Approximate Reliability of Multi-state Two-Terminal Networks by Stochastic Analysis

Peican Zhu, Yangming Guo, Fabrizio Lombardi, and Jie Han

-

***Abstract*— Reliability is an important feature in the design and maintenance of a large-scale system. Usually, for a two-terminal network reliability is a measure for the connectivity between the *source* and *sink* nodes. Various approaches have been presented to evaluate the system reliability, however they become cumbersome or prohibitive due to the large computational complexity, especially when multiple states are considered for the nodes. In this paper, a stochastic approach is presented for estimating the reliability of a two-terminal multi-state network. Randomly permuted sequences with fixed numbers of multiple values are generalized from non-Bernoulli binary sequences (so with fixed numbers of 0s and 1s) to model the multi-state property. The state probabilities are represented by randomly permuted sequences to improve both computational efficiency and accuracy. Stochastic models are then constructed for arcs and nodes with different capacities. The proposed stochastic analysis is capable of predicting a system's reliability at high accuracy and without a need for constructing the commonly-used but complex multi-state minimal cut vectors (MMCVs). Non-exponential probability distributions and correlated signals are readily handled by the stochastic approach for a generalized multi-state two-terminal network. Results obtained through the stochastic analysis are compared with exact values and those found by Monte Carlo (MC) simulation. The accuracy, efficiency and scalability of the stochastic approach are assessed by evaluating several case studies.**

***Index Terms*—Stochastic computation, non-Bernoulli sequence, randomly permuted sequences, stochastic logic, network reliability evaluation, multi-state two-terminal network.**

ACRONYM

DFT	dynamic fault tree
pdf	probability density function
cdf	cumulative density function
MMCV	multi-state minimal cut vectors
MC	Monte Carlo
IE	Inclusion-exclusion
LBP	lower boundary point
UBP	upper boundary point

NOTATION

t	mission time
V	set of network nodes
E	set of network arcs
W_a	represent the max-capacity of arc
W_n	represent the max-capacity of nodes
S	source node in the network
T	sink node in the network
I_i, J_i	intermediate nodes in the network
$\rightarrow$	unidirectional arc
$\leftrightarrow$	bidirectional arc
S_T	stochastic sequence received by the sink node T
λ	failure rate for an exponential distribution
d	the demand for the network which ensures the transformation process to be successful
L	sequence length for the stochastic approach
R	reliability of the network

I. INTRODUCTION

A *two-terminal* network often connects its two end terminals, referred to as the *source* and the *sink* respectively. This type of network is widely utilized in many systems such as computer and communication systems [1], [2], power delivery [3] and oil/gas production systems [4]. If failures occur in these systems, substantial losses are likely incurred. Since the goal is to provide reliable and better services, the analysis of network reliability is necessary for evaluating the performance of different topological designs [5]. The reliability evaluation of such networks usually deals with the connection probability between the *source* and *sink* nodes [6].

Numerous approaches are presented in the literature for predicting the reliabilities of two-terminal binary-state networks. In these networks, the components are assumed to be either fully working or fully failed. However, this Boolean assumption is not applicable to a system with a multi-state property [7-11]. A component in a multi-state network often degrades gradually due to factors such as leakage. A binary state analysis is likely to result in a faulty outcome during decision making, so further incurring penalties such as large economic losses and security risks. Hence, the investigation of the reliability of a multi-state two-terminal network is highly important to avoid the inaccuracy incurred by the Boolean assumption.

Multi-state two-terminal networks are usually considered as flow-networks consisting of arcs with independent, discrete, multi-valued capacities. For a given problem, the capacity is indicated by a non-negative integer as a flow requirement; the capacity distribution can be determined through continuous observation and forecasting. The maximum number of information units through an arc or node is referred to as the max-capacity; usually, the capacity is described by a discrete random variable taking values from 0 to the maximum capacity. For multi-state two-terminal networks, the reliability is usually referred to as the probability that the maximal flow from the *source* to the *sink* is not smaller than the so-called demand value denoted by d, i.e., $\Pr(\phi(\boldsymbol{x})) \geq d$, where $\boldsymbol{x}$ is the component state vector, and $\phi(\cdot)$ is the system structure function that determines the system state, given the state vector.

Various probabilistic methods have been proposed to analyze two-terminal networks using exact and approximate algorithms. In [13], an exact and direct algorithm is presented for analyzing a multi-state two-terminal network. An enumeration method is capable of predicting the reliability of a multi-state two-terminal network in an exact and straightforward manner through enumerating all possible combinations of arc states [12]; however, it is computationally expensive, i.e., the reliability analysis of a multi-state system is NP-hard.

Some exact but indirect approaches have also been proposed; they are based on binary-state minimal paths/cuts. A network can be decomposed into disjoint paths for deriving an accurate result. These algorithms are mostly minimal cut based [14-20]; however, a large computational complexity is still incurred. For minimal cut based approaches, all lower/upper boundary points (LBPs/UBPs), also known as multi-state minimal path/cut, must be determined in advance. For multi-state networks, multi-state minimal cut vectors (MMCVs) must be determined. Then, the network reliability is computed by state space decomposition [21], inclusion–exclusion (IE) [15], or sum-of-disjoint-product methods [19], provided that all LBPs/UBPs are derived. Unfortunately, the processes of searching for binary-state minimal paths/cuts, LBPs/UBPs, the decomposition/IE/SDP methods are all NP-hard. Similarly, the MMCV deriving process becomes cumbersome or even impossible to solve with an increase in network size. In [22], all LBPs can be searched without knowing all binary-state minimal paths in advance; however, the network is still solved in terms of two NP-hard problems. Similarly, although a reduction in complexity has been widely investigated, the reliability analysis of a multi-state system is still NP-hard [23], thus the application of probabilistic approaches is largely limited and specifically for large systems.

For a simulation-based approach such as Monte Carlo (MC), the accuracy cannot be easily established even though the results from a large sample size are often considered stable. Due to its slow convergence, a large sample size is required for achieving a high accuracy. Furthermore, for the MC simulation in [24], MMCVs need to be determined in advance. For practical applications, capacity distributions of arcs or nodes are not limited to fixed values, but varying with time (either exponentially or non-exponentially). The provision to consider such scenarios tremendously increases the complexity of the analysis.

Recently, stochastic computational approaches were proposed for reliability analysis of logic circuits [25], [26] and dynamic fault trees (DFTs) [27] using random binary bit streams. Furthermore, a stochastic analysis has been used for predicting the reliability of a binary-state two-terminal network [28]. Stochastic models are presented for imperfect arcs and nodes; non-Bernoulli sequences of random permutations of fixed numbers of 1's and 0's are utilized to encode the signal probabilities. The use of non-Bernoulli sequences leads to an efficient evaluation; the accuracy is shown to be high if a reasonable sequence length is used.

In this paper, a stochastic analysis of a multi-state two-terminal network is proposed. A randomly permuted sequence of fixed numbers of multiple values [29] is utilized for modeling the multi-state property. Stochastic models are constructed for the information transition process through multi-state arcs; for an anticipated transition, the number of information units received by the *sink* should be no smaller than the pre-specified demand value d. Then, a stochastic analysis is performed to efficiently find the probability for the connection between the *source* and *sink*. Lastly, the node capacity can also be easily investigated through the proposed stochastic analysis.

The remainder of the paper is organized as follows. Section II reviews the preliminaries for multi-state two-terminal networks and the fundamentals for stochastic computation. Sections III presents the stochastic models for multi-state arcs. Section IV deals with the analysis of general multi-state two-terminal networks using the proposed approach. Various case studies are presented in Section V to validate the proposed approach. Finally, Section VI concludes the paper.

II. Preliminaries and Review

Let a multi-state two-terminal network be denoted as $G(V, E, W)$ consisting of a node set V and an arc set E. In this paper, a pre-specified demand value d is required for reliable communication between the *source* S and the *sink* T. An example of a multi-state two-terminal network is illustrated in Fig. 1.

- The node set V consists of nodes such as the *source* (S in Fig. 1), the *sink* (T in Fig. 1) and intermediate nodes (I_m in Fig. 1). The node set is denoted as $V = \{I_j | 1 \leq j \leq Num\}$, where Num indicates the total number of nodes in the considered network, thus $Num = m + 2$, where m indicates the number of intermediate nodes in the investigated network.
- The arc set E consists of arcs between the nodes in the network (either unidirectional or bidirectional); $E = \{e_i | 1 \leq i \leq N\}$, where N denotes the total number of arcs.
- Let $W_a(e_i)$ and $W_n(n_j)$ represent the max-capacity of arc i, i.e., e_i for $1 \leq i \leq N$ and the max-capacity of node j, i.e., n_j for $1 \leq j \leq Num$, respectively. Hence, for the network in Fig. 1, the capacity vectors for nodes and arcs are denoted as $\boldsymbol{W_a} = (W_a(e_1), W_a(e_2), \cdots, W_a(e_N))$ and $\boldsymbol{W_n} = \left(W_n(n_1), W_n(n_2), \cdots, W_n(n_{Num})\right)$, respectively. If the max-capacity of a node is not specified, then the number of information units through this node is assumed to be infinity; otherwise, the capability of information transition is limited by the max-capacity of a node.

A two-terminal network fails when the anticipated delivery service with a demand of d units cannot be supplied from the *source* to the *sink* through multi-state arcs [24]. Hence, for a provided capacity, once the available capacity received by the *sink* is no less than d, it is regarded as a successful communication. Thus, the *reliability* of a multi-state two-terminal network is defined as the probability of a successful communication between the *source* and the *sink* of the network.

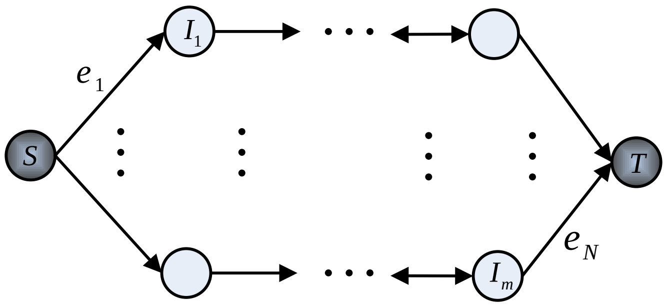

Fig. 1. A general structure for a multi-state two-terminal network. I_i, $i \in \{1, 2, \cdots, m\}$, is an intermediate node; e_j, $j \in \{1, 2, \cdots, N\}$, is an intermediate arc.

In this work, the arc or node capacities are assumed to be distributed with known probabilities; they are either fixed, or time-dependent (either exponentially or non-exponentially distributed). Some assumptions are presented as follows:

1) The network configuration is *a priori* known;
2) If the capacity distribution is time-dependent, the mission time is specified;
3) The max-capacity of an arc or a node is provided;
4) The demand value of d is pre-specified;
5) All possible states of the arcs are provided.

Stochastic computation has been proposed in 1960s for reliable circuit design [30]. In a *stochastic sequence*, a number of bits are proportionally assigned to have a specific value; for instance, a fixed number of 1's is utilized in a non-Bernoulli sequence to indicate a probability. Stochastic computation can be efficiently implemented by logic gates, while the stochastic sequence at the output encodes the (output) probability. In stochastic computation, the number of 1's in the output sequence is not deterministic, but probabilistic due to the inevitable stochastic fluctuations. However, the fluctuations can be greatly reduced by using the non-Bernoulli sequences [26].

In this paper, a multi-state two-terminal network is analyzed by stochastic computation. The non-Bernoulli sequence is generalized to a multi-state scenario, referred to as a randomly permuted sequence. As analyzed in [29], the inaccuracy can also be significantly reduced by the utilization of randomly permuted sequences; the effect of fluctuation is considerably reduced by utilizing a reasonable sequence length. For a component with M states, the probability distribution of the states is given by a vector $P = [p_M, p_{M-1}, \cdots, p_1]$, with $\sum_{i=1}^{M} p_i = 1$. This probability vector can be encoded into a randomly permuted sequence of fixed numbers of multiple values [29]. Fig. 2(a) illustrates an encoding example of a ternary signal for stochastic computation by utilizing a sequence length of 10 values. .

Fig. 2(b-e) shows the logic gates for stochastic computation used in this paper. A sequence length of 10 values is shown in Fig. 2(b)-(d) for the encoding and computing processes. A longer sequence length is usually required for achieving a higher accuracy, as shown in Fig. 2(e). A multiple-valued comparator is depicted in Fig. 2(c), by which the input sequence $S(A)$ is compared with a sequence of 1s. For the kth bit, if $S(A)_k \geq 1$, then $S(Out)_k = 1$; otherwise, $S(Out)_k$ is 0. The multiplexer in Fig. 2(e) computes the weighted sum with the output determined by the inputs and the distributions of 0s and 1s in the control sequence.

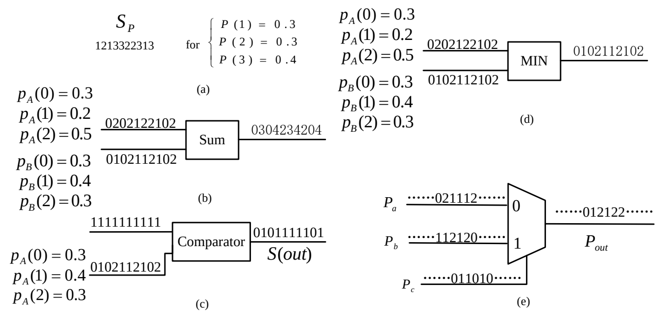

Fig. 2 (a) A stochastic encoding example of a ternary signal by utilizing a sequence length of 10 values [29]. (b) A sum module (i.e., adder) with randomly permuted sequences as inputs; (c) A comparator module with the input sequence being compared with a sequence of 1s; (d) A minimum module with randomly permuted sequences as inputs; (e) A multiplexer with the output sequence affected by the control sequence.

III. Proposed Stochastic Models for Multi-state Arcs

In a two-terminal network, the anticipated information, e.g., the telecommunication data in a wireless network or gas for a supply network, is delivered from the *source* to the *sink*. Usually, a demand value of d is specified a priori. For each transition, if the number of information units received by the *sink* is equal or larger than d, then this delivery is regarded as a reliable transition; hence, the reliability of a multi-state two-terminal network is considered as the probability of reliable transitions.

As presented in Fig. 1, a general two-terminal multi-state network consists of two types of nodes, i.e., (1) a *source* or *sink* that only has outer or inner arcs, as in Fig. 3(a); and (2) intermediate nodes that usually have outer and inner arcs, as in Fig. 3(b). An arc connecting two intermediate nodes can be either bidirectional or unidirectional, while an arc connecting an intermediate node with the *source* or *sink* is usually unidirectional.

A. *A stochastic model for nodes with unidirectional multi-state arcs*

A unidirectional arc usually exists between either the *source* S and an intermediate node I_i, or an intermediate node I_i and the *sink* T; an unidirectional arc may also exist between two intermediate nodes. Fig. 3(a) illustrates a node A with two unidirectional multi-state inner-arcs (where A can be either the *sink* T or intermediate nodes with unidirectional arcs). For the multi-state arc e_i (or e_j), the capacity distribution is assumed to be given by a probability vector $\boldsymbol{p}(e_i)$ (or $\boldsymbol{p}(e_j)$). The elements in the probability vector can be either fixed or varying with time. If the capacity of node is not specified, then the stochastic model for the node with unidirectional multi-state arcs in Fig. 3(a) is shown in Fig. 3(b).

Let $S(B)$ and $S(C)$ represent the output stochastic sequences for nodes B and C respectively. Let $S(\boldsymbol{p}(e_i))$ and $S(\boldsymbol{p}(e_j))$ denote the randomly permuted sequences encoding the capacity for the state probability distributions $\boldsymbol{p}(e_i)$ and $\boldsymbol{p}(e_j)$ for arcs

e_i and e_j, respectively. Then, for the kth trial, the states (i.e., the numbers of information units) of nodes B and C are represented as $S(B)_k$ and $S(C)_k$ respectively; $S(\boldsymbol{p}(e_i))_k$ and $S(\boldsymbol{p}(e_j))_k$ denotes the real capacities of arcs e_i and e_j for the kth trial. Then, the numbers of information units that can be transformed through arcs e_i and e_j are given as $S'(p(e_i))_k$= MIN($S(p(e_i))_k$, $S(B)_k$) and $S'(p(e_j))_k$ = MIN($S(p(e_j))_k$, $S(C)_k$), respectively. Thus, the anticipated number of information units that can be transformed through a specific arc is limited by the arc capacity.

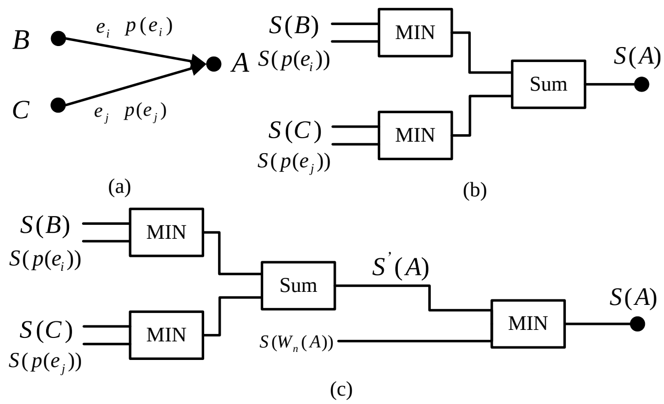

Fig. 3. (a) An illustration of node A with multi-state unidirectional arcs; (b) A stochastic model for node A without the incorporation of max-capacity; (c) A stochastic model for node A with the incorporation of max-capacity .

The units of information received by node A is equal to the sum of information units from the inner-arcs connected with it; for example, for the k th trial, $S(A)_k = sum(S'(p(e_i))_k, S'(p(e_j))_k)$, as depicted in Fig. 3(b). This can be efficiently implemented by a sum module (or adder) with randomly permuted sequences. As aforementioned, if the max-capacity of node A is not specified, then the capability of the information transmission through node A is assumed to be infinity. Thus, for the kth trial, the kth bit of $S(A)$ can be obtained according to the previous analysis; then, a stochastic sequence $S(A)$ can be derived if the simulation is performed for a number of times. $S(A)$ encodes the signal probability distribution of node A, so the signal probabilities for A can be obtained by analyzing $S(A)$.

If the max-capacity of A, i.e., $W_n(A)$, is specified, the number of information units (greater than this specified max-capacity) cannot be processed through A. Hence, the information units at A are limited by the provided max-capacity value. For the kth trial, the number of information units received by A is calculated as $S'(A)_k = sum(S'(p(e_i))_k, S'(p(e_j))_k)$. Let $S'(A)_k$ denote the k th bit of $S'(A)$. Thus, if $S'(A)_k \geq S(W_n(A))_k$, then $S(W_n(A))_k$ is selected as the output of A, i.e., $S(A)_k = S(W_n(A))_k$, because the information units being sent from A are limited by its max-capacity. Otherwise, $S(A)_k = S'(A)_k$. This can be efficiently implemented by the MIN module in Fig. 3(c).

The stochastic model in Fig. 3(b) accurately implements the information transmission process for a node with unidirectional arc. If the max-capacity property of a node is specified, then this can be dealt with by using the stochastic model in Fig. 3(c). The stochastic model for node A with two unidirectional inner-arcs can then be generalized to a general scenario, i.e., a node with multiple unidirectional inner-arcs.

B. *Stochastic analysis of nodes with bidirectional arcs*

A bidirectional arc usually exists between two intermediate nodes, e.g., I_i and I_j, as shown in Fig. 4(a). The bidirectional arc between nodes I_i and I_j is represented by L_{ij} ; the bidirectional arc can equivalently be represented as in Fig. 4(b). Arc L_{ij} indicates the connection from I_i to I_j , while L_{ji} represents the opposite direction. Due to limitations in actual capacity [31], only a single-directional flow is adopted for network realization, because the direction is determined by comparing the actual inner information units for nodes I_i and I_j. Let $S\_in(I_i)$ and $S\_in(I_j)$ denote the actual inner information units received by I_i and I_j respectively. For the kth trial, if $S\_in(I_i)_k \geq S\_in(I_j)_k$, then the information flow is transformed from I_i to I_j; otherwise, the flow is in the opposite direction. The stochastic model for the transition process between I_i and I_j in Fig. 4(a) is further implemented by the stochastic models in Fig. 4(c) and Fig. 4(d) for different flow directions.

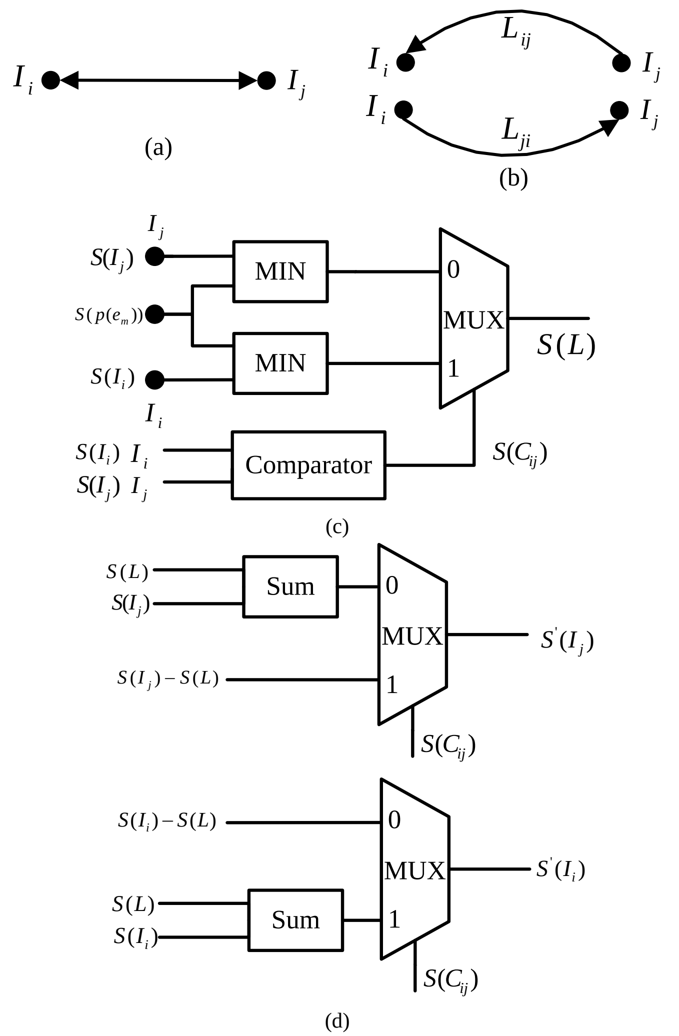

Fig. 4. (a) Illustration of two intermediate nodes I_i and I_j connected by a bidirectional arc; (b) Equivalent representation of the bidirectional arc by using

unidirectional arcs; (c) Stochastic model for deriving the information transit direction through the bidirectional multi-state arc; (d) Derivation of the sequences encoding the information units for nodes I_i and I_j.

Let $S(p(e_m))$ denote the stochastic sequence generated for the state probability distribution of the multi-state arc L_{ij} (denoted as e_m for simplicity). For the kth trial, the state of the multi-state arc is denoted by $S(p(e_m))_k$. As analyzed previously, the transition process can only be performed in one direction, i.e. either from I_i to I_j or vice versa. If the direction is from I_i to I_j, i.e., $S(I_i)_k > S(I_j)_k$, then the information through e_m is represented as $S(L)_k$= MIN($S(p(e_m))_k$, $S(I_i)_k$) because the information units to be transformed are limited by the capacity of arc e_m. Otherwise, $S(L)_k$ = MIN($S(p(e_m))_k$, $S(I_j)_k$). This is efficiently implemented by utilizing the 2-to-1 multiplexer in Fig. 4(c) with the output sequence of a comparator as the control sequence, i.e., $S(C_{ij})$. $S(C_{ij})$ is obtained by comparing the sequences for nodes I_i and I_j.

The state of node I_j is derived as $S'(I_j)_k = S(L)_k + S(I_j)_k$, while the remaining information units for node I_i are equal to $S'(I_i)_k = S(I_i)_k - S(L)_k$ with the information sent from I_i to I_j excluded. Otherwise, the information transmission process is performed in the opposite direction; hence, $S'(I_i)_k = S(L)_k + S(I_i)_k$, while the information units for node I_j are equal to $S'(I_j)_k = S(I_j)_k - S(L)_k$. Therefore, $S(L)$ can be derived as per the previous analysis.

In general, if the state probability distribution of the multi-state arc L_{ij} is provided, then the corresponding stochastic sequence $S(p(e_m))$ can be obtained through stochastic encoding. Thus, the stochastic model in Fig. 4(d) accurately derives the sequences for nodes connected by a bi-directional arc.

C. Requirements for the flow of nodes

For arcs connected with the *sink* in Fig. 5 (a), let $S(e_{Ti})$ denote the stochastic sequence received by the *sink* through e_{Ti}, where $i \in \{1, 2, \cdots, j\}$. Hence, the sequence received by the *sink* T is given by $S(T) = \sum_{i=1}^{j} S(e_{Ti})$.

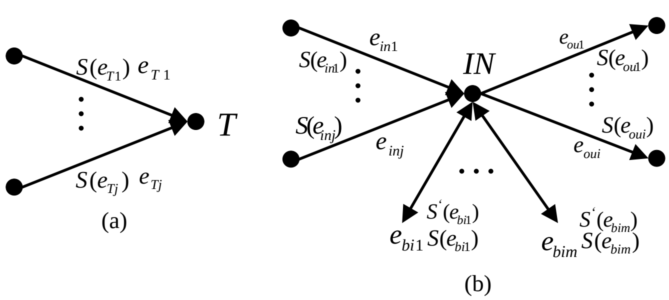

Fig. 5. (a) A *sink* node T with j inner-arcs (unidirectional), i.e., $e_{T1} \cdots e_{Ti}$; (b) An intermediate node IN with j inner-arcs (unidirectional), i outer-arcs (unidirectional) and m bidirectional-arcs.

For the intermediate node IN with multiple arcs as in Fig. 5 (b), there are j inner-arcs ($e_{in1} \cdots e_{inj}$), i outer-arcs ($e_{ou1} \cdots e_{oui}$) and m bidirectional arcs ($e_{bi1} \cdots e_{bim}$). As per the previous analysis, only one direction is possible at a time for the bidirectional arc. Once its direction is established, the inner and outer arcs are determined for the stochastic analysis. Let $S(e_{in})$ and $S(e_{ou})$ denote the information sequences for the intermediate node IN through the inner-arc and the outer-arc respectively. For the kth trial, the input and output units can be represented as $S(e_{in})_k$ and $S(e_{ou})_k$. For any intermediate node, the total number of information units received should be equal to the number of units sent from this node; hence, the restrictive condition of $S(e_{in})_k$ = $S(e_{ou})_k$ must also be met.

Furthermore, for any arc e_l in the investigated network, let $S(e_l)_k$ denote the kth bit processed through arc e_l, then $0 \leq S(e_l)_k \leq W_a(e_l)$, where $W_a(e_l)$ indicates the max-capacity of arc e_l.

D. A stochastic model to process the sequence for the sink node

For reliable transmission, the available capacity from the *source* to the *sink* should be no less than the pre-specified demand d, thus a comparator is necessary to process the information units received by the *sink*. This process can be efficiently implemented by the adoption of a comparator.

For the comparator, the two input sequences are referred to as $S(T)$ and $S(d)$; here, $S(T)$ indicates the stochastic sequence received at the *sink* T, while $S(d)$ is a sequence with each bit equals to d. The output sequence of the comparator is $S(system)$. For the jth trial (or bit), $S(T)_j$ represents the available information units received by the sink T; if $S(T)_j \geq S(d)_j$, then $S(system)_j = 1$ indicates a successful transmission; otherwise, it is set to be 0. Hence, the system reliability is given by

$$R = \frac{\sum_{i=1}^{L} S(system)}{L} \tag{1}$$

where L is the sequence length of stochastic computation.

IV. Analysis Procedure for a General Multi-state Two Terminal Network

A procedure for evaluating the reliability of a general multi-state two-terminal network is as follows:

(1) Construct a stochastic model for the investigated two-terminal network. The stochastic models in Fig. 4(b) and Fig. 4(c) are applied for the established unidirectional or bidirectional arcs.

(2) The state distribution probabilities (i.e., the capacity distribution) for the arcs at a specified mission time are determined by the provided parameters.

(3) Encode the obtained probability distribution into randomly permuted sequences for different arcs; if the capacity properties of nodes are specified, then the encoding process can be performed. A fixed number of each value is assigned in the sequence to indicate the probability distribution, as shown in Fig. 2(a).

(4) Propagate the obtained randomly permuted sequences in step (3) through the constructed stochastic model for the investigated multi-state two-terminal network.

(5) Derive the stochastic sequence at the *sink* node.

(6) The stochastic model for the comparator is utilized to process the sequence obtained at the *sink* node to find the reliability;

(6) The reliability of the multi-state two-terminal network with a demand d can be derived as per Equation (1).

The aforementioned procedures is given as the flow chart in Fig. 8.

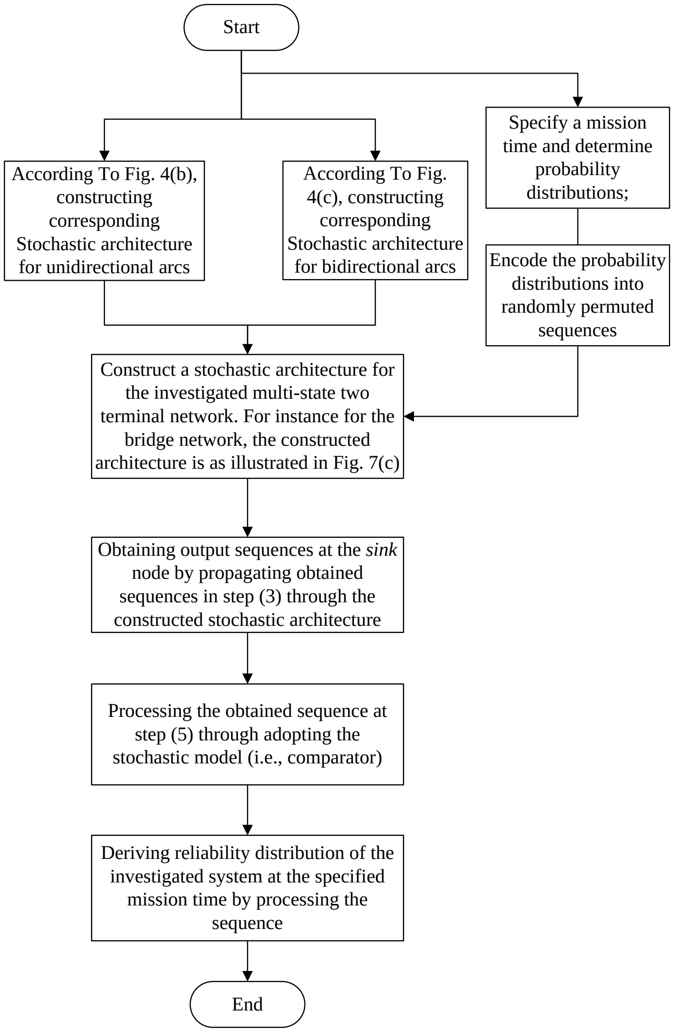

Fig. 6 A flowchart for the evaluation procedure of a general multi-state two terminal network by stochastic analysis.

V. CASE STUDIES

In this section, several cases are considered to assess the efficiency of the proposed stochastic approach. The accuracy of the stochastic approach is compared with Monte Carlo simulation. Different scenarios are investigated; all simulations are run on a computer with a 3.50 GHz E5-1620 microprocessor and a 16 GB memory.

Example 1: A bridge network [24] is shown in Fig. 7(a). This network consists of four nodes (2 intermediate nodes, A and B, a source S, and a sink T) and five arcs (4 unidirectional arcs and 1 bidirectional one). This is the simplest example of a multi-state two-terminal network.

An equivalent representation of the bridge network in Fig. 7(a) is illustrated in Fig. 7(b). It is analyzed to validate the proposed methodology. The max-capacity of each node is assumed to be infinity for this network and the demand is set to 3. So, the objective is to find the probability that the number of message units received by the *sink* is greater than or equal to 3 units. Table 1 shows the different probabilities assigned to the corresponding arc states [24].

Table 1 Arc state parameters for Example 1 [24].

Scenario 1								
Arc	States				Probability (p_{ij})			
e_1	0	1	2	3	0.050	0.025	0.025	0.900
e_2	0	1	2		0.025	0.025	0.950	
e_3	0	1			0.050	0.950		
e_4	0	1			0.020	0.980		
e_5	0	1	2		0.075	0.025	0.900	

Following the general procedure in Fig. 6, Example 1 is analyzed by:

(1) constructing the stochastic model in Fig. 7(c) by using the models in Fig. 4 and Fig. 5 for the unidirectional arcs, SA, SB, AT and BT, or the bidirectional arch, AB

(2) encoding the state distribution probabilities (i.e., the capacity distribution) in Table 1 for the arcs into randomly permuted sequences;

(3) propagating the obtained stochastic sequences through the stochastic model in Fig. 7(c);

(4) Obtaining the stochastic sequence at the *sink* node, T;

(5) using the stochastic model for the comparator to process the sequence obtained at the *sink* node for deriving the reliability.

(6) The reliability of the multi-state two-terminal network with a demand d is derived as per Equation (1).

Let S_{AB_bi} denote the sequence representing the messages received through the bidirectional arc. Let S_{inA_uni} and S_{outA_uni} denote the inner and outer summations of the sequences for node A, respectively, through the remaining unidirectional arcs; similar notations are applicable to S_{inB_uni} and S_{outB_uni}. Then by applying the stochastic model of Fig. 4(c), the stochastic sequences for the information units transmitted through the bidirectional arc can be determined. For the kth trial, the direction of the bidirectional arc is determined by comparing the kth bits of sequences S_{inA_uni} and S_{inB_uni}. For example, if $(S_{inA\_uni})_k > (S_{inB\_uni})_k$, then the message flow is from A to B. Otherwise, the transmission process is completed in the opposite direction. The stochastic model of Fig. 4(d) can be utilized to determine the states of nodes A and B.

The network is analyzed by different approaches. The obtained results are reported in Table 2; L and N denote the simulated sequence length for the stochastic approach and the simulation runs for MC. In this case, the exact reliability is 0.8310 as obtained by the inclusion/exclusion method in [16]. For a large number of simulation runs or a large sequence length, the results by MC simulation or the stochastic approach follow approximately a Gaussian distribution. At each point, simulation is performed 30 times to derive the mean and variance. Table 2 compares the mean and variance for the reliabilities found using the stochastic analysis and MC simulation. It can be seen that the obtained reliability is close to the accurate value, which shows that the stochastic approach can evaluate a network with a high accuracy.

For a confidence level of 95%, the error is calculated as $E = \frac{z_c}{\mu}\sqrt{\frac{v}{m}}$, where μ and v denote the accurate mean and the

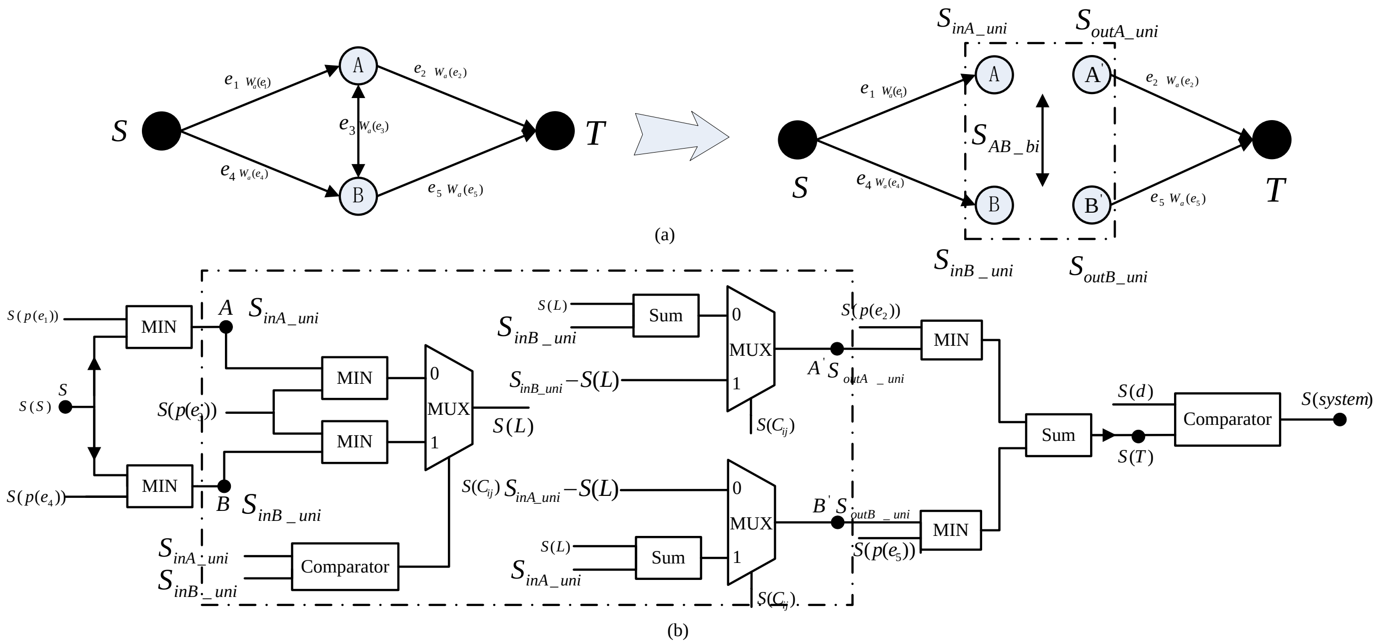

Fig. 7. (a) A bridge network [24]; (b) Equivalent representation of the network in (a); (c) Stochastic model for the bridge network.

Table 2 Mean and variance for the reliability of the multi-state bridge network, obtained by different approaches (each simulation is performed 30 times).

Algorithm in [16]	**Reliability**	0.8310	**Avg. (s)**	0.0013
Stochastic computation	***L***	**Reliability**	**Variance**	**Avg. (s)**
	1k	0.8290	1.541×10^{-5}	0.0009
	10k	0.8302	1.130×10^{-6}	0.0069
	100k	0.8307	1.087×10^{-7}	0.0518
Monte Carlo	***N***	**Reliability**	**Variance**	**Avg. (s)**
	1k	0.8270	1.408×10^{-4}	0.0025
	10k	0.8241	1.632×10^{-5}	0.0132
	100k	0.8299	1.163×10^{-6}	0.1381

variance of the distribution of the results respectively, m denotes the number of simulation runs or sequence length, and the parameter z_c is equal to 1.96. It indicates that the error decreases by increasing m. For m = 10,000, the standard deviations for the stochastic analysis and the MC simulation are found to be 2.507×10^{-6} and 9.528×10^{-5}, respectively. Therefore, the obtained error for the stochastic analysis is smaller than MC simulation at a confidence level of 95%. Thus, the stochastic analysis is more accurate than MC simulation. If an error is specified, then the sequence length to meet the error requirement can be determined.

For assessing the computing efficiency, the average run time (Avg.) is also shown in Table 2. For the algorithm in [16], all MMCVs are identified as a prior, however the required time for determining the MMCV is not included. As can be seen, the stochastic analysis is more efficient than MC simulation for the same sequence length or number of simulation runs. Compared with the algorithm in [16], the efficiency of stochastic analysis is also comparable with an acceptable accuracy (for L = 10,000 with a relative error 0.09%).

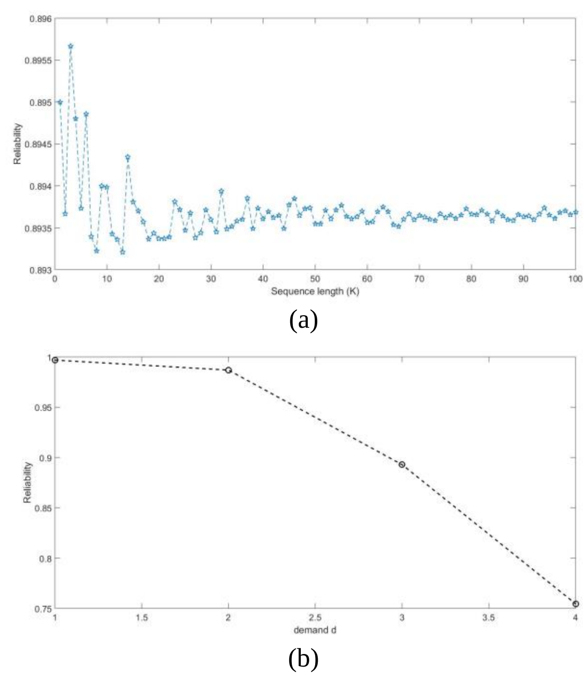

Fig. 8 (a) Convergence of the reliability obtained by stochastic analysis for the network in Fig. 7; (b) Illustration of the relationship between the demand *d* and reliability, obtained by the stochastic analysis with a sequence length of 100k bits.

As shown in Table 2, the accuracy of the stochastic approach is affected by the sequence length; a more accurate result can be obtained by increasing the sequence length (as indicated by the decreasing variance). This is also validated by Fig. 8(a) in which the simulation results converge with the increase of sequence length. The stochastic sequence length is determined by a tradeoff between accuracy and efficiency. However, the stochastic approach provides a relatively accurate estimate of the reliability of a two-terminal network with a reasonable sequence length. The relationship between reliability and demand value is further presented in Fig. 8(b); so the system reliability is reduced if a larger demand value is specified.

Example 2: As shown in Fig. 9, this network consists of 12 nodes (a *source*, S, a *sink*, T and 10 intermediate nodes) and 21 unidirectional arcs. The arcs could be in one of three states (i.e., 0, 3, 5). The state distribution is reported in Table 3. According to [32], 669 MMCVs need to be determined in advance. Thus, it is difficult to use an exact approach to analyze a network of such size. Therefore, the reliability can be estimated by using the proposed stochastic analysis and MC simulation. The results are shown in Table 4, in which L denotes the simulated sequence length for the stochastic analysis and the average run time (Avg.) indicates the efficiency of the stochastic analysis.

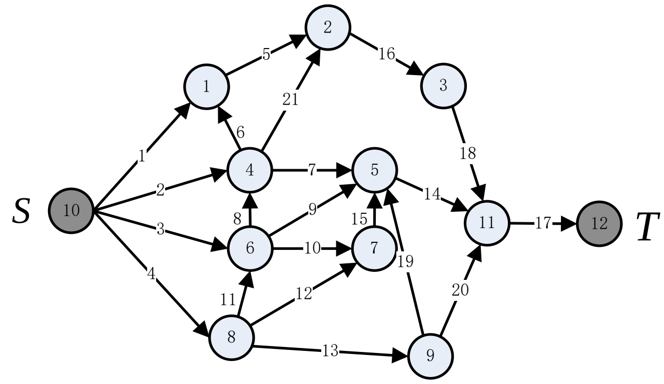

Fig. 9 The two-terminal network of [24].

In Table 4, the efficiency and the accuracy are indicated by the average run time and variance in reliability, respectively. The stochastic analysis can predict the reliability more accurately than MC simulation, as indicted by the smaller variance. The stochastic analysis is also more efficient than MC simulation, as indicated by the shorter simulation time. For a different threshold as shown in Table 4, the average run time varies slightly with the demand value. This is incurred by the different simulation runs; however, the threshold has almost no effect on the run time.

The approximate reliability obtained by the stochastic analysis is further compared with the bounds obtained by MESP [32] and MLQ [33]. The respective absolute relative errors are reported in Table 5 for different cases. For the relative error, the stochastic analysis finds the reliability at an acceptable inaccuracy. For the three cases in Table 3, the relative error for Case 2 is generally smaller than the other two

Table 3 Arc state distribution for different cases for the network in Fig. 9 [24].

Arc	*States*			*Case 1*			*Case 2*			*Case 3*		
				State Probability (p_{ij})			*State Probability* (p_{ij})			*State Probability* (p_{ij})		
1	0	3	5	0.1163	0.0616	0.8221	0.0939	0.0288	0.8773	0.0457	0.1129	0.8414
2	0	3	5	0.1624	0.1224	0.7152	0.0199	0.0017	0.9784	0.1614	0.0474	0.7912
3	0	3	5	0.2014	0.0900	0.7086	0.0293	0.0208	0.9499	0.0448	0.0729	0.8823
4	0	3	5	0.0689	0.1155	0.8156	0.0606	0.0417	0.8977	0.0982	0.0039	0.8979
5	0	3	5	0.1863	0.1366	0.6771	0.0188	0.0076	0.9736	0.1820	0.1068	0.7112
6	0	3	5	0.2244	0.0214	0.7542	0.0995	0.0328	0.8677	0.0190	0.1769	0.8041
7	0	3	5	0.2221	0.1334	0.6445	0.0072	0.0325	0.9603	0.0159	0.1230	0.8611
8	0	3	5	0.1265	0.0762	0.7973	0.0698	0.0337	0.8965	0.1198	0.0032	0.8770
9	0	3	5	0.2993	0.0343	0.6664	0.0852	0.0483	0.8665	0.1082	0.0007	0.8911
10	0	3	5	0.3016	0.0813	0.6171	0.0540	0.0498	0.8962	0.0153	0.1136	0.8711
11	0	3	5	0.2385	0.0785	0.6830	0.0572	0.0352	0.9076	0.0562	0.0778	0.8660
12	0	3	5	0.3460	0.0268	0.6272	0.0191	0.0375	0.9434	0.0673	0.1711	0.7616
13	0	3	5	0.3511	0.0441	0.6048	0.0690	0.0272	0.9038	0.1830	0.1752	0.6418
14	0	3	5	0.0326	0.0182	0.9492	0.0672	0.0017	0.9311	0.0410	0.1582	0.8008
15	0	3	5	0.0231	0.1268	0.8501	0.0197	0.0118	0.9685	0.0309	0.0766	0.8925
16	0	3	5	0.0373	0.0830	0.8797	0.0457	0.0066	0.9477	0.0245	0.1499	0.8256
17	0	3	5	0.0222	0.0192	0.9586	0.0490	0.0378	0.9132	0.0131	0.0753	0.9116
18	0	3	5	0.0052	0.0411	0.9537	0.0243	0.0082	0.9675	0.0347	0.0175	0.9478
19	0	3	5	0.3935	0.0625	0.5440	0.0830	0.0465	0.8705	0.0558	0.1944	0.7498
20	0	3	5	0.0650	0.0457	0.8893	0.0314	0.0137	0.9549	0.1740	0.1876	0.6384
21	0	3	5	0.1260	0.0495	0.8245	0.0503	0.0356	0.09141	0.1905	0.0572	0.7523

Table 4 Mean and variance for the reliability of the network in Example 2, obtained by the stochastic approach and Monte Carlo (MC) simulation for different cases (the simulation is performed for 30 times), with average run time (Avg.) also shown.

		Case 1			**Case 2**			**Case 3**		
	L/N	**Reliability (Mean)**	**variance**	**Avg. (s)**	**Reliability (Mean)**	**variance**	**Avg. (s)**	**Reliability (Mean)**	**variance**	**Avg. (s)**
Stochastic analysis d = 5	10k	0.9384	6.4421×10^{-7}	0.0166	0.9078	2.9463×10^{-7}	0.0171	0.8876	9.5905×10^{-7}	0.0168
	100k	0.9380	2.5084×10^{-7}	0.2960	0.9079	5.4950×10^{-8}	0.1020	0.8878	2.4366×10^{-7}	0.1020
	1000k	0.9381	1.8174×10^{-8}	1.7824	0.9079	5.3667×10^{-9}	1.9151	0.8877	2.2634×10^{-8}	1.8017
MC Simulation d = 5	10k	0.9379	4.7775×10^{-6}	0.0254	0.9090	6.1207×10^{-6}	0.0294	0.8870	1.68725×10^{-5}	0.0275
	100k	0.9381	4.2368×10^{-7}	0.2552	0.9076	1.4242×10^{-6}	0.2915	0.8875	1.0975×10^{-6}	0.3432
	1000k	0.9381	4.7928×10^{-8}	3.2966	0.9079	8.5410×10^{-8}	3.3464	0.8877	1.3801×10^{-7}	3.3953
Stochastic analysis d = 3	10k	0.9727	6.3818×10^{-7}	0.0168	0.9466	3.5187×10^{-7}	0.0169	0.9829	2.6892×10^{-7}	0.2020
	100k	0.9727	4.2585×10^{-7}	0.1021	0.9467	2.3317×10^{-8}	0.1033	0.9827	4.4653×10^{-8}	0.1016
	1000k	0.9726	5.3776×10^{-8}	1.8026	0.9467	3.3812×10^{-9}	1.8524	0.9828	2.3960×10^{-9}	1.8155
MC Simulation d = 3	10k	0.9724	3.5335×10^{-6}	0.0261	0.9465	3.4947×10^{-6}	0.0249	0.9832	2.3883×10^{-6}	0.0269
	100k	0.9725	2.9296×10^{-7}	0.2855	0.9466	4.3404×10^{-7}	0.3281	0.9826	1.5714×10^{-7}	0.3362
	1000k	0.9727	3.1540×10^{-8}	2.8291	0.9468	5.1360×10^{-8}	2.5509	0.9828	2.1847×10^{-8}	3.4313

Table 5 Accuracy and performance of stochastic analysis, with L = 100k bits and d = 5.

Case	Stochastic Analysis	Bounds		Relative error		Avg.(s)
		MESP	MLQ	MESP	MLQ	
1	0.9381	0.9308	0.9308	0.783%	0.732%	0.1050
2	0.9079	0.9105	0.9105	0.286%	0.286%	0.1016
3	0.8875	0.8911	0.8911	0.406%	0.409%	0.1034

cases. This validates the claim in [24] that if the components' state probabilities are high for a perfect performance, the proposed simulation-based approach predicts more accurate results with a smaller relative error. The average run time (Avg.) of the stochastic analysis is also shown in Table 5. The simulation time is mainly affected by the sequence length, but it is not significantly affected by a change in the arc state distribution.

VI. CONCLUSION

In this paper, a stochastic computational approach is proposed to evaluate the reliability of a general multi-state two-terminal network. The reliability denotes the probability of connectivity between the *source* and *sink* nodes. Both arcs and nodes at different capacity distributions have been considered. Random permutations have been used to improve the computational efficiency and accuracy of the stochastic approach. Due to the stochastic encoding, a capacity distribution, either fixed or varying with time, and the arcs, either unidirectional or bidirectional, can be stochastically modeled for the information transmission process. A stochastic model is then constructed through combinations of logic gates. Validated by two case studies, the proposed stochastic approach can efficiently and accurately predict the reliability of a general two-terminal network under a capacity distribution.

Acknowledgement:
This work was supported in part by the National Science Foundation for Young Scientists of China (Grant No. 61601371).

REFERENCES

[1] K. K. Aggarwal, *et al*. A simple method for reliability evaluation of a communication system. *IEEE Trans Commun* 1975; COM-23: 563–5.

[2] M. A. Samad. An efficient algorithm for simultaneously deducing MPs as well as cuts of a communication network. *Microelectron Reliab* 1987; 27: 437–41.

[3] W. J. Ke, S. D. Wang. Reliability evaluation for distributed computing networks with imperfect nodes. *IEEE Trans Reliab* 1997; 46: 342–9.

[4] T. Aven. Availability evaluation of oil/gas production and transportation systems. *Reliab Engng* 1987;18: 35–44.

[5] A. H. S. Ang, W. H. Tang. *Probability Concepts in Engineering Planning and Design. Decision, Risk, and Reliability*, vol. II. John Wiley and Sons, New York, 1990.

[6] K. K. Aggarwal and K. B. Misra. A Fast Algorithm for Reliability Evaluation. *IEEE Trans Reliab*1975; 24(1): 83–85.

[7] Natvig B, Sørmo S, Holen A, Høgasen G. Multistate reliability theory—a case study. *Adv Appl Probab* 1986; 18: 921–32.

[8] R. Boedigheimer, K. Kapur. Customer-driven reliability models for multistate coherent systems. *IEEE Trans Reliab* 1994; 43(1):46–50.

[9] G. Levitin, A. Lisnianski, H. Ben-Haim, D. Elmakis. Redundancy optimization for series-parallel multi-state systems. *IEEE Trans Reliab* 1998; 47(2):165–72.

[10] J. E. Ramirez-Marquez, D. Coit. A heuristic for solving the redundancy allocation problem for multistate series-parallel systems. *Reliab Eng Syst Saf* 2003; 83(3):341–9.

[11] R. Billinton, W. Zhang. State extension for adequacy evaluation of composite power systems—applications. *IEEE Trans Power Syst* 2000; 15(1):427–32.

[12] S. Tsukiyama, I. Shirakawa, H. Ozaki and H. Ariyoshi, An Algorithm to Enumerate All Cut Sets of a Graph in Linear Time per Cutset, *Journal of the Association for Computing Machinery* 1980, 27(4): 619–632,.

[13] C. C. Jane, Y. W. Laih. A practical algorithm for computing multi-state two-terminal reliability. *IEEE Trans Reliab* 2008, 57: 295-302.

[14] C. C. Jane, J. S. Lin, J. Yuan. On reliability evaluation of a limited-flow network in terms of minimal cutsets. *IEEE Trans Reliab*1993, 42, 354–361.

[15] J. S. Lin, C. C. Jane, J. Yuan. On reliability evaluation of a capacitated-flow network in terms of minimal pathsets. Networks 1995, 25, 131–138.

[16] Y. K. Lin. A stochastic model to study the system capacity for supply chains in terms of minimal cuts. *International Journal of Production Economics* 2010, 124(1): 181-187.

[17] Y. K. Lin. Evaluate the performance of a stochastic-flow network with cost attribute in terms of minimal cuts. *Reliab Eng Syst Saf* 2006, 91(5): 539-545.

[18] W. C. Yeh, C. Bae, C. L. Huang. A new cut-based algorithm for the multi-state flow network reliability problem. *Reliab Eng Syst Saf* 2015, 136: 1-7.

[19] M. J. Zuo, Z. Tian, H. Z. Huang. An efficient method for reliability evaluation of multistate networks given all minimal path vectors. *IIE Transactions* 2007, 39, 811– 817.

[20] C. Tanguy. Exact two-terminal reliability of some directed networks. In Proc. Design and Reliable Communication Networks (DRCN), La Rochelle, France, 2007, pp. 155–162.

[21] T. Aven. Reliability evaluation of multistate systems with multistate components. IEEE Transaction on Reliability1985, 34, 473–479.

[22] W. C. Yeh. A novel method for the network reliability in terms of capacitated minimum-paths without knowing minimum-paths in advance. *Journal Operational Research Society* 2005, 56, 1235–1240.

[23] C. J. Colbourn. The combinatorics of network reliability. New York: Oxford University Press; 1987.

[24] J. E. Ramirez-Marquez and D. W. Coit. A Monte-Carlo simulation approach for approximating multi-state two-terminal reliability, *Reliab Eng Syst Saf* 2005; 87: 253-264.

[25] H. Aliee and HR Zarandi. A Fast and Accurate Fault Tree Analysis Based on Stochastic Logic Implemented on Field-Programmable Gate Arrays. *IEEE Trans Reliab* 2013, 62(1): 13 – 22.

[26] J. Han, H. Chen, J. Liang, P. Zhu, Z. Yang and F. Lombardi. A Stochastic Computational Approach for Accurate and Efficient Reliability Evaluation. *IEEE Trans. Computers* 2014, 63(6): 1336-1350.

[27] P. Zhu *et al*. A Stochastic Approach for the Analysis of Fault Trees with Priority AND Gates. *IEEE Trans. Reliab* 2014, 63(2): 480-494.

[28] P. Zhu, J. Han, Y. Guo and, F. Lombardi. Reliability and Criticality Analysis of Communication Networks by Stochastic Computation. *IEEE Network*, to appear, 2016.

[29] P. Zhu and J. Han. Stochastic multiple-valued gene networks. *IEEE Trans. Biomed Circ Syst* 2014, 8(1): 42–53.

[30] B. R. Gaines. Stochastic Computing Systems. Advances in Information Systems Science 1969, 2: 37-172.

[31] J Edmonds，Theoretical Improvements in Algorithmic Efficiency for Network Flow Problems. *Journal of the Association for Computing Machinery* 1972, 19(2): 248-26.

[32] J. E. Ramirez-Marquez, D. Coit, and M. Tortorella. Multistate two-terminal reliability-a cut approach. *Rutgers University IE Working Paper* 03-135, 2003.

[33] J. E. Ramirez-Marquez, D. Coit, and M. Tortorella. Bounds for Multistate Network TwoTerminal Reliability. Rutgers University IE Working Paper 03-121.