Reliability Engineering and System Safety 189 (2019) 21–30

Contents lists available at ScienceDirect

Reliability Engineering and System Safety

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

Analytical probability propagation method for reliability analysis of general complex networks

Yanjie Tong[a,*], Iris Tien[b]

[a] Ph.D. Student, School of Civil and Environmental Engineering, Georgia Institute of Technology, Atlanta, GA 30332-0355, USA
[b] Assistant Professor, School of Civil and Environmental Engineering, Georgia Institute of Technology, Atlanta, GA 30332-0355, USA

ARTICLE INFO

Keywords:
Reliability analysis
Probability propagation
System reliability
Approximate analysis
Network graphs

ABSTRACT

Reliability analysis of complex networks is often limited by large and exponentially increasing computational requirements with system size. In this paper, a new approximated analytical method the authors call the probability propagation method (PrPm) is proposed to calculate the reliability of general complex networks. The proposed method originates from the idea of belief propagation for inference in network graphs to pass a joint probability distribution between nodes in the network. At each step, the distribution is updated and passed as the message to its direct neighbors. After the message passes to the terminal node, an approximation of the network reliability is found. In this paper, the derived updating rules for message passing are provided, as well as a precise formulation of the error compared to the exact solution. The method is applied to three test applications: an example from a previous study on network reliability, a power distribution network, and a general grid network. Results from these applications show high accuracy for the proposed method compared to exact solutions where possible for comparison. In addition, the authors show orders of magnitude increases in computational efficiency of PrPm compared to existing approaches. This includes reducing the computational cost for analyses from an exponential increase in computation time with the size of the system to a quartic increase. The proposed PrPm enables accurate and computationally tractable reliability assessments of larger, complex networks.

1. Introduction

The reliability analysis of systems is important to assess and predict the performance of general complex networks. Furthermore, inference over the network enables identification of critical components in the system to support decision makers in setting inspection, maintenance, or replacement policies. Many approaches exist to assess the reliability of systems. These can generally be categorized as analytical or simulation-based. Analytical approaches often require computationally intensive total enumeration, either of the states of the components of a system or of its link or cut sets. These processes result in exact assessments of system reliability; however, they are typically characterized by exponential increases in computational cost with system size. As an alternative, simulation-based methods can be used. These result in approximations of the reliability with increasing the number of sample points generally yielding more accurate approximations of the exact solution. However, for large complex networks, generating a sample point and determining its outcome is time consuming. Several methods to increase efficiency in sampling as well as to generate unbiased sample points have been developed. In this paper, we propose an alternative analytical method, called the probability propagation method (PrPm), to achieve accurate and computationally tractable reliability assessments of complex networks. The systems of interest consist of connected components modeled as networks of links and nodes. PrPm originates from the idea of belief propagation to pass messages from node to node. The passing of an approximated joint probability distribution results in an analytical solution for the system reliability. The accuracy of the resulting approximated solution is influenced by the assumptions made in message propagation. However, the computation time is reduced significantly from an exponential to a quartic increase with system size. In applying the proposed method to three test examples, the performance of PrPm is investigated compared to existing methods.

2. Background and related work

A brief description of existing methods for system reliability analysis is now provided, as well as the background for the proposed method.

* Corresponding author.
E-mail addresses: yjtong@gatech.edu (Y. Tong), itien@ce.gatech.edu (I. Tien).

https://doi.org/10.1016/j.ress.2019.04.013
Received 3 August 2017; Received in revised form 4 February 2019; Accepted 3 April 2019
Available online 09 April 2019

This is intended to provide an overview of methods for network reliability assessment rather than to serve as a comprehensive list. The reader is referred to texts such as Birolini [2] for more details on reliability engineering.

At a fundamental level, systems can be assessed as a combination of the two basic network configurations: parallel and series. These configurations can be used to model redundancy and linear connections between components, respectively. Reliability analysis for simple networks is easily determined by the characteristics of parallel and series systems. However, most realistic systems are in complex configurations, e.g. critical infrastructure systems such as power distribution networks with multiple sources and system interconnects, or overlapping pipeline designs for water and gas networks, which cannot be reduced to simple series and parallel configurations.

One method to analytically assess the reliability of general complex networks is through total enumeration, which lists all possible combinations of components and their corresponding outcomes in the system. Criticism for total enumeration comes from its exponential increase in computational cost as the number of components in the system increases.

An alternative analytical approach is based on the recursive decomposition algorithm (RDA) as presented in Dotson and Gobien [9] and described in Lim and Song [12]. Selective RDA is proposed to improve the efficiency of the original RDA by identifying the most reliable paths. While the number of disjoint sets and computational cost is reduced heavily in the test network, a rigorous proof of faster convergence compared to using the shortest path is not provided. In cases where the most reliable paths are not significantly more dominant than others, the computational cost may still follow an exponential increase. In Kim and Kang [11], the authors extend the application of RDA from one initial and one terminal node to general multi-initial and multi-terminal node networks. In RDA, certain components in a link set are considered to be failed in a graph. Graphs are then decomposed into sub-graphs recursively by eliminating the failed components in the previous step. Decomposition continues recursively until all disjoint link sets are identified. However, the number of sub-graphs will increase exponentially with the number of nodes in the graph, resulting in an exponential increase in computational cost in some cases.

Another method for analyzing the reliability of systems in complex configurations is through the use of Bayesian networks (BNs). One input required for the analyses is the set of minimum link sets (MLSs) or minimum cut sets (MCSs) of the system. Several efficient methods, e.g. EG-CUT algorithm for undirected graphs proposed by Shin and Koh [16], have been developed to enumerate all MCSs or MLSs, which is an NP-hard problem [17]. By using a blocking mechanism repeatedly, MCSs can be generated at $O(en)$ per minimal cut set, where e is the number of edges and n the number of nodes in the graph. In Tien and Der Kiureghian [19], BNs are used to probabilistically model system performance. Exact solutions for system reliability are achieved by performing inference calculations on the values in the conditional probability distributions defining the performance of each node in the BN. Computational limits in generating the BN for a general complex network, however, still exist [20], particularly for nodes in the BN with many parent nodes on which they depend.

MLSs and MCSs on their own can provide crude lower and upper bounds of system reliability. In Ebeling [7], the reliability bounds of the system are determined by considering all MLSs to be in parallel (survival of any link set yields survival of the system) and all MCSs to be in series (failure of any cut set yields failure of the system). However, this method still relies on the generation of all MLSs and MCSs, which is NP-hard. In addition, the bounds provided by this method can be wide as it assumes that all MLSs and MCSs are independent of each other.

As an alternative to analytical solutions, simulation-based methods are widely used to assess the reliability of networks. Several sampling methods have been proposed to achieve improved efficiency in estimating low system failure probabilities, e.g., the random walk on graphs [4] and the refined stratified sampling strategy [15]. Bulteau and El Khadiri [3] combine importance sampling and stratified Monte Carlo principles to generate nodal states. However, after a sample point is generated, it still needs to be tested against the MLSs or MCSs to determine the network outcome. Rejection sampling is used by Cheng et al. [5] in parametric sensitivity analysis and approximation of probability of failure. Compared with direct Monte Carlo simulation and extended Monte Carlo simulation analysis, rejection sampling improves both accuracy and efficiency. However, this method requires finding a distribution from which to sample.

For flow network reliability measures, subset simulation also improves on accuracy and efficiency compared to basic Monte Carlo, e.g., the subset simulation-based network reliability analysis in Zuev et al. [23]. One of the main challenges, however, is calculating the indicator function, which defines the system state given states of the links. Although sampling size for subset simulation is small compared with traditional Monte Carlo, it is still expensive to evaluate the indicator function for each sample point. One of the advantages of our proposed method is the absence of an indicator function as the distribution of nodal states is determined by propagation across the network as described in the following section.

We propose a new analytical method, called PrPm, to obtain accurate and computationally tractable reliability assessments of general networks. The proposed method originates from the idea of belief propagation to perform inference in network graphs. Belief propagation is a message-passing algorithm that provides an exact solution for acyclic graphs. The reader is referred to Coughlan [6] and Barber [1] for more details on the method. In general, a message is calculated and passed to other nodes in the graph, where it is updated before continuing propagation. The message, which is a partial sum reusable for the marginalization, is obtained by calculating the marginal distribution of each unobserved node conditioned on any observed nodes. The message carried by a node is updated according to the message received from its direct neighbors. Based on the Hammersley-Clifford theorem, for nodes in the graph X, the joint distribution $p(X) = \frac{1}{Z}\prod_{c\in\xi}\Psi_{x_c}$, where Z is the normalization constant, ξ is the set of maximal cliques of the graph, and Ψ are the potential functions. The number of terms in the joint distribution $p(X)$ grows exponentially as the number of nodes in the network increases. The advantage of belief propagation is that marginal probabilities can be computed in a time that grows only linearly with the number of nodes in the system [22]. However, for cases where the joint distribution $p(X)$ cannot be expressed explicitly, as for a general network, belief propagation loses its advantage.

In this paper, we propose the new PrPm based on the idea of belief propagation to analyze the reliability of general complex networks. As an overview of the method, we first begin at the source node. We calculate the two-node joint probability distribution, which is the message we pass from the source node to its direct neighbor(s). The two-node joint distribution is used as an efficient approximation of the full joint probability distribution over all nodes. We continue passing the message to direct neighbors according to a propagation sequence, which we determine based on the network configuration. Because of the two-node distribution message, we expand the network as needed so that every node in the network receives a message from at most two direct neighbors. This nodal expansion step facilitates the calculation while retaining the topological characteristics of the network. Finally, the message that is passed to the terminal node provides the approximation of the reliability of the network, where reliability is defined as the probability of reaching the terminal node from the source node.

The rest of the paper is organized as follows. In the following section, the proposed PrPm is described in more detail, including determination of the propagation sequence, providing the updating rules for the message passing, describing the nodal expansion process, and providing a precise derivation of the error compared to the exact solution. We then apply the method to three test applications: a simple example from a previous study on network reliability for validation against the exact solution, a more complex system of a real power distribution network, and a general grid network for comparison with existing approaches and analysis of systems of increasing size. The performance of the method for the three examples is shown in terms of accuracy and computational cost. In the grid example, further discussion on the two-node versus full joint distribution is provided. The paper concludes with a comparison between the proposed method and existing methods in terms of computation time and accuracy for network reliability analysis.

3. Proposed method: PrPm

3.1. Probability propagation sequence

The objective of the proposed PrPm is to propagate the message throughout the entire network starting from the source node to the terminal node. To do this, the sequence of probability propagation must be determined. The following terminology is used: if a node does not carry any message, it is labeled as a non-propagated node. Once a node receives a message from its neighbors, it is recognized as a propagated node. In each step of probability propagation, the message passes from the propagated node to its non-propagated direct neighbors. The sequence of nodes in receiving and passing the message is determined based on the three rules listed below:

1 Newly defined propagated nodes must be the direct neighbors of propagated nodes.
2 Newly defined propagated nodes should not separate any two non-propagated nodes, which guarantees that every node in the network is considered.
3 Newly defined propagated nodes should not connect with each other, which guarantees that every link in the network is considered.

An example of the propagation sequence determination is shown in Fig. 1. The source and terminal nodes are marked as *S* and *T*, respectively. Note that *S* and *T* can occur anywhere in the network. The other nodes are numbered for clarity in the illustration. The method is applicable for both directed and undirected graphs as the derivation of the updating rules in the following section does not depend on the directivity of the links. In the case of directed links, such as one described by two unidirectional links where the reliability from a node *i* to node *j* differs from that from *j* to *i*, one would need only to specify two sets of link reliabilities for the two directions for R_1 and R_2 in the updating rules described later in Tables 1 and 2. The method is also applicable for cyclic networks, an example of which is illustrated in Fig. 1.

In Fig. 1 and in the rest of the paper, we use three symbols to denote the different node types. The empty circle represents a non-propagated node that has not yet received any message. The solid circle represents a propagated node that will not be involved in any future massage passing. We name these as non-boundary nodes. The solid diamond represents a propagated node that will be involved in future propagation steps. We name these as boundary nodes.

The top left graph in Fig. 1 shows the initial state. In it, the source node is the only node that carries a message and is labeled as a solid diamond. The remaining nodes are labeled as empty circles because they have not yet received any message. Following rule #1, *S* is ready to propagate its message to its direct neighbors, nodes 2, 8, 12, and 6. The next step of the propagation is shown in the second graph from top left. Note that if node *S* propagates to nodes 2, 8, 12, and 6 at the same time, node 1 will be separated from the terminal node, which violates rule #2. Therefore, the next nodes propagated are 2, 8, and 12. The next step of the propagation is shown in the third graph from top left. From nodes 2, *S*, and 12, if we pass the message to nodes 1, 6, and 11 in the

Table 1
Updating rules when receiving message from one direct neighbor.

A	C	Pr		A	C	N	Updates
0	0	P_1	⇒	0	0	0	$P_1(1\text{-}R_1R)$
				0	0	1	P_1R_1R
0	1	P_2		0	1	0	$P_2(1\text{-}R_1R)$
				0	1	1	P_2R_1R
1	0	P_3		1	0	0	$P_3(1\text{-}R_1R)$
				1	0	1	P_3R_1R
1	1	P_4		1	1	0	$P_4(1\text{-}R_1R)$
				1	1	1	P_4R_1R

Table 2
Updating rules when receiving message from two direct neighbors.

A	B	C	Pr		A	B	C	N	Updates
0	0	0	P_1	⇒	0	0	0	0	P_1
					0	0	0	1	0
0	0	1	P_2		0	0	1	0	P_2
					0	0	1	1	0
0	1	0	P_3		0	1	0	0	$P_3(1\text{-}R_2R)$
					0	1	0	1	$P_3(1\text{-}R_1)R_2R$
0	1	1	P_4		0	1	1	0	$P_4(1\text{-}R_2R)$
					0	1	1	1	$P_4(1\text{-}R_1)R_2R$
1	0	0	P_5		1	0	0	0	$P_5(1\text{-}R_1R)$
					1	0	0	1	$P_5RR_1(1\text{-}R_2)$
1	0	1	P_6		1	0	1	0	$P_6(1\text{-}R_1R)$
					1	0	1	1	$P_6(1\text{-}R_2)R_1R$
1	1	0	P_7		1	1	0	0	$P_7\{1\text{-}[1\text{-}(1\text{-}R_1)(1\text{-}R_2)]R\}$
					1	1	0	1	$P_3R_1R_2R+P_5RR_1R_2+P_7[1\text{-}(1\text{-}R_1)(1\text{-}R_2)]R$
1	1	1	P_8		1	1	1	0	$P_8\{1\text{-}[1\text{-}(1\text{-}R_1)(1\text{-}R_2)]R\}$
					1	1	1	1	$P_4R_1R_2R+P_6RR_1R_2+P_8[1\text{-}(1\text{-}R_1)(1\text{-}R_2)]R$

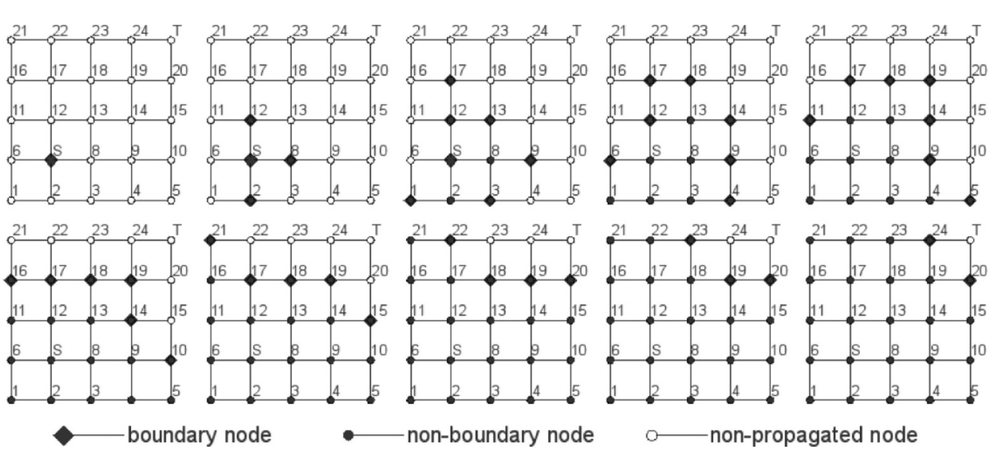

Fig. 1. Propagation sequence illustration from source node *S* to terminal node *T*.

same step, links 6 − 1 and 6 − 11 are excluded from the network, which violates rule #3. Therefore, the next nodes propagated are 1, 3, 9, 13, and 17. The propagation continues until reaching the final step of the propagation as shown in the bottom right graph. By receiving the message from boundary nodes 24 and 20, the reliability at the terminal node T is determined. All steps are shown in Fig. 1. In some cases, including in the example network shown in Fig. 1, there are multiple propagation sequences satisfying the three rules above. For example, for the top right graph in Fig. 1, you may choose 11 → 19 → 5 or 5 → 11 → 19, etc. As long as the sequence satisfies the propagation rules, it is acceptable. A rule of thumb is to prioritize propagating to nodes with one direct neighbor as this yields no approximation in the calculation.

3.2. Message passing and updating rules

We now discuss the message that is passed from node to node and how it is updated during propagation. We assume that each node receives messages from at most two direct neighbors. The situation where a node has more than two direct neighbors is addressed through a nodal expansion procedure presented in the following subsection. We also assume a binary network, i.e., one where nodes can be in one of two states such as 0 or 1 indicating failure or survival, respectively. For multi-state networks, the proposed PrPm is still workable if the multi-state network is converted into a binary state network. For example, we can classify a system of multiple states as achieving or not achieving a certain level of service, or define survival as link flow capacity over a certain threshold and failure otherwise. Here, for message passing, two cases are considered: when a node receives a message from one direct neighbor as shown in Fig. 2, or from two direct neighbors as shown in Fig. 3. In these figures, we denote the node that receives a message as N, the direct neighbors that pass the message as A and B, and a general boundary node that is not a direct neighbor to N as C.

For the first case (Fig. 2), node N receives a message from one direct neighbor A. The message is the joint distribution of the two nodes A and C from the previous propagation step. If it is the initial step, the message is the prior distribution of the source node. Reliability R_i denoted with a subscript indicates reliability of a link. The survival of node N is dependent on the survival of node A and the reliability of the link $A - N$ denoted R_1. The new discrete three-node joint distribution $p(A, C, N)$ is then derived using the updating rules shown in Table 1, where the distribution indicates the probability that each node is in one of two states, failure indicated 0 or success indicated 1. It is noted that no approximations are made in this calculation. Table 1 provides the updated probabilities that nodes A, C, and N are in each of the possible combinations of states 0 or 1. R denotes the reliability of node N, i.e., $R = P(N = 1)$, which is previously defined. Once the three-node joint distribution is obtained, we can easily define the new two-node joint distributions $p(A, N)$ and $p(C, N)$.

For the second case (Fig. 3), node N receives a message from two direct neighbors A and B. The message we need for the calculation is the joint distribution of the two nodes A and B and the marginal

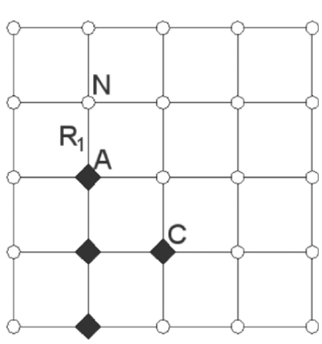

Fig. 2. Message passing illustration from one direct neighbor.

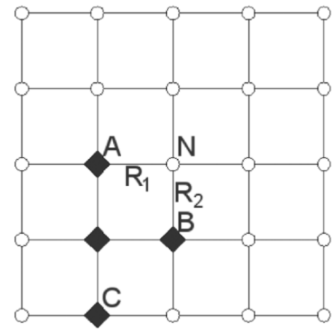

Fig. 3. Message passing illustration from two direct neighbors.

distribution of node C, which can be inferred from the two-node joint distribution including node C. In updating the message, we assume that node C is separated from nodes A and B, which indicates that links $A - N$ and $B - N$ have no influence on node C. This underestimates reliability of node C. A detailed analysis of the error introduced by this assumption is provided later in this section.

Table 2 shows the updating rules to build the four-node joint distribution $p(A, B, C, N)$ from the three-node joint distribution $p(A, B, C)$, where R, R_1, and R_2 indicate the reliabilities of node N, link $A - N$, and link $B - N$, respectively. The new joint distributions $p(A, N)$, $p(B, N)$, and $p(C, N)$ for future propagation steps can be defined accordingly. Based on the four-node joint distribution $p(A, B, C, N)$, $p(A, B)$, $p(A, C)$, and $p(B, C)$ are updated as well.

One important result from the updating rules given in Tables 1 and 2 is that we need the joint distributions of only two nodes rather than all nodes during the message-passing process. While this yields an approximated solution, PrPm reduces the computational cost from an exponential increase with the number of nodes in the network $O(2^n)$ to a quartic increase $O(n^4)$. A detailed analysis of the computational complexity of the method is provided later in this section.

3.3. Nodal expansion

The updating rules in Tables 1 and 2 are based on the assumption that every node receives a message from no more than two nodes. In a general network, however, it is possible that a node receives a message from a greater number of nodes. For example, as shown in Fig. 4(a), a node i can have four or five direct neighbors. The updating in Tables 1 and 2 will not work for these configurations. However, we can expand the node i as shown in Fig. 4(b). It is easy to prove that the configurations in Fig. 4(a) are equivalent to the configurations in Fig. 4(b), for which the previously derived updating rules are applicable. For example, for the four-neighbor case, instead of updating node i directly, we update the node sequentially $i_1 \rightarrow i_2 \rightarrow i_3 \rightarrow i_4$ as an alternative. Similarly, for the five-neighbor case, the updating rules are applicable if we update the node $i_1 \rightarrow i_2 \rightarrow i_3 \rightarrow i_4 \rightarrow i_5$ as shown on the right. In the proposed method, nodal expansion is performed before beginning the message passing. Without affecting the connectivity of original network, the additional links created by nodal expansion are set to be 100% reliable.

3.4. Overall method

The full flowchart of the proposed PrPm is shown in Fig. 5. First, we determine the propagation sequence based on the network configuration and propagation rules. This provides the sequence of steps in the probability propagation process for when and how each node receives the message from the other nodes. Then, we expand the nodes in the network as necessary to ensure that every node receives a message from at most two direct neighbors. Next is the message passing between nodes, where we define and update the message based on link and node

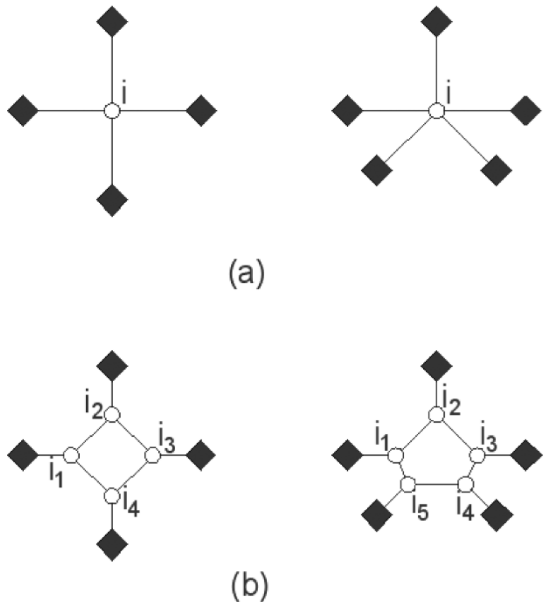

Fig. 4. Nodal expansion illustration from multiple direct neighbors (a) to two direct neighbors (b).

reliabilities and the derived updating rules. After the message propagates to the terminal node, the approximated analytical solution of the network reliability is obtained.

3.5. *Target networks*

The proposed PrPm is applicable to directed, undirected, cyclic, and acyclic networks. In the case of directed networks, the method is able to analyze networks with bidirectional or unidirectional links as long as the individual link reliabilities are specified. PrPm works for both single-source-single-sink networks as shown in the first and third test applications, and multiple-sources-single-sink networks as shown in the second test application. For applications on networks with multiple sinks, for example as in Liu and Li [13], evaluations can be done on each terminal node separately. Applicable networks of the proposed method should have independent or conditionally independent links or nodes as we do not consider the link conditional probabilities in calculating nodal joint distributions. However, the method can be applied to achieve significant computational savings for systems with dependent components by conditioning on common parents of the links or nodes.

3.6. *Computational complexity analysis*

The computational complexity of the method derives from the nodal expansion and updating rules. As we need to expand the node to ensure that every node receives information from at most two direct neighbors, for a network with n nodes, the newly defined propagated nodes connect to $O(n)$ neighbors with the maximum number being n for a fully connected network. Thus, there will be $O(n^2)$ nodes in total. According to the updating rules, for each newly defined propagated node, the computational cost for that node is $O(n^2)$, as the number of C nodes is $O(n^2)$ with the maximum number being $n^2 - 2$, excluding node A and node N as shown in Fig. 2. Therefore, the total computational cost is the combined individual computational costs, $O(n^2)O(n^2) = O(n^4)$.

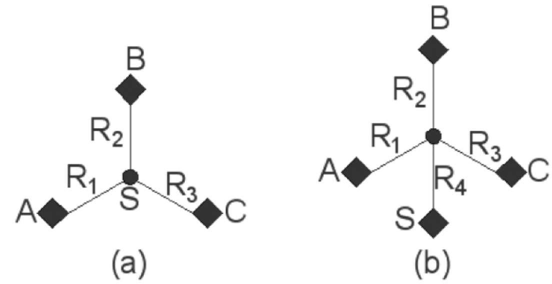

Fig. 6. Illustration of the exact (a) and extreme (b) cases for error analysis.

3.7. *Error analysis*

We now discuss the approximation error of the proposed method compared to the exact solution. The error in message passing arises from building the three-node joint distribution $p(A, B, C)$ from $p(A, B)$ and $p(C)$ as shown in Table 2 and the assumption that node C is separated from nodes A and B. The exact case and the extreme case for assessing the error are shown in Fig. 6(a) and (b), respectively. For the purposes of the illustration, the source node S is taken as the previously propagated node. In Fig. 6(a), nodes A, B, and C are connected to S by three independent links with reliabilities R_1, R_2, and R_3. Based on the assumptions made in Table 2 that node C is independent of nodes A and B, PrPm will give us the exact joint probability in this scenario. However, Fig. 6(b) shows the extreme case, i.e., the worst case in terms of error, where a common link with reliability R_4 is shared by paths from all nodes A, B, and C to S. The assumption in generating the joint distribution will not hold in this case because the reliability of node C is influenced by the states of nodes A and B. For this case, the comparison between the exact distribution and the approximated distribution for PrPm is shown in Table 3.

In Table 3, the rightmost column gives the difference between the exact and PrPm values for each element of the joint distribution $p(A, B, C)$, denoted Δ_i. Analyzing the expressions for Δ_i enables us to quantify and analyze the error in the approximation. Specifically, we see that $\Delta_1, \Delta_4, \Delta_6, \Delta_8$ are greater than 0, which means that we underestimate their probability shares in the proposed method; $\Delta_2, \Delta_3, \Delta_5, \Delta_7$ are less than 0, which means that we overestimate their probability shares. In addition, $\Delta_7 + \Delta_8 = 0$, indicating that the underestimation of the probability for the more likely-to-be-survived state ($A = 1, B = 1, C = 1$) is equal to the overestimation of the probability for the less likely-to-be-survived state ($A = 1, B = 1, C = 0$). Likewise, $\Delta_5 + \Delta_6 = 0$ and $\Delta_3 + \Delta_4 = 0$, indicating that the differences in their probability shares are reallocated from the more likely-to-be-survived

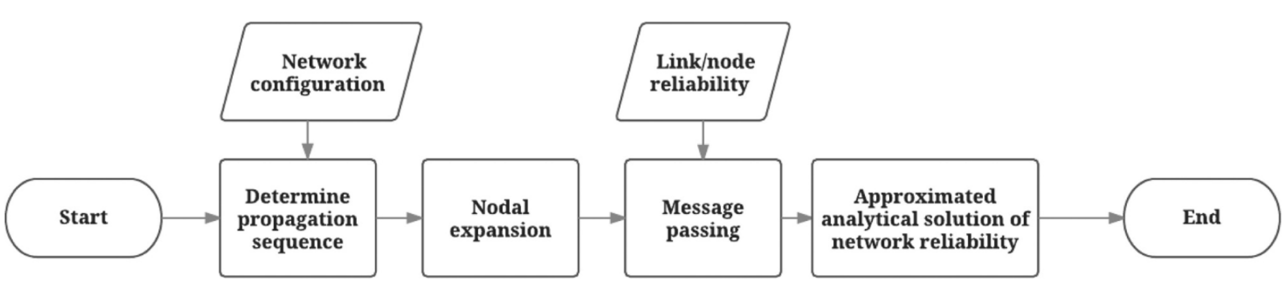

Fig. 5. Flowchart of the proposed PrPm.

Table 3
Comparison between exact solution and distribution obtained from PrPm.

A	B	C	Exact	PrPm	Difference	
0	0	0	$1 + R_4(R_1R_2+R_1R_3+R_2R_3-R_1-R_2-R_3-R_1R_2R_3)$	$[1 + R_4(R_1R_2-R_1-R_2)](1-R_3R_4)$	Δ_1	$(R_1+R_2-R_1R_2)R_3R_4(1-R_4)$
0	0	1	$(1-R_1)(1-R_2)R_3R_4$	$[1 + R_4(R_1R_2-R_1-R_2)]R_3R_4$	Δ_2	$-(R_1+R_2-R_1R_2)R_3R_4(1-R_4)$
0	1	0	$(1-R_1)R_2(1-R_3)R_4$	$(1-R_1)R_2(1-R_3R_4)R_4$	Δ_3	$-(1-R_1)R_2R_3R_4(1-R_4)$
0	1	1	$(1-R_1)R_2R_3R_4$	$(1-R_1)R_2R_3R_4^2$	Δ_4	$(1-R_1)R_2R_3R_4(1-R_4)$
1	0	0	$R_1(1-R_2)(1-R_3)R_4$	$R_1(1-R_2)(1-R_3R_4)R_4$	Δ_5	$- R_1(1-R_2)R_3R_4(1-R_4)$
1	0	1	$R_1(1-R_2)R_3R_4$	$R_1(1-R_2)R_3R_4^2$	Δ_6	$R_1(1-R_2)R_3R_4(1-R_4)$
1	1	0	$R_1R_2(1-R_3)R_4$	$R_1R_2(1-R_3R_4)R_4$	Δ_7	$- R_1R_2R_3R_4(1-R_4)$
1	1	1	$R_1R_2R_3R_4$	$R_1R_2R_3R_4^2$	Δ_8	$R_1R_2R_3R_4(1-R_4)$

states to the less-likely-to-be-survived states as well. This underestimates the reliability. The only contrary case is for Δ_1 and Δ_2, where $\Delta_1 + \Delta_2 = 0$. However, $\Delta_1 = \Delta_4 + \Delta_6 + \Delta_8$; therefore, the magnitude of the overestimation error equals the sum of the underestimation errors.

In addition, $P(R|A = 0, B = 0, C = 1) - P(R|A = 0, B = 0, C = 0) > P(R|A = 1, B = 1, C = 1) - P(R|A = 1, B = 1, C = 0)$, where $P(R|A, B, C)$ denotes the reliability of the network given the states of nodes *A*, *B*, and *C*. On the left-hand side of the inequality, terminal node *T* cannot be reached from nodes *A* and *B*; while, on the right-hand side of the inequality, terminal node *T* can be reached from nodes *A* and *B*. The inequality holds because paths from node *C* to terminal node *T* may share common links with paths from node *A* or *B* to *T*. For the same reason, $P(R|A = 0, B = 0, C = 1) - P(R|A = 0, B = 0, C = 0) > P(R|A = 0, B = 1, C = 1) - P(R|A = 0, B = 1, C = 0)$ and $P(R|A = 0, B = 0, C = 1) - P(R|A = 0, B = 0, C = 0) > P(R|A = 1, B = 0, C = 1) - P(R|A = 1, B = 0, C = 0)$. Since $\Delta_1 = \Delta_4 + \Delta_6 + \Delta_8$, it yields that $[P(R|A = 0, B = 0, C = 1) - P(R|A = 0, B = 0, C = 0)]\Delta_1 > [P(R|A =1, B = 1, C = 1) - P(R|A = 1, B = 1, C = 0)]\Delta_8 + [P(R|A = 1, B = 0, C = 1) - P(R|A = 1, B = 0, C = 0)]\Delta_6 + [P(R|A = 0, B = 1, C = 1) - P(R|A = 0, B = 1, C = 0)]\Delta_4$. This indicates the reallocation of the probability shares created by the proposed method overestimates the reliability of the network, resulting in the upper bound.

However, we also assume there is no connection between nodes *A* and *B* and node *C*. Under this assumption, *C* cannot be reached from *A* and *B* when links $A - N$ and $B - N$ are added as in Fig. 3. This underestimates the connectivity of the network and tends the reliability toward the lower bound. These two effects, overestimation of the joint distribution *P*(*A, B, C*) and underestimation of the reliability of node *C*, combine and cancel out the error to some extent. In practice, the actual error will fall between the errors given by the two extreme cases. Thus, the result obtained by the proposed method becomes a relatively accurate approximation to the exact solution as shown in the test applications.

A close look at the difference terms in Table 3 also reveals the performance of the method under high system reliability and low system reliability scenarios. All difference terms, Δ_1 to Δ_8, share a common factor $R_4(1 - R_4)$. For both a highly reliable system and in a low reliability setting such as under a hazard, the term $R_4(1 - R_4)$ tends to 0, reducing the error in these cases.

In addition, due to the source of the approximation error, the accuracy of the proposed PrPm increases as system failure probability decreases. This is in contrast to most sampling-based approaches. The sources of the error in PrPm compared to the exact solution are 1) overestimation by the three-node joint distribution and 2) underestimation by assuming that *C* cannot be reached from *A* and *B* when links $A - N$ and $B - N$ are added as shown in Fig. 3.

4. Test applications

We now apply the proposed PrPm to three test applications and assess the performance in terms of accuracy and computational cost. All results are based on computations run in MATLAB_R2016b on a 16 GB RAM computer. All examples have exact solutions and computational costs for comparison with the proposed method. We demonstrate the procedure of calculation in detail in the first example, which is a simple single-source-single-sink network. A more complex, real-world, multiple-source-single-sink network is analyzed in the second example, including assessments of the system with increasing link reliabilities. A highly connected grid network is tested in the third example to assess the performance of the proposed method in terms of both accuracy and efficiency for systems of increasing size.

4.1. Seven-component network

First, we apply PrPm to an example from a previous study on network reliability [21], which is shown in Fig. 7. This network is chosen as it is irreducible to series and parallel components. It facilitates simple illustration of the method, and the exact solution can be obtained to compare accuracy with the result achieved by PrPm. For this example, the reliability of each link is assumed to be 0.9. It is noted that as PrPm calculates the network failure probability analytically, it is equally computationally efficient for varying link failure probabilities of any value across the network, including for highly reliable networks with low probabilities of failure. For this example, nodes are considered to be perfectly reliable. In the figure, *S* and *T* represent the source and terminal nodes, respectively.

Following the overall process of the proposed PrPm shown in Fig. 5, we first determine the propagation sequence, i.e., the order of nodes to receive messages by the propagation and updating rules. In this case, we pass messages following two possible sequences: $S \to 1 \to 2 \to 3 \to 4 \to T$ or $S \to 1 \to 3 \to 2 \to 4 \to T$. In either case, the sequence ensures every node is propagated before reaching the terminal node with both sequences working equally. For this network, no node requires nodal expansion.

We then pass the message through the network according to the probability distribution updating rules in Tables 1 and 2. For illustration, we choose the first sequence above. We begin with node *S*. Initially, we have $P(S = 1) = 1$ and $P(S = 0) = 0$. The message is then updated to node 1, the direct neighbor of *S*. Let nodes 1, ...,4 be denoted N_1, ...,N_4. In this step, $P(N_1 = 1) = 0.9$ and $P(N_1 = 0) = 0.1$. Next, the message is updated to node 2 with $P(N_1 = 1, N_2 = 1) = 0.81$, $P(N_1 = 1, N_2 = 0) = 0.09$, $P(N_1 = 0, N_2 = 1) = 0$, and $P(N_1 = 0, N_2 = 0) = 0.1$. Node 3 is then updated with message

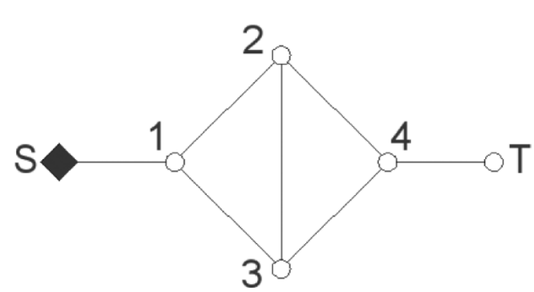

Fig. 7. Example irreducible seven-component network.

Table 4
Performance comparison for seven-component network between exact and PrPm solutions.

	Reliability	Computation time (sec)
Exact solution (BN)	0.7926	35.00
PrPm solution	0.7926	0.06

$P(N_2 = 1, N_3 = 1) = 0.8748$, $P(N_2 = 1, N_3 = 0) = 0.0081$, $P(N_2 = 0, N_3 = 1) = 0.0081$, and $P(N_2 = 0, N_3 = 0) = 0.109$. Then, message is updated to node 4 with $P(N_4 = 1) = 0.8806$ and $P(N_4 = 0) = 0.1194$. Finally, we reach the terminal node with a message $P(T = 1) = 0.7926$ and $P(T = 0) = 0.2074$ to complete the reliability calculation.

In this case, as there is no C node during the propagation as shown in Fig. 3, the result obtained by PrPm is the exact solution with no errors. As a comparison, we cite the results from Tong and Tien [21], which provide an exact solution for the reliability of the network by a Bayesian network (BN) formulation. The comparison is given in Table 4. In this case, the reliability value computed using PrPm is exact. In terms of the computational cost, we see that computation time is reduced by more than two orders of magnitude or 500 times to arrive at the exact answer.

4.2. Power distribution network

Next, we apply PrPm to a more complex system, which is the example four-substation power distribution network from Pacific Gas and Electric [14] shown in Fig. 8, also investigated in Der Kiureghian and Song [8] and Tien [18]. The original system consists of 59 components, including circuit breakers, switches, and transformers. Each triplet configuration in the system of switch-breaker-switch can be easily represented as a single component. Therefore, 22 components are shown in Fig. 8. For this example, all components are assumed to be independent and no nodal failure is considered. Previous studies assess network reliability based on varying component failure probabilities. Here, we convert to link failure probabilities. Compared with the original network, links $1-2$, $3-10$, $5-13$, $7-8$, $11-19$, $14-21$ and $16-18$ are assumed to be perfectly reliable as there are no additional elements on these links. All other links, which have circuit breakers, switches, and transformers located on them, have a probability of failure p_f. The network has multiple sources: nodes 1, 7, and 8; node T is the terminal node. Note that PrPm is able to accommodate the case of multiple source nodes across the network. As a reference, the method of total enumeration is used to obtain the exact solution. Results from Monte Carlo simulation are also provided for comparison. For this network, the existence of the C node as shown in Fig. 3 during the message-passing process introduces errors into the propagation. Therefore, the results obtained by PrPm are an approximation in this case.

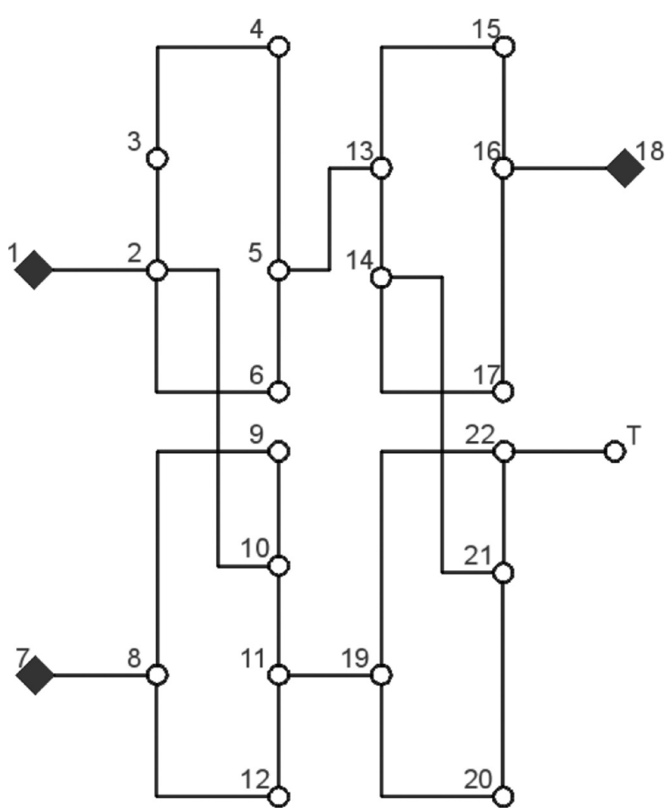

Fig. 8. Power distribution network example.

To investigate the accuracy and computational cost of PrPm over networks of varying reliabilities, we obtain results over a range of link failure probabilities. Table 5 provides the comparison among total enumeration to obtain the exact solution, Monte Carlo simulation, and PrPm. Results are given in terms of system reliability assessment and computation time (in seconds) as p_f increases from 0.01 to 0.2. For Monte Carlo simulation, 10,000 realizations are simulated for each p_f. From Table 5, PrPm outperforms Monte Carlo simulation in both accuracy and computation time. The percentage error relative to reliability for both PrPm and Monte Carlo decrease with smaller probabilities of failure. However, the accuracy relative to system failure probability decreases for Monte Carlo, as expected for simulation-based methods, while PrPm increases in accuracy as failure probabilities decrease as described in the error analysis section.

The network reliabilities obtained by PrPm and the exact solution are plotted in Fig. 9 to show the trend in accuracy across link and system reliabilities. Over the investigated range of link failure probabilities, the maximum percentage error is 0.6357%, with decreasing error for systems of increasing reliability. As discussed in the error analysis section, errors should decrease for cases with low link reliabilities such as under hazard scenarios as well. As an additional comparison, if the link failure probability increases to 0.85, PrPm provides a solution with 0.0801% error, with the exact solution and PrPm indicating system failure probabilities of 0.9982 and 0.9974, respectively. In terms of computation, as PrPm provides an analytical solution, the burden of the method remains constant across system failure probabilities. Therefore, for all cases, PrPm increases the efficiency of obtaining the solution by more than three orders of magnitude and 1800 times, indicated as time ratio in Table 5, in terms of computation time.

4.3. Grid network

Finally, we apply PrPm to analyze the reliability of a general grid network. First, we take a 5 × 5 grid as an example, with the source and terminal nodes at the corners of the grid as shown in Fig. 10. Link failure probability p_f is assumed to be 0.1 and node reliability 1 across the network. We use the proposed PrPm and updating rules listed in Tables 1 and 2 to obtain the approximated system reliability solution. For comparison, we use results from implementing the recursive decomposition algorithm (RDA). The RDA solution results in an upper bound and lower bound. By performing the decomposition recursively, the gap between the bounds is reduced to obtain a single system reliability value as shown in Table 6. Comparison between the performance of RDA and PrPm is listed in Table 6. With a sacrifice of 1.25% in accuracy, we reduce the computation time by more than four orders of magnitude and 35,000 times for the 5 × 5 grid.

To explain the approximation error, we now provide further discussion on the error in propagating the two-node joint probability distribution compared to the full joint distribution in the context of

Table 5
Performance comparison for power distribution network among exact solution, PrPm, and Monte Carlo simulation varying p_f.

p_f	Exact solution Reliability	Exact solution Time (sec)	PrPm Reliability	PrPm Time (sec)	Percentage error (%) relative to reliability	Percentage error (%) relative to failure probability	Time ratio
0.0100	0.9899	113.63	0.9899	0.0629	0	0	1806.52
0.0500	0.9476		0.9474		0.0211	0.3817	
0.1000	0.8900		0.8888		0.1348	1.0909	
0.1500	0.8264		0.8233		0.3751	1.7857	
0.2000	0.7551		0.7503		0.6357	1.9600	

p_f	Exact solution Reliability	Exact solution Time (sec)	Monte Carlo Reliability	Monte Carlo Time (sec)	Percentage error (%) relative to reliability	Percentage error (%) relative to failure probability	Time ratio
0.0100	0.9899	113.63	0.9912	1.1013	0.1313	12.8713	103.18
0.0500	0.9476		0.9454		0.2322	4.1985	
0.1000	0.8900		0.8936		0.4045	3.2727	
0.1500	0.8264		0.8211		0.6413	3.0530	
0.2000	0.7551		0.7635		1.1124	3.4300	

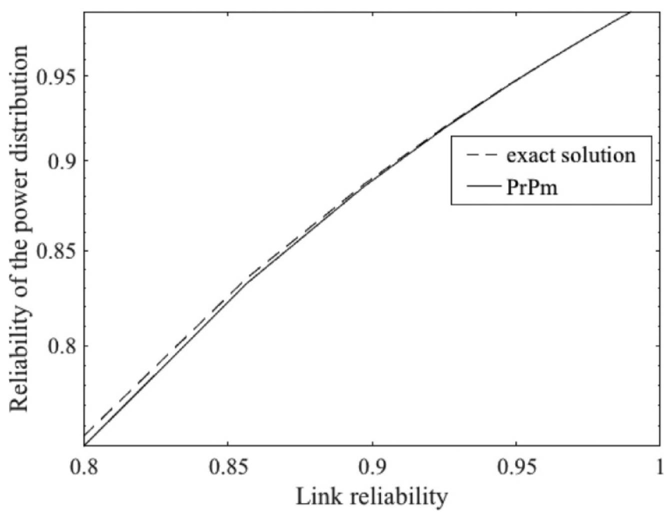

Fig. 9. Exact solution compared to results by PrPm varying p_f.

Fig. 10. A 5 × 5 grid network.

Table 6
Performance comparison for grid network between RDA and PrPm.

RDA Reliability	RDA Time (sec)	PrPm Reliability	PrPm Time (sec)	Percentage error (%)	Time ratio
0.9755	1405.52	0.9877	0.0402	1.2564	35,050.37

analyzing the grid network reliability. As previously discussed, the error in the PrPm approximation comes from the C node in Fig. 3 and using the distributions $p(A, B)$ and $p(C)$ to estimate $p(A, B, C)$ in Table 2. A more accurate result can be obtained by considering the joint distribution of all boundary nodes at each step. By making assumptions on the connectivity within the boundary nodes, i.e., based on whether or not the boundary nodes are connected with each other, we can find the upper bound and lower bound of the system reliability.

For example, suppose we have nodes $\{A,...,I\}$ configured as part of a network as shown in Fig. 11. We use the joint distribution of all boundary nodes, $p(E, F, G, H, I)$, to update the joint distribution of the newly defined propagated nodes, $p(A, B, C, D)$. Let 0 again denote the failure of a node and 1 denote survival. When we update $p(A, B, C, D)$ from $P(E = 1, F = 1, G = 0, H = 1, I = 1)$, the upper bound of $p(A, B, C, D)$ can be found by assuming that nodes E and F, also nodes H and I, are connected; the lower bound of $p(A, B, C, D)$ can be found by assuming that there is no connection between nodes E and F, or between nodes H and I. A similar strategy can be applied to the other node combinations of E, F, G, H, and I. By doing so, we can find the upper bound and lower bound of the system reliability at the terminal node.

To quantify the effect of considering the joint distribution of all boundary nodes compared to the two-node distribution, we assess the accuracy and computation time to obtain the network reliability of grids of increasing size for the two cases. We take the corner-to-corner reliability of the grid network as the example, with link reliability of 0.9 and node reliability 1. Table 7 shows the results of the two-node joint distribution approximation compared to the upper and lower bounds considering the joint distribution of all boundary nodes as the size of the grid increases from 3 × 3 to 100 × 100. The full joint distribution calculation becomes intractable after a grid size of 12 × 12. In Table 7, the obtained bounds from the full joint distribution are guaranteed to include the exact solution. Computation times for both PrPm and the full distribution calculation are provided. Percentage error is calculated for the PrPm approximation result compared to the median of the bounds.

From Table 7, we see that when considering the full joint distribution, there is an exponential increase in computation time as the size of the grid increases. For a propagation step with n_b boundary nodes, calculating the joint distribution requires the storage and updating of 2^{n_b} elements, resulting in an exponentially increasing computational complexity with n at $O(2^{n_b})$. To improve the computational efficiency of reliability assessment of the network, the proposed method only considers the joint distribution of two nodes. With this, the accuracy of the

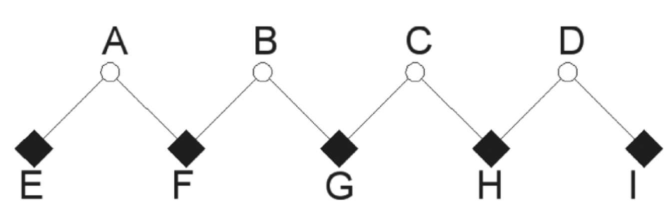

Fig. 11. Illustration for considering the joint distribution of all boundary nodes.

Table 7
Performance comparison for grid network between PrPm and considering the joint distribution of all boundary nodes.

Grid size	PrPm		Full joint distribution			Percentage error (%)	Time ratio
	Reliability	Time (sec)	Reliability bounds		Time (sec)		
			Upper	Lower			
3 × 3	0.9833	0.1003	0.9725	0.9724	0.16	1.1157	1.60
4 × 4	0.9872	0.1092	0.9751	0.9750	0.26	1.2461	2.39
5 × 5	0.9877	0.1109	0.9756	0.9755	0.72	1.2455	6.49
6 × 6	0.9878	0.1162	0.9756	0.9756	2.83	1.2505	24.40
7 × 7	0.9878	0.1162	0.9757	0.9757	10.88	1.2401	93.79
8 × 8	0.9878	0.1162	0.9757	0.9757	42.98	1.2401	370.52
9 × 9	0.9878	0.1162	0.9757	0.9757	174.56	1.2401	1504.83
10 × 10	0.9878	0.1162	0.9757	0.9757	723.55	1.2401	6237.50
11 × 11	0.9878	0.1174	0.9757	0.9757	2986.20	1.2401	25,523.08
12 × 12	0.9878	0.1287	0.9757	0.9757	12,688.82	1.2401	98,362.95
20 × 20	0.9878	0.2188	/	/	/	/	/
30 × 30	0.9878	0.6423	/	/	/	/	/
40 × 40	0.9878	1.7913	/	/	/	/	/
50 × 50	0.9878	4.6558	/	/	/	/	/
75 × 75	0.9878	46.1766	/	/	/	/	/
100 × 100	0.9878	196.0232	/	/	/	/	/

Table 8
Performance comparison for grid network between PrPm and Monte Carlo simulation.

Grid size		PrPm		Monte Carlo simulation	
		Reliability	Percentage error (%)	Reliability	Percentage error (%)
$p_f = 0.1$	3 × 3	0.9833	1.1157	0.9729	0.0411
	4 × 4	0.9872	1.2461	0.9712	0.3897
	5 × 5	0.9877	1.2455	0.9787	0.3178
	10 × 10	0.9878	1.2401	0.9776	0.1947
$p_f = 0.01$	3 × 3	0.9998948	0.0102809	/	/
	4 × 4	0.9998979	0.0102050	/	/
	5 × 5	0.9998980	0.0102031	/	/
	10 × 10	0.9998980	0.0102031	/	/

result is slightly lowered by 1.24%, but the computational cost is reduced by several orders of magnitude, with computational savings increasing as the size of the network increases. With the consideration of the two-node joint distribution, the time complexity of computation for the proposed method is quartic at $O(n^4)$.

To further assess the performance of the method for general grid networks, Table 8 provides a comparison of the accuracy of PrPm compared to results from Monte Carlo simulation. The reader is referred to Dueñas-Osorio [10] for details on the Monte Carlo simulation. For efficiency comparison, as the Monte Carlo simulations are tested on a different computer, they are not included here. For accuracy comparison, results in Table 8 are shown for systems with link failure probabilities of 10% and 1%. Although the average percentage error (0.24%) given by Monte Carlo simulation outperforms the average percentage error of 1.21% by PrPm for the case of link reliabilities of 0.9, Monte Carlo has a major limitation in that in the rare event condition, it can be computationally intractable to generate enough samples to calculate system reliabilities for low failure probability systems, e.g., systems with high link reliabilities. This is shown for networks with link reliabilities of 0.99. For the Monte Carlo simulation in this case, the result fails to converge under 7.8 h of computation for a 3 × 3 grid. In PrPm, computational efficiency is related only to the topology of the network and not influenced by the link reliability.

5. Conclusion

In this paper, a new approximated analytical method, the probability propagation method (PrPm), is proposed to analyze the reliability of general networks. Compared to existing analytical algorithms such as RDA and inference in Bayesian networks, computational complexity with increasing nodes in the network n is reduced from an exponential increase $O(2^n)$ to a quartic increase $O(n^4)$. The method does not require the computationally intensive enumeration of component states, MLSs, or MCSs to determine the system outcome. While the method results in an approximated value for network reliability with a small sacrifice in accuracy, compared with simulation-based methods, the proposed analytical PrPm solution does not require generating sample points or proving convergence. The source of the error in the proposed approximation is analyzed analytically, showing terms that both overestimate and underestimate the system reliability to effectively cancel out to obtain a solution. The performance of PrPm is investigated using three example networks. In the first example, PrPm results in the exact solution as all boundary nodes are direct neighbors at each step. In the second example, PrPm is shown to work for a network with multiple sources. Computational time is reduced by more than 1800 times with a maximum error in the reliability result of 0.64% compared to the exact solution. In the last example, the results show that the computation time does not exponentially increase with system size as with other methods, and the error is stable. Many sampling-based approaches are limited by computational tractability to analyze rare events. For PrPm, as the method calculates the network reliability analytically, it is equally computationally efficient across reliability values. Throughout, the proposed PrPm achieves accurate estimates of network reliability with orders of magnitude savings in computation time. This enables accurate and computationally tractable reliability assessments of larger, complex networks.

Acknowledgments

Support for this work by the National Science Foundation through grant no. CNS-1541074 is acknowledged.

References

[1] Barber D. Bayesian reasoning and machine learning. Cambridge University Press; 2012.
[2] Birolini A. Reliability Engineering: theory and practice. 4th ed. Berlin: Springer; 2004.
[3] Bulteau S, El Khadiri M. A Monte Carlo simulation of the flow network reliability using importance and stratified sampling. PhD diss., INRIA 1998;32(3):271–87.
[4] Cheng W, Cox J, Whitlock P. Random walks on graphs and Monte Carlo methods. Math Comput Simul 2017;135:86–94https://www.sciencedirect.com/science/article/abs/pii/S0378475415002748.
[5] Cheng L, Lu Z, Zhang L. Application of Rejection Sampling based methodology to variance based parametric sensitivity analysis. Reliab Eng Syst Saf 2015;142:9–18.
[6] Coughlan J. A tutorial introduction to belief propagation. The Smith-Kettlewell Eye Research Institute; Aug 2009.
[7] Ebeling CE. An introduction to reliability and maintainability engineering 2nd.

Waveland press, Inc.; 2010. p. 110–2.
[8] Der Kiureghian A, Song J. Multi-scale reliability analysis and updating of complex systems by use of linear programming. Reliab Eng Syst Saf 2008;93:288–97.
[9] Dotson W, Gobien JO. A new analysis technique for probabilistic graphs. IEEE Trans Circu Syst 1979;26(10):855–65.
[10] Dueñas-Osorio, L., "Reliability of Grid Networks and Recursive Decomposition Algorithms," <http://duenas-osorio.rice.edu/Content.aspx?id=2147483674>, accessed April 2017.
[11] Kim Y, Kang W. Network reliability analysis of complex systems using a non-simulation-based method. Reliab Eng Syst Saf 2013;110:80–8.
[12] Lim H-W, Song J. Efficient risk assessment of lifeline networks under spatially correlated ground motions using selective recursive decomposition algorithm. Earthq Eng Struct Dyn 2012;41(13):1861–82.
[13] Liu W, Li J. An improved recursive decomposition algorithm for reliability evaluation of lifeline networks. Earthq Eng Eng Vibra 2009;8(3):409–19. September.
[14] Ostrom, D., "Database of seismic parameters of equipment in substations." <http://peer.berkeley.edu/lifelines/lifelines_pre_2006/final_reports/413-FR.pdf>, 2004, accessed January 10, 2017.
[15] Shields MD, Teferra K, Hapij A, Daddazio RP. Refined stratified sampling for efficient Monte Carlo based uncertainty quantification. Reliab Eng Syst Saf 2015;142:310–25https://www.researchgate.net/publication/276211091_Refined_Stratified_Sampling_for_efficient_Monte_Carlo_based_uncertainty_quantification.
[16] Shin YY, Koh JS. An algorithm for generating minimal cutsets of undirected graphs. Korean J Comput Appl Math 1998;5(3):681–93.
[17] Suh H, Chang CK. Algorithms for the minimal cutsets enumeration of networks by graph search and branch addition. Proceedings of the 25th Annual IEEE Conference on Local Computer Networks, Tampa, FL7. 2000. p. 100–10. November 8–10.
[18] Tien I. Bayesian network methods for modeling and reliability assessment of infrastructure systems. Risk and reliability analysis: theory and applications. Springer International Publishing; 2017. p. 417–52.
[19] Tien I, Der Kiureghian A. Algorithms for Bayesian network modeling and reliability assessment of infrastructure systems. Reliab Eng Syst Saf 2016;156:134–47https://www.sciencedirect.com/science/article/pii/S0951832016302988.
[20] Tien I, Der Kiureghian A. Reliability assessment of critical infrastructure using Bayesian networks. J Infrastruct Syst 2017;23(4).
[21] Tong Y, Tien I. Algorithms for Bayesian network modeling, inference, and reliability assessment for multi-state flow networks. J Comput Civil Eng 2017;31(5).
[22] Yedidia JS, Freeman WT, Weiss Y. Understanding belief propagation and its generalizations. Explor Artif Intell New Millen 2003;8:236–9.
[23] Zuev K, Wu S, Beck J. General network reliability problem and its efficient solution by subset simulation. Probab Eng Mech 2015;40:25–35.