficient modeling of a high-dimensional probability distribution by visualizing conditional independence between r.v.'s.

BNs have a few limitations. As the number of parent nodes increases, the memory required to store the conditional probability $P(N_i|Pa(N_i))$ grows exponentially. In other words, given too many parent nodes, it becomes infeasible to quantify a BN. Another limitation is that a BN graph must not have any directed cycle, which limits the class of problems that can be handled by the BN methodology.

2.2. Junction tree algorithm

A junction tree (JT) is a graphical method that enables a structured way for inferring a BN model. A BN graph can be transformed into a JT graph, for which multiple general-purpose algorithms are available for computing marginal probabilities

(Barber 2012). Once a JT graph is constructed, one can perform probabilistic inference by passing *messages* (which are in the form of probability distributions) between the cliques in the tree. This message-passing process is equivalent to distributing and combining local probability information across a JT graph. After updating the message of all cliques, one can compute the marginal probability distribution of any r.v. by visiting a clique that the r.v. of interest belongs to.

JT models are advantageous for inferring BN models especially because of accessible computer programs that can handle the whole analysis process of a JT model. In this paper, we use the BRML toolkit by Barber (2012).

2.3. Dual graph

A dual representation of a network converts arcs and nodes from a primal (i.e., original) network. In other words, in a dual network, arcs in a primal network become nodes, and node pairs are connected if their corresponding arcs are directly connected in a primal network. Such alternative representation often reveals hidden properties of a network that do not appear apparent in a primal network (Porta et al. 2006).

3. PROPOSED NRA METHOD: BUILDING BAYESIAN NETWORK USING DUAL GRAPH

3.1. Procedures of the proposed method

We propose a new NRA method that utilizes BN and dual graphs. Advantages of the proposed method are two-fold: (1) the method can evaluate the reliability of networks whose exact solution was previously considered unattainable, and (2) the computational complexity of an arbitrary network can be quantified from the perspective of NRA.

Figure 1 illustrates a summary of the proposed procedure. First, one simplifies a target network by eliminating components that are not connected to an O-D pair of interest. Then, a BN graph is built by using a dual representation of the simplified network. Next, the BN graph is

transformed into a JT graph, for which a message-passing is scheduled; this can be done by employing one of the existing JT algorithms.

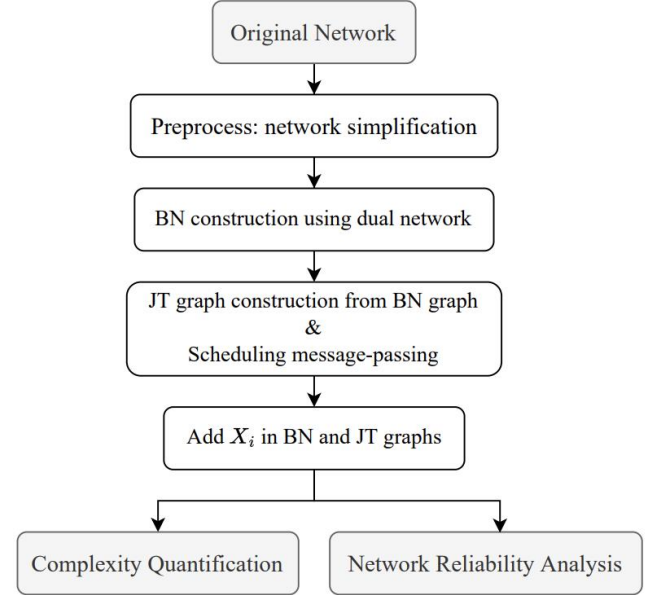


Figure 1: Flowchart of the proposed NRA method.

The final JT model can be used for two purposes: (1) quantifying the computational complexity of NRA and (2) performing NRA. The following subsections explain each step in detail.

3.2. Step 1: Preprocess–network simplification

Especially when dealing with large-scale networks, quite a large proportion of vertices may be unreachable from the origin or to the destination. Such vertices unnecessarily complicate NRA. Therefore, the proposed method first removes those vertices and the arcs connected thereto, which can be identified by any basic algorithm used for connectivity analysis. From numerical experiments, we found that this strategy greatly improves the efficiency of the proposed algorithm.

This preprocess is not mandatory. Even if this is omitted, unnecessary nodes can be eliminated through marginalization during the message-passing in a JT graph. However, with the proposed preprocess, computation becomes much more efficient than with marginalization. Especially in random networks that often have

many isolated components from an O-D pair, this strategy enables one to solve problems that seem intractable in their original forms.

3.3. Step 2: BN construction using dual graph

The method builds a BN, in which nodes represent components that can fail. Then, a directed edge is created for each pair of nodes whose corresponding arcs are connected in the target network. Since this study considers arc failures only, the resulting BN has a topology equivalent to the dual representation of a target network.

For example, consider an example network in Figure 2, which consists of 4 vertices (blue circles) and 5 directed arcs (green arrows). The origin and destination vertices are marked in the figure. Then, using the dual representation, the corresponding BN can be constructed as in Figure 3. In the BN, T_1, \dots, T_N , (in this case, $N = 5$) is a binary r.v., which takes state 1 if the head of arc i is reachable from the origin vertex and 0, otherwise. S is a binary r.v. whose state becomes 1 if the destination vertex can be reached from the origin vertex, and 0, otherwise. While conditional probability tables (CPTs) of T_i will be discussed in Section 3.5, the CPT of S is constructed as Table 1 (Byun and Song 2021a).

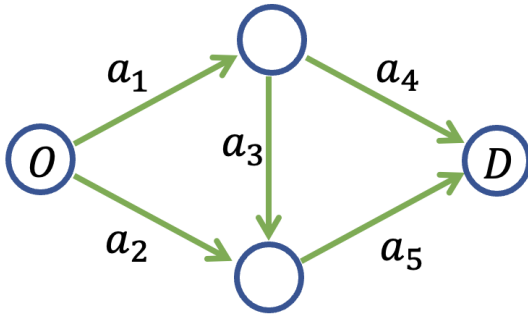


Figure 2: Example network.

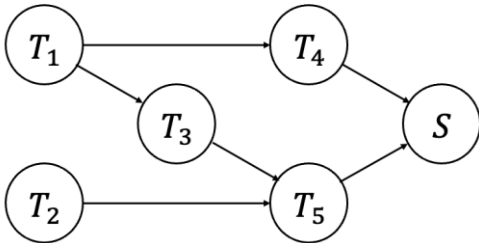


Figure 3: BN graph for the NRA of the example network.

Table 1: CPT of S given T_1, \dots, T_N .

$P(S T_1, \dots, T_N)$	$S = 1$	$S = 0$
$\sum_{T_i \in \text{Pa}(S)} T_i \geq 1$	1	0
$\sum_{T_i \in \text{Pa}(S)} T_i = 0$	0	1

The proposed method is applicable only to maximum flow analysis (for which connectivity analysis is a special case). This is because the method achieves efficiency by exploiting conditional independence between arcs that are not directly connected; that is, the connectivity status of an arc (from an origin vertex) is independent to the status of other vertices when the status of directly connected arcs is known. For instance, in the BN graph in Figure 3, the connectivity of arc 5 to the origin vertex is independent to arcs 1 and 4, being conditioned on the connectivity of arcs 2 and 3.

Such conditional independence may not hold for other types of analysis. For example, in a traffic simulation analysis, traffic is sequentially assigned by referring to traffic flows on both preceding and succeeding arcs.

3.4. Step 3: JT construction and message-passing scheduling

For probabilistic inference, a BN graph constructed in Section 3.3 can be used to build a JT graph. This can be done automatically using existing algorithms such as the maximum weight spanning tree algorithm (Barber 2012). Once a JT graph is constructed, message-passing can be scheduled, for which several algorithms can be used (Barber 2012).

For example, for the BN graph in Figure 3, a JT graph can be constructed as in Figure 4. Then, a message-passing schedule is obtained as $1 \rightarrow 3$, $2 \rightarrow 3$, and $3 \rightarrow 4$.

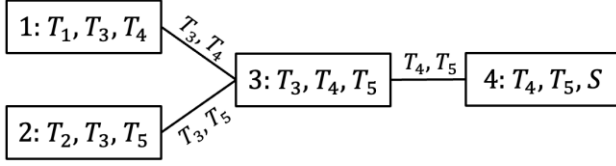


Figure 4: JT graph corresponding to Figure 3.

3.5. Step 4: Addition of component events

The nodes T_i represent a topology-based perspective of NRA. In addition, one needs to include r.v.'s to represent the states of component events, denoted as X_i , i.e., X_i represents whether arc i is functional ($X_i = 1$) or not ($X_i = 0$). By construction, X_i becomes a parent node of T_i for each $i = 1, \dots, N$. Then, the CPT of T_i can be constructed as shown in Table 2 if arc i is directly connected to the origin node. One can use CPT in Table 3 if arc i is reachable from the origin node but not directly connected to it.

Table 2: CPT of T_i given X_i .

$P(T_i X_i)$	$T_i = 1$	$T_i = 0$
$X_i = 1$	1	0
$X_i = 0$	0	1

Table 3: CPT of T_i given X_i and T_1, \dots, T_N .

$P(T_i X_i, T_1, \dots, T_N)$	$T_i = 1$	$T_i = 0$
$\left(\sum_{T_k \in Pa(T_i)} T_k \right) \cdot X_i \geq 1$	1	0
$\sum_{T_k \in Pa(T_i)} T_k = 0$	0	1
$X_i = 0$	0	1

In quantifying the CPTs of X_i , there are largely two cases: (1) component events are statistically independent and (2) dependent. In the first case, one can simply add a X_i node and an edge heading from T_i to X_i for each i . For instance, for the example network, a BN is constructed as shown in Figure 5. Then, each node X_i , $i = 1, \dots, N$, is assigned a CPT that represents $P(X_i)$. Similarly, the JT graph can be modified by simply adding a X_i to the clique(s) where T_i appears, e.g., Figure 6 for the example network.

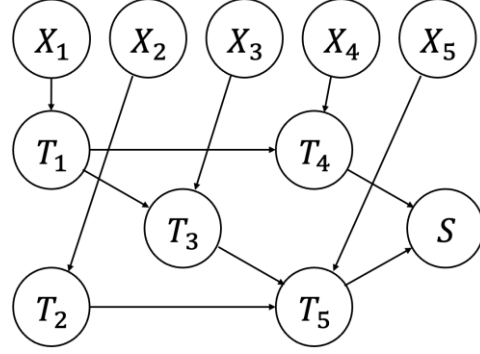


Figure 5: BN graph of the example network when component events are independent.

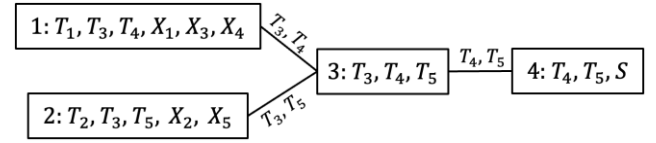


Figure 6: JT graph corresponding to Figure 5.

On the other hand, in the second case, X_i nodes are all connected to each other, and the computation becomes complicated. For example, Figure 7 illustrates a modified BN graph of the example network. In this case, it is required to quantify a single joint CPT $P(X_1, \dots, X_N)$ over all nodes X_1, \dots, X_N , whose size increases exponentially with N . Such increase in computational complexity is also observed in the modified JT graph, which becomes a single large clique that contains all nodes $X_1, \dots, X_N, T_1, \dots, T_N$, and S , as shown in the example network in Figure 8. More details of complexity quantification are discussed in Section 4.

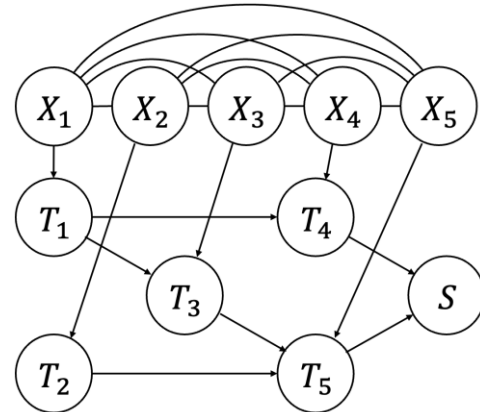


Figure 7: BN graph of the example network when component events are dependent.

1: $T_1, T_2, T_3, T_4, T_5, S, X_1, X_2, X_3, X_4, X_5$

Figure 8: JT graph corresponding to Figure 7.

4. UTILIZATION OF THE CONSTRUCTED JUNCTION TREE GRAPH

4.1. Network reliability analysis

Once a junction graph and a message-passing schedule are set up, NRA can be carried out by evaluating the marginal distribution of the system event, $P(S)$. This procedure is straightforward when component events are statistically independent (e.g., the JT in Figure 6).

On the other hand, when they are dependent (e.g., Figure 8), advanced inference strategies such as *Rao-Blackwellized particles* or *conditioning* (Koller and Friedman 2009, Byun and Song 2021b) can be employed to make the analysis affordable. The Rao-Blackwellized approach circumvents memory issues by applying sampling to a subset of r.v.'s while performing exact inference over other r.v.'s. Meanwhile, if there are common-cause variables (e.g., intensity of an earthquake), applying the conditioning technique to those variables can significantly reduce a required memory (Byun and Song 2021b). Even when there is no common-source variable, one can artificially model such variables, e.g., Bensi et al. (2011) and Song and Kang (2009).

4.2. Quantifying the complexity of NRA

A constructed JT graph can be used to quantify the computational complexity of NRA by evaluating the sum of the memory required to store the CPTs of the cliques. Specifically, the memory demanded by a clique C_j is the product of the number of states of the r.v.'s in C_j . Therefore, the required memory is proportional to 2^{N_j} where N_j is the number of r.v.'s in clique C_j . For instance, in the illustrative network, the number of probabilities to be stored is $2^3 + 2^3 + 2^3 + 2^3 = 32$ as all cliques consist of 3 components. It is noted that a required memory is governed by the largest clique.

Such utility is beneficial in that the quantification of network topology complexity

remains inconclusive. While there are several metrics developed to this end (e.g., Valiant 1979, Ball 1986), the proposed approach provides a direct metric for NRA. Before performing NRA, one can use the proposed approach to measure the complexity of a given network topology and select an appropriate NRA method. For instance, if the given topology is too complicated to apply analytical methods, one can use a sampling method or advanced BN inference algorithms.

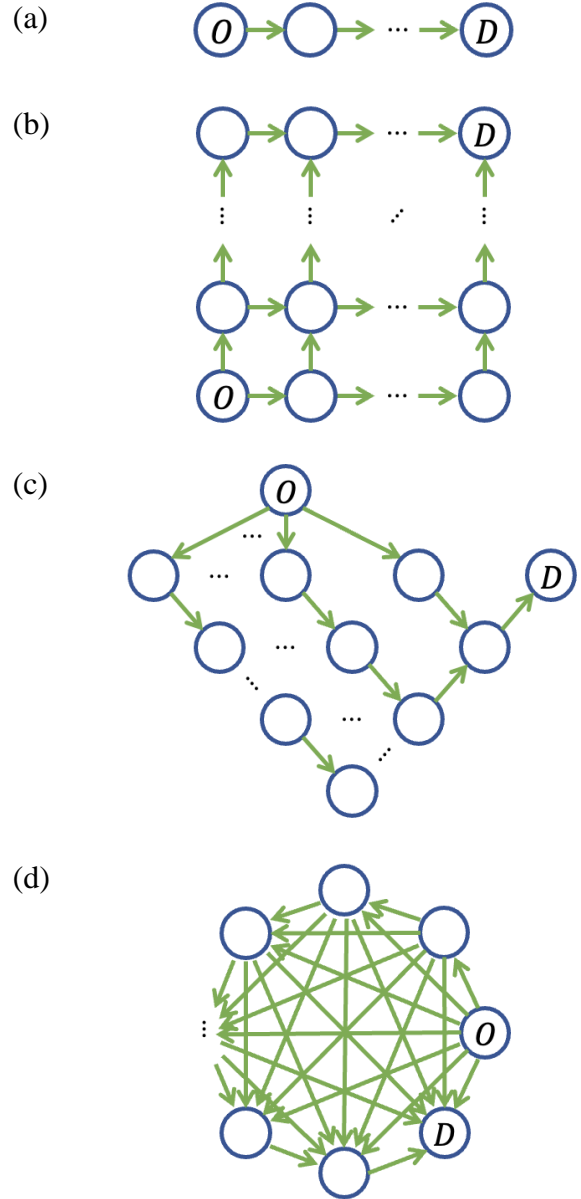


Figure 9: Typical topologies of networks: (a) line, (b) grid, (c) tree, and (d) complete.

5. NUMERICAL EXAMPLES

5.1. Complexity quantification of typical topologies

This section investigates the computational complexity of four typical network topologies: line, grid, tree, and complete networks in Figure 9.

Since the largest clique governs computational complexity, we show the largest clique size of the corresponding JT graphs in Figure 10, where the number of arcs varies from 4 to 184. In the line and tree networks, the complexity is not affected by an increasing number of arcs. On the other hand, the maximum clique size in complete networks increases linearly, which indicates that the memory demand increases exponentially. The maximum clique size of the grid structure also increases with the number of arcs, but at a much slower rate than the complete network.

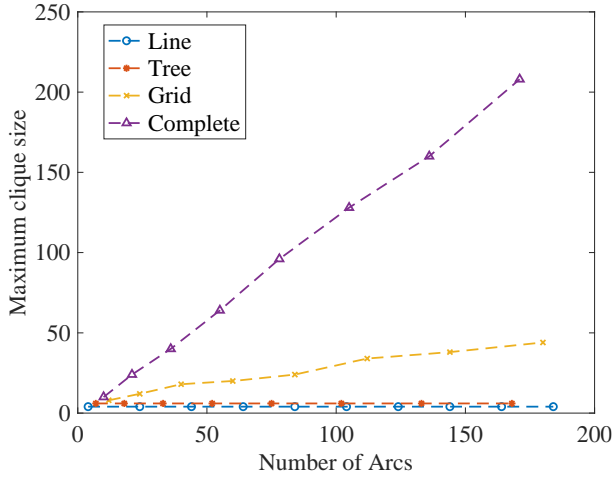


Figure 10: Maximum clique size of each topology with network size represented by number of arcs.

5.2. Eastern Massachusetts highway network

The Eastern Massachusetts (EMA) highway network consists of 129 directional arcs and 74 vertices (modified from Zhang et al. (2018)), as shown in Figure 11(a). The failure probability of each arc, $P(X_i = 0)$, is set as 0.1 for $i = 1, \dots, 129$. Using the strategy described in Section 3.2, the network is simplified to that in Figure 11(b), where the numbers of arcs and vertices are reduced to 85 and 47, respectively.

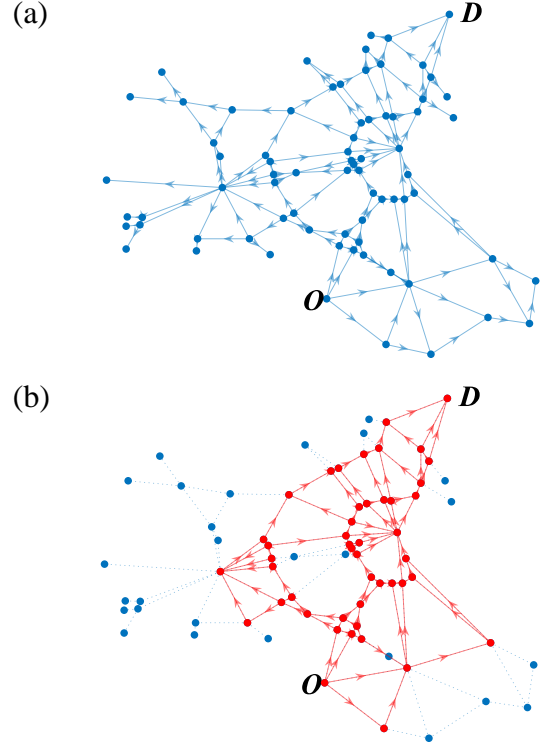


Figure 11: EMA highway network: (a) original network and (b) simplified network by preprocess.

Table 4: Analysis results of EMA highway network.

	Failure probability	Computational time (sec)
NRA using BN (w/ preprocess)	2.957%	0.525
NRA using BN (w/o preprocess)	2.957%	0.757
MCS (c.o.v. 1%)	2.952%	19.99

Considering the independent component events X_i , the maximum clique sizes of the JT graph with and without preprocessing are identical as 16. The result implies that, in this example, preprocess does not incur any difference in computational complexity although preprocessing slightly shortens the time by reducing message-passing between cliques. Table 4 shows the computational costs and the network failure probability estimates by the proposed method with and without preprocess, compared to the results of Monte Carlo simulation (MCS). The results confirm that the proposed method provides

consistent estimates of the network failure probability, while taking only about 2.6~3.8% of the computation time taken by MCS. It is noted that the result computed by the proposed method is an exact solution.

6. CONCLUSIONS

In this paper, an efficient NRA method is proposed by utilizing a Bayesian network (BN) and dual graph representation. The proposed method can assess the reliability of networks whose exact solutions cannot be obtained using existing methods. Moreover, the proposed method provides a useful metric to quantify complexity of an NRA problem, which reflects not only the number of components, but also network topology. Furthermore, the proposed method can be easily implemented by using existing BN algorithms and/or BN software programs as it is based on the well-established BN theory. The numerical examples demonstrate the performance of the proposed method, which include the Eastern Massachusetts highway network consisting of 129 arcs.

The proposed method has a few limitations, which remain as future research topics. First, the method cannot handle networks that have directed cycles. Second, the statistical dependence between component events increases computational cost exponentially with the number of network components. These issues can be addressed by developing advanced inference algorithms.

7. ACKNOWLEDGMENT

The first and third authors are supported by the National Research Foundation of Korea (NRF) Grant funded by the Korean government (NRF-2021R1A2C2003). The second author is supported by the Alexander von Humboldt Foundation.

8. REFERENCES

- Ball, M. O. (1986). Computational complexity of network reliability analysis: An overview. *Ieee transactions on reliability*, 35(3), 230-239.
- Ball, M. O., Colbourn, C. J., & Provan, J. S. (1995). Network reliability. *Handbooks in operations research and management science*, 7, 673-762.
- Barber, D. (2012). *Bayesian reasoning and machine learning*. Cambridge University Press.
- Bensi, M., Der Kiureghian, A., & Straub, D. (2011). Bayesian network modeling of correlated random variables drawn from a Gaussian random field. *Structural Safety*, 33(6), 317-332.
- Byun, J. E. & Song, J. (2021a). Generalized matrix-based Bayesian network for multi-state systems. *Reliability Engineering & System Safety*, 211, 107468.
- Byun, J. E. & Song, J. (2021b). A general framework of Bayesian network for system reliability analysis using junction tree. *Reliability Engineering & System Safety*, 216, 107952.
- Koller, D., & Friedman, N. (2009). *Probabilistic graphical models: principles and techniques*. MIT press.
- Porta, S., Crucitti, P., & Latora, V. (2006). The network analysis of urban streets: A dual approach. *Physica A: Statistical Mechanics and its Applications*, 369(2), 853-866.
- Song, J. & Kang, W. H. (2009). System reliability and sensitivity under statistical dependence by matrix-based system reliability method. *Structural Safety*, 31(2), 148-156.
- Valiant, L. G. (1979). The complexity of enumeration and reliability problems. *SIAM Journal on Computing*, 8(3), 410-421.
- Zhang, J., Pourazarm, S., Cassandras, C. G., & Paschalidis, I. C. (2018). The price of anarchy in transportation networks: Data-driven evaluation and reduction strategies. *Proceedings of the IEEE*, 106(4), 538-553.