Received February 9, 2020, accepted February 23, 2020, date of publication March 2, 2020, date of current version May 20, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2977830

Reliability Modeling and Analysis of Generalized Majority Systems by Stochastic Computation

NING WANG[1], WEI FENG[2], HAILUN ZHANG[2], AND SHUMIN LI[3]
[1]School of Automobile, Chang'an University, Xi'an 710064, China
[2]School of Computer Science and Technology, Northwestern Polytechnical University, Xi'an 710072, China
[3]School of Management Engineering, Zhengzhou University, Zhengzhou 450001, China

Corresponding author: Shumin Li (lishumin@zzu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 71971030 and Grant 71940016, in part by the Natural Science Basic Research Plan in Shaanxi Province of China under Grant 2019JM-495, in part by the Foundation of He'nan Educational Committee of China under Grant 20A630033, and in part by the Fundamental Research Funds for the Central Universities (Chang'an University) under Grant 300102220203.

ABSTRACT The k-out-of-n: G(F) majority voter consists of n components (or modules) and a number of the components are required to be operating correctly for the overall system to be correct. As per the state discretization of the components, such a system is usually classified as either a binary system or a multi-state system. In practice, the operating conditions of different components may contribute differently to the operation of the entire system. In this manuscript, the k-out-of-n: G(F) majority voter is generalized as a consecutive-weighted-k-out-of-n: G(F) voter with either binary states or multiple states. To overcome the drawbacks of existing approaches, a stochastic analysis is proposed for assessing the system reliability. In the stochastic analysis, the input signal probabilities are encoded into non-Bernoulli sequences with fixed numbers of 0s and 1s for the Boolean case, or randomly permuted sequences for the multi-state scenario. By using stochastic logic, the reliability of a general system consisting of consecutive-weighted-k-out-of-n majority voters is efficiently and accurately predicted. The results are validated by an analysis of several case studies. Although the accuracy of the stochastic analysis is closely related with the employed sequence length, it is shown that a stochastic approach is more efficient than a universal generating function (UGF) method, while still retaining an acceptable accuracy.

INDEX TERMS Stochastic computation, non-Bernoulli sequence, stochastic logic, consecutive-weighted-k-out-of-n: G(F) majority voter, reliability evaluation.

I. INTRODUCTION

The k-out-of-n: G(F) majority voter [1] has been widely utilized to ensure the correct operation of computing systems for numerous critical applications, including the engine systems [2], unmanned aerial vehicles (UAVs) [3] and power systems [4], [5] in the defense and aerospace industry. Thus, the output performance distribution (OPD) is of great interest and has been extensively investigated in the technical literature [1], [6]–[9]. In particular, its reliability has been investigated to reflect the OPD. The k-out-of-n: G majority voter consists of n components (modules, units), and it correctly operates provided the minimal total weight of all good (fault-free, correct) components is not less than a pre-specified threshold k.

The associate editor coordinating the review of this manuscript and approving it for publication was Zhaojun Li.

For example, a k-out-of-n binary state system is referred to as a k-out-of-n: G binary state system if and only if at least k of the n components function. As long as the reliability of the k-out-of-n: G system can be found, the unreliability of an equivalent (n-k + 1)-out-of-n: F system can also be easily determined (where F indicates a failure). For example, it may be possible to drive a car if at least four cylinders are firing in its V8 engine. Thus, the functionality of the engine is specified by a 4-out-of-8: G system. Equivalently, a car cannot be driven if less than four cylinders fire (i.e., at least 5 cylinders are non-functioning); so, the system can also be described as a 5-out-of-8: F system [10]. Hence, the performance evaluation of a k-out-of-n: F majority voter is equivalent to the analysis of a (n-k+1)-out-of-n: G majority voter. In this paper, without loss of generality, the reliability of a k-out-of-n: G system is treated in detail.

The components of a k-out-of-n: G majority voter do not always equally contribute to the overall performance of the system. Thus, the traditional binary k-out-of-n: G majority voter has been further generalized as a weighted binary k-out-of-n: G voter [11]. In a weighted binary k-out-of-n: G system, component i is assigned a positive integer w_i, that indicates the utility (contribution) of component. Then, the total weight of all components is calculated as $w = \sum w_i$. Therefore, the system is operational if and only if $w \geqslant k$, where k is the pre-specified threshold that ensures the system correctness. If for any $i \in \{1, \cdots, \mathrm{n}\}$, $w_i = 1$, then the system is simplified as a k-out-of-n: G voter.

For the performance evaluation of a general weighted majority voter, a universal generating function (UGF) can be used to derive the exact expressions [12]; this approach is also applicable to the analysis of multi-state systems [13], [14]. However, it incurs a high computational complexity when n increases; alternatively, a recursive algorithm can be used to evaluate the reliability of a binary weighted k-out-of-n: G system [11]. The recursive approach is more efficient than the UGF analysis to evaluate a weighted binary k-out-of-n: G system, as shown in the run time comparison in [15]. Capacity loss and residual capacity in binary weighted-k-out-of-n: G systems have been investigated in [16]; a voter consisting of components with random weights has been further discussed in [8]. Furthermore, imperfect fault coverage has also been studied in k-out-of-n systems [17]–[19].

In practice, the state of a component is not limited to a binary variable; for instance, a component might have three states: totally working, partially operating (in a degraded mode) and failure, i.e. the Boolean assumption for the states is not always applicable. Therefore, a weighted binary majority voter has been generalized to a weighted multi-state voter [20], the component has more than two states and each state has a positive integer weight which indicates its contribution to the system's working state, the system is operational if and only if the total weight of consecutive working component is at least k; this topic has been widely investigated in the literature [21]–[25]. A stochastic multiple valued (SMV) approach has been proposed to efficiently predict the reliability of weighted multi-state voter with non-repairable components and dynamically repairable components, and reliabilities of components are represented by stochastic sequences [26]. The recursive approach is also applicable to the analysis of a general weighted multi-state majority voter (as presented in [15]). It is also proved to be more efficient than the UGF method for the analysis of a general weighted multi-state majority system.

Recently, stochastic analysis has been performed for evaluating logic circuits [27] and computing the reliability of dynamic fault trees (DFTs) [28]. The stochastic approach has been shown to be able to analyze DFTs consisting of non-exponentially distributed components. In [28], the stochastic analysis is performed to investigate the reliabilities of 2-out-of-3 and 3-out-of-5 binary majority voters. Furthermore, the stochastic approach has been used to analyze a multi-valued network by utilizing stochastic multi-valued logic gates [29].

Redundancy techniques have been developed to ensure high reliability and availability in dependable systems [30]–[34]. So given the occurrences of different failures, a stochastic computational model (as proposed in this work) can efficiently be employed by utilizing only slight modifications for the different failure scenarios, such as the analysis of soft errors using a stochastic model [27]. Stochastic models are also very flexible; approximate computing has been advocated for designing simpler redundant systems, while retaining acceptable error rates and improving performance metrics such as delay and power [35]. Also in this case, the analysis of an approximate system can be performed by utilizing a stochastic technique.

As the stochastic approach leads to an efficient and accurate evaluation of performance by using non-Bernoulli sequences. In this paper, the system reliability of consecutive-weighted multi-state voters is investigated through a stochastic analysis: a non-Bernoulli sequence is generalized to randomly permuted sequences of fixed numbers of values. In this manuscript, stochastic computational models are first proposed for a general k-out-of-n: G system. Then, stochastic models are respectively proposed for a consecutive-weighted binary k-out-of-n: G system and a multi-state system. For the multi-state system, randomly permuted sequences of fixed numbers of values are utilized to indicate the corresponding signal probabilities. It is shown that the reliability of a consecutive-weighted binary or multi-state k-out-of-n: G system is efficiently found by using the randomly permuted sequences. Based on the proposed stochastic architecture, the reliability evaluation of a general system consisting of various gates, such as the Priority AND (PAND) gate, the Spare gate and the majority voter, can be efficiently performed. Finally, as signal correlation is preserved by stochastic sequences, systems with repeated common components can also be analyzed. The accuracy is found to be very high by using reasonable sequence lengths in the simulation of several case studies.

The remainder of the paper is organized as follows. Section 2 reviews the preliminaries on stochastic computation. In Sections 3, the modified stochastic model for a binary-state majority voter is presented; then the stochastic model is generalized for the analysis of weighted majority voter. A stochastic model is proposed for a general consecutive-weighted multi-state majority voter. Case studies are presented in Section 4 to show the efficiency and capability of the proposed stochastic approach for analyzing different systems. Finally, Section 5 concludes the paper.

II. REVIEW AND PRELIMINARIES

A. STOCHASTIC COMPUTATION

Stochastic computation was presented for reliable circuit design in the 1960s [36]. In stochastic computation, the signal

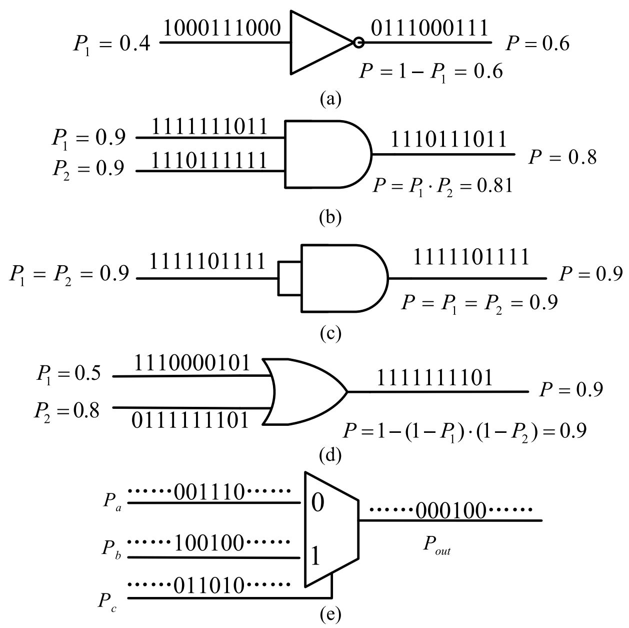


FIGURE 1. Logic gates for stochastic computation.

probability is indicated by a stochastic sequence, in which each bit is set to a specific value. Logic gates are used for stochastic computation with input signal probabilities (Figure 1). The output probability is obtained by analyzing the corresponding output sequence, i.e. Boolean logic operations are transformed into probabilistic computations in the real domain. Figure 1 shows the logic gates for the stochastic analysis employed in this paper. In Figure 1 (a), an inverter with a random binary sequence as input. In Figure 1 (b), an AND gate with s-independent inputs. In Figure 1 (c), an AND gate with totally correlated inputs. In Figure 1 (d), an OR gate with s-independent inputs. In Figure 1 (e), A 2-to-1 multiplexer inputs.

In stochastic computation, the number of 1's in the output sequence is probabilistically incurred by stochastic fluctuations [27]. To reduce the stochastic fluctuations, non-Bernoulli sequences are used to encode the initial signal probabilities [27]. In Figure 1(a-d), a sequence length of 10 bits is utilized for the stochastic encoding and computing process; a longer sequence length is usually required for higher accuracy (Figure 1(e)). The multiplexer computes the weighted sum with the output being affected by the control sequence. Correlation among signals is usually incurred by the reconvergence of fanout signals. It can be inherently handled by stochastic logic because signal dependencies are maintained and propagated (as indicated in Figure 1(c)) [27]; this is a desirable property when dealing with the common component of a system [37].

For a generalized system consisting of multi-state components, the non-Bernoulli sequence is generalized to a multi-state scenario. For simplicity, all components are assumed to have the same number of states and each component is discretized into M states. For a component with M states,

$$\underset{1213322313}{S_P} \text{ for } \begin{cases} P(1) = 0.3 \\ P(2) = 0.3 \\ P(3) = 0.4 \end{cases}$$

FIGURE 2. Stochastic encoding of a ternary signal using a sequence length of 10 values [29].

the probability distribution of each state is given by a vector $P = [p_M, p_{M-1}, \ldots, p_1]$, with $\sum p_i = 1$. This signal probability vector can be encoded into a multiple-valued stochastic sequence; hence, it is referred to as a randomly permuted sequence of fixed numbers of multiple values [29].

The use of randomly permuted sequences (as in Figure 2) in a simulation significantly reduces the amount of stochastic fluctuations and the effect of fluctuations is made negligible when using an appropriate sequence length.

B. ASSUMPTION

The following assumptions are made for a general consecutive-weighted-k-out-of-n: G majority voter.

(1) The state of a component i, i.e., x_i, is assumed to be discretized into M_i values; then $x_i \in \{1, 2, \cdots, M_i\}$ ($M_i \geqslant 2$ and $i \in \{1, 2, \cdots, n\}$). If for every i, $M_i = 2$, then the system is referred to as a binary majority voter. Otherwise, it is referred to as a multi-state majority voter.

(2) A component i has a positive integer weight, i.e., $w_i \geqslant 1$, which indicates its contribution to the system's working state. If for component i, w_i is larger than 1, then the k-out-of-n system is generalized as a weighted k-out-of-n system.

(3) The system is operational if and only if the total weight of consecutive working component is at least k; this is an a priori specified threshold.

(4) A component is working when its weight is greater than 0.

(5) All components are working at the beginning of the mission time.

(6) All states of the n components are mutually s-independent.

During the specified mission time, the components are assumed to be non-repairable [38]. For some applications, (e.g., flight control and space missions), it is difficult to repair or replace a failed component when a failure occurs. Maintenance is only performed when the system is in a specific state (such as for example an idle state and/or in a specific location), so the system has all good components at the beginning of each mission for most applications.

III. PROPOSED STOCHASTIC MODELS

The 4 models for the binary majority voters, the weighted binary majority voters, the weighted multi-state majority voters, and the consecutive-weighted multi-state majority voter are considered next; these models exploit specific stochastic properties in their analysis to assess different features for calculating the system reliability.

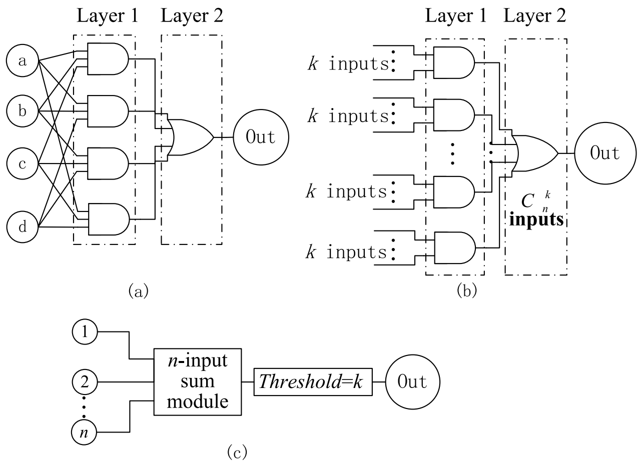


FIGURE 3. The stochastic models for majority voters.

TABLE 1. Truth table for a 3-out-of-4 binary-state majority voter.

		ab			
		00	01	11	10
cd	00	0	0	0	0
	01	0	0	1	0
	11	0	1	1	1
	10	0	0	1	0

A. THE BINARY MAJORITY VOTER

For a binary majority voter, each component has two exclusive states: working and failure, indicated by 0 and 1 respectively. As analyzed in [28], a 2-out-of-3 majority voter can be implemented by stochastic logic. The logical relationship in the system can be modeled by the combinations of the logic gates.

The stochastic models for majority voters consist of stochastic logic. For example, a stochastic model for a 3-out-of-4 majority voter is presented in Figure 3(a). The truth table is shown in Table 1. Once any three of the four inputs are equal to 1, the output of the 3-out-of-4 majority voter is 1. If the input signal probabilities are encoded by non-Bernoulli sequences, the output signal probability can be efficiently derived by analyzing the output sequence. As revealed by the results in [27], the accuracy in the system reliability for the stochastic approach increases if a longer sequence length is utilized.

Based on the stochastic model of Figure 3(a), a stochastic architecture for a general k-out-of-n majority voter is presented in Figure 3(b). The stochastic model in Figure 3(b) is classified into two layers. Layer 1 consists of a number of multiple-input AND gates. The number of inputs for the AND gate is k which is determined by the majority voter. The inputs of an AND gate are given by the possible combinations of k elements being selected from n; hence, the number of inputs for the OR gate in Layer 2 is C_n^k. If the input signal probabilities are encoded as non-Bernoulli sequences, the output sequence can be obtained through propagating the stochastic sequences in the stochastic model of Figure 3(b).

TABLE 2. The simulation results by utilizing different stochastic models.

k			1	2	3	4	5
Accurate results [7]			0.9688	0.8125	0.5000	0.1875	0.0313
Model in Figure. 3(b)	*L* (bits)	256	0.9676	0.8160	0.4944	0.1917	0.0337
		512	0.9673	0.8113	0.4991	0.1904	0.0307
		1000	0.9679	0.8135	0.5021	0.1856	0.0308
		10000	0.9694	0.8122	0.5007	0.1868	0.0309
Model in Figure. 3(c)		256	0.9697	0.8137	0.5020	0.1919	0.0306
		512	0.9663	0.8110	0.5029	0.1863	0.0320
		1000	0.9694	0.8122	0.5007	0.1868	0.0309
		10000	0.9687	0.8114	0.4997	0.1870	0.0312

By increasing min(k, n-k) or n, the number of input combinations (i.e., C_n^k) grows rapidly. To avoid the enumeration of the combinations, a modified stochastic model is presented in Figure 3(c). The inputs are accumulated by an n-input sum module (equivalent to the bit-addition of n sequences), and then the majority voting process is implemented by a threshold module. The threshold module is defined as follows: if the value is bigger than or equal to a pre-specified threshold, then the output is 1; otherwise, it is equal to 0. An example of the threshold logic operation is illustrated in Figure 4, in which the threshold is set to be 2.

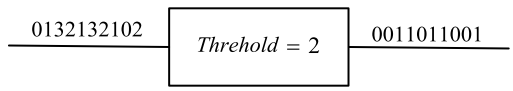


FIGURE 4. An example of the threshold logic.

The generalized stochastic models of Figure 3(b) and (c) are applied to calculate the output signal probability of a k-out-of-5 majority voter. The simulation results by utilizing different stochastic models are illustrated in Table 2 for different sequence lengths. Stochastic analysis is performed for the investigated system with different sequence lengths. The input signal probabilities are 0.5. Furthermore, the accurate results obtained by using the approach in [7] are presented in Table 2 for comparison. As per the results in Table 2, the two stochastic models are capable of evaluating general k-out-of-n majority voters by producing very close results. The stochastic fluctuation of the stochastic approach decreases with an increase of sequence length. By utilizing an appropriate sequence length L, the reliability of a general k-out-of-n majority voter is effectively and accurately found.

B. THE WEIGHTED BINARY MAJORITY VOTER

A weighted binary k-out-of-n model is generalized in [11]; the "weight" indicates the contribution of each component to the correct operational state of the system. In a weighted binary majority voter, the weight of the system is equal to the sum of the weights of the components; if a component fails, then its contribution to the system or the weight is 0.

For a weighted binary k-out-of-n majority voter, each component i has a positive integer weight, i.e., $w_i > 0$ for $I = 1, 2, \ldots, n$. The total weight of all components is w, which is calculated as $w = \sum w_i$ [15]. Therefore, k is the minimal total weight of the working components that ensures the system to be operational (note that k may be larger than n). An illustrative example for a general weighted binary k -out-of-n majority voter (k is a pre-specified threshold, which might be larger than n) is presented in Figure 5(a), while a stochastic model for the k-out-of-n majority voter is presented in Figure 5(b). The binary state of a component is first multiplied by the corresponding weight and the weight of the working components is computed by an n-input sum module; then, a threshold logic is applied to determine whether the system works correctly or not. If the input signal probability is encoded as a non-Bernoulli sequence, then the system reliability can be easily determined by analyzing the output sequence. If for any i, w_i is always equal to 1, then the weighted binary majority voter is simply reduced to a traditional binary majority voter.

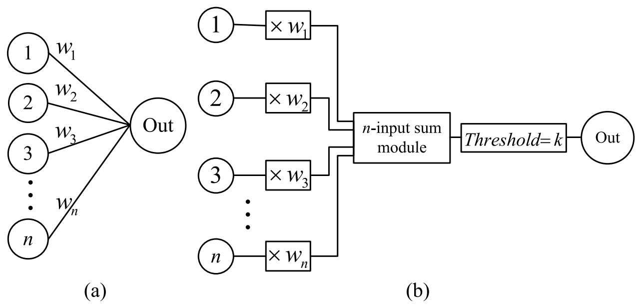


FIGURE 5. Illustration of a general weighted binary k-out-of-n majority voter (a) and a corresponding stochastic model (b).

Example 1: Consider a weighted binary 5-out-of-3: G system (Figure 6(a), here, let k be a pre-specified threshold, here $k \in \{0, 2, 4, 6, 8, 10, 12\}$); the weights for the components are 2, 6 and 4 respectively. The system is operational if and only if the total weight of the functional components is at least 5. An example is presented to use the voter in Figure 6(a) as a subsystem and to illustrate the capability of a stochastic analysis for correlated signals.

The UGF of component i is a polynomial function that relates the probability of each state to the performance of the component. The UGF defines the OPD for the investigated system. For the binary weighted k-out-of-n system, the UGF of component i is given by $U_i(z) = p_i z^{wi} + (1 - p_i)z^0$; here, p_i denotes the reliability of component i and w_i indicates the weight of component i if it is in the working state. The operator $\Omega = \Omega(U_1(z), U_2(z), \ldots, U_n(z))$ in [12] can be utilized for describing the UGF of a system, where Ω is determined by the corresponding structure of the components.

The UGFs for the input components are obtained from [15]:

$$U_1(z) = q_1z^0 + p_1z^2$$
$$U_2(z) = q_2z^0 + p_2z^6$$
$$U_3(z) = q_3z^0 + p_3z^4$$

where $q_i = 1 - p_i$ and p_i indicates the reliability of component i, $i = 1, 2, 3$.

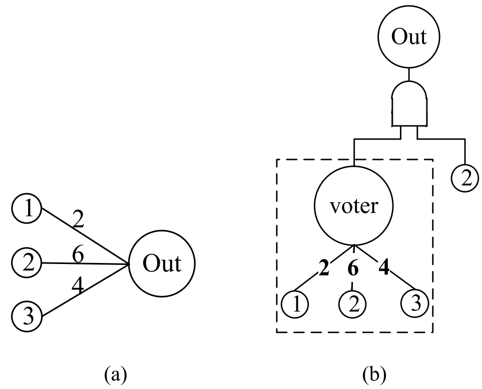


FIGURE 6. Illustration of a general weighted k-out-of-3 majority voter. (b) A system with the voter in (a) functioning as a subsystem.

Then, the exact expression obtained by the UGF analysis is [15]:

$$\begin{aligned} U_s(z) = {} & q_1q_2q_3z^0 + p_1q_2q_3z^2 + q_1q_2p_3z^4 \\ & + (q_1p_2q_3 + p_1q_2p_3)z^6 + p_1p_2q_3z^8 \\ & + q_1p_2p_3z^{10} + p_1p_2p_3z^{12} \end{aligned} \tag{1}$$

The analysis in [14] is applied to find the reliability of the system for any given threshold value. For $k = 5$, the reliability of the system is based on (1) and given by:

$$R_S(5) = p_2 + q_2p_1p_3$$

By applying the stochastic model of Figure 5(b), a stochastic analysis is performed for the investigated majority voter. To assess the accuracy of the stochastic analysis, the obtained results are compared with an accurate UGF approach. The arbitrarily generated input signal probabilities are shown in Figure 7(a); then the accurate reliability is given by (2) and plotted in Figure 7(b) (for the threshold 5). Figure 7(c) shows the absolute difference between the results obtained by a stochastic analysis and the UGF approach. The difference decreases with an increase of sequence length, thus achieving a higher accuracy.

The average run time for the stochastic analysis is compared with the recursive method in [11] and Monte Carlo (MC) simulation. The run time is obtained by a software-based simulation for the stochastic approach throughout this paper. The results are provided in Table 3, which illustrate the efficiency of a stochastic analysis. The input parameters are k and n, respectively indicating the threshold value and the total number of components. As shown in Table 3, the efficiency of a stochastic computation approach compares favorably with the recursive method. In the MC simulation, the binary input signal, either 0 or 1, is randomly generated according to the signal probability. The difference between MC and the stochastic analysis is in the input sequences. For the stochastic analysis, the sequence consists of random permutations of fixed numbers of 1s and 0s, while for the MC simulation, each sequence is approximately a Bernoulli sequence generated by using pseudorandom numbers.

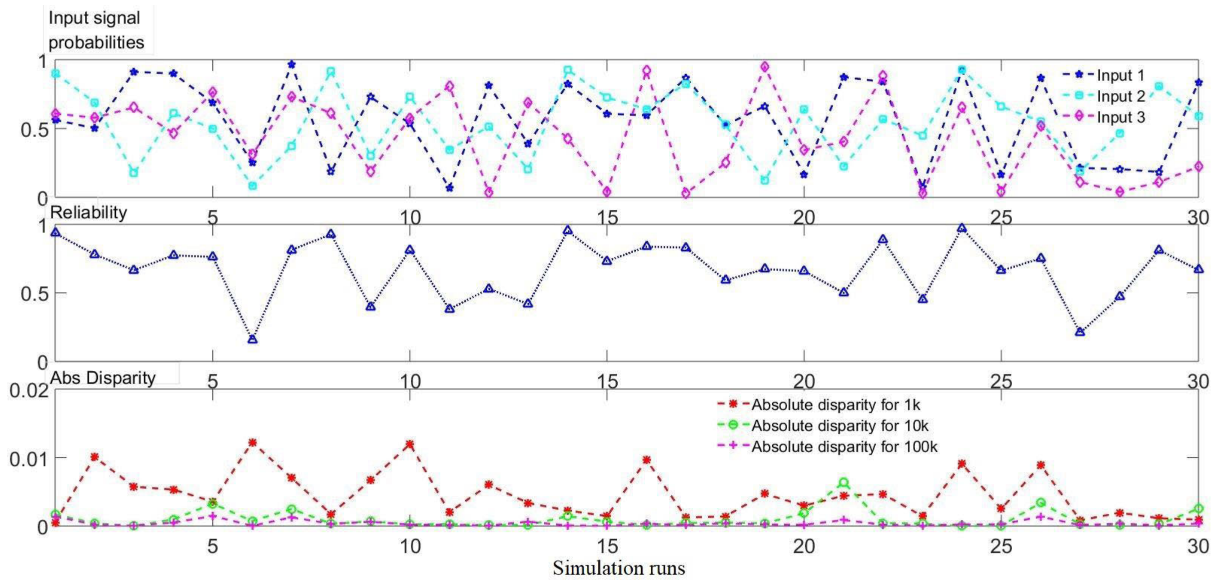


FIGURE 7. Results for the example in Figure 6(b). (a) Arbitrary inputs; (b) Accurate reliability; (c) absolute difference between stochastic and accurate results.

TABLE 3. Average run time (Avg.) for the recursive method [11], stochastic analysis and Monte Carlo (MC) simulation for the majority voter with randomly generated weights.

n	Avg. (s) for [11]	Avg. (s) for stochastic analysis			Avg. (s) for MC simulation		
		L = 1,000	L = 10,000	L = 100,000	N = 1,000	N = 10,000	N = 100,000
3	0.0002	0.0003	0.0017	0.0192	0.0003	0.0015	0.0138
4	0.0009	0.0003	0.0018	0.0222	0.0003	0.0022	0.0173
5	0.0020	0.0003	0.0026	0.0310	0.0003	0.0027	0.0219
6	0.0033	0.0004	0.0032	0.0361	0.0011	0.0028	0.0316
7	0.0022	0.0004	0.0037	0.0420	0.0011	0.0034	0.0322
8	0.0045	0.0005	0.0042	0.0449	0.0012	0.0040	0.0406
9	0.0053	0.0008	0.0053	0.0501	0.0012	0.0046	0.0407
10	0.0071	0.0009	0.0058	0.0529	0.0012	0.0048	0.0444
11	0.0075	0.0008	0.0063	0.0619	0.0013	0.0050	0.0487
12	0.0067	0.0008	0.0064	0.0681	0.0013	0.0057	0.0537
13	0.0068	0.0010	0.0064	0.0753	0.0014	0.0061	0.0559
14	0.0072	0.0011	0.0065	0.0798	0.0015	0.0068	0.0634
15	0.0078	0.0011	0.0078	0.0804	0.0016	0.0076	0.0681
16	0.0108	0.0012	0.0081	0. 0900	0.0017	0.0081	0.0757
18	0.0111	0.0014	0.0085	0.0959	0.0020	0.0088	0.0829
20	0.0117	0.0018	0.0079	0.1192	0.0020	0.0126	0.1258

The reliabilities of the binary majority voters (the scenarios for $n = 3$ and 4 in Table 3; here, L and N denote the sequence length for the stochastic analysis and the number of simulations for MC respectively.) found by stochastic analysis are further compared with the values obtained by using the UGF approach. The voter is also analyzed by MC simulation. Some parameters (e.g., the input signal probabilities and weights) are randomly generated and the benchmark is analyzed for different approaches. The sequence length (or the number of simulation runs) is varied in the stochastic analysis (or MC simulation). The stochastic analysis or the MC simulation is performed 30 times to determine the reliability. The obtained values are compared with the accurate value found by the UGF approach. Then the mean and variance of the differences in 30 simulations are determined. The obtained variances (in log) for the different scenarios are plotted in Figure 8. For $n = 3$, the variance decreases with an increase of the sequence length or simulation runs. If the sequence length and the number of simulation runs are 10,000, then the difference between the found and the accurate mean values are 1.5667×10^{-4} and 2.800×10^{-4}, respectively, for the stochastic analysis and MC simulation. While both values are small, the stochastic analysis is more accurate than MC simulation. This result also applies to the case when $n = 4$ and the other scenarios in Table 3.

The example in Figure 6(b) is also analyzed by using stochastic analysis and MC simulation; again, 30 simulation runs with randomly generated input signal probabilities are performed. The input signal probabilities are shown in Figure 9(a), and the absolute difference between the reliabilities obtained by the stochastic analysis and MC are plotted in Figure 9(b) (the threshold for the majority voter is set to 5). As shown in Figure 9(b), the difference between the two approaches decreases with an increase of sequence length for the stochastic analysis or simulation runs for MC. At the same sequence length and number of simulation runs, the stochastic analysis generates more accurate results than MC simulation.

C. THE WEIGHTED MULTI-STATE MAJORITY VOTER

If the state of a component is not limited to a binary value, then the weighted binary-state majority voter is generalized

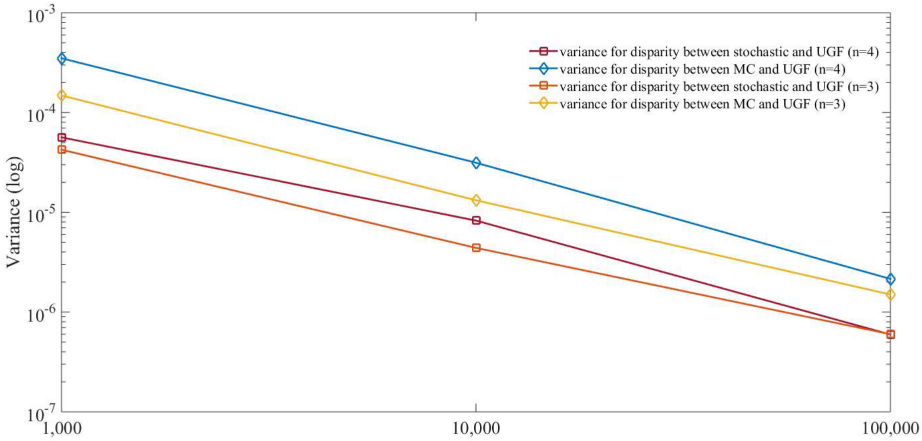


FIGURE 8. Variance (in log) of the difference between a stochastic analysis (or MC) and UGF.

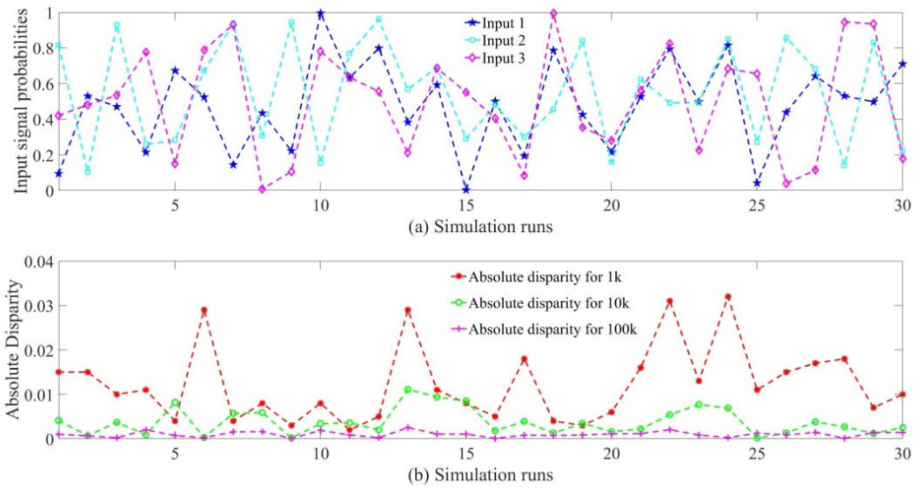


FIGURE 9. Simulation results of 30 runs for the example in Figure 6(b). (a) Arbitrary inputs; (c) absolute difference between stochastic and MC results.

to a weighted multi-state voter. For a weighted multi-state majority voter, every possible state of a specific component contributes to some extent to the system's output performance distribution.

For further application of stochastic computation in the analysis of a weighted multi-state system, the randomly permuted multiple-valued sequence can be coded by a combination of stochastic sequences with the number of sequences calculated as $[\log_2(j)]$ where j denotes the number of states for a specific component. An example is given in Figure 10 by using a stochastic sequence combination to represent the randomly permuted sequence of Figure 2 for a ternary signal; here, the sequence length is set to be 10 bits while "00", "01" and "10" indicate state 1, 2 and 3 respectively. In this case, a combination of two stochastic sequences is sufficient to encode the investigated ternary signal.

A stochastic model for a general weighted multi-state k-out-of-n majority voter is shown in Figure 11. Here, $w_{i,j}$ represents the weight distributions for state j of component i. S_i denotes the stochastic sequences determined by the

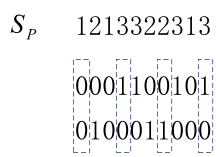


FIGURE 10. Stochastic decoding of a ternary signal, i.e., $p = [0.3, 0.3, 0.4]$.

input signal probabilities. For simplicity, we assume that each component has j states. The stochastic sequences for the input signal probabilities operate as the control sequences of the multiplexers. Depending on the state values, different weight contributions are selected; then, an n-input sum module is utilized to compute the total weight of the working components. Finally, the status of the system is determined by the output sequence of the threshold logic module.

Example 2(a): A weighted multi-valued k-out-of-3: G system is considered to show the efficiency of the proposed stochastic model. The system is operational if and only if

TABLE 4. Reliability distribution of component *i* in state *j*, i.e., $p_{i,j}$, and corresponding weight distribution $W_{i,j}$ [15].

	j=1		j=2		j=3	
	$p_{i,j}$	$w_{i,j}$	$p_{i,j}$	$w_{i,j}$	$p_{i,j}$	$w_{i,j}$
i=1	0.1	0	0.2	1	0.7	2
i=2	0.4	0	0.2	2	0.4	3
i=3	0.3	0	0.5	2	0.2	4

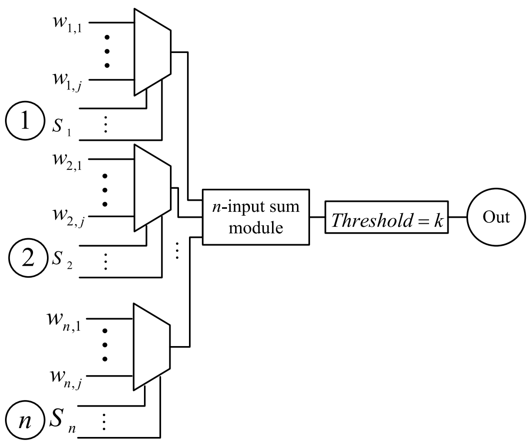


FIGURE 11. A stochastic model for a general weighted k-out-of-n majority voter.

the total weight of the functional components is at least k. For simplicity, it is assumed that each component has three states.

The reliability distribution and weight contributions of the components are given in Table 4. Consider the input signal probabilities and the weight distributions in Table 4, the analysis of Example 2(a) is performed by using the stochastic and UGF methods [12]. The simulation results for both approaches are presented in Table 5 for $k \in [2, 7]$. It can be seen that the stochastic analysis provides very accurate results using a reasonable sequence length compared with the UGF approach. As indicated by the simulation time in Table 5, the stochastic analysis can efficiently predict the output reliability. As the reliability by the UGF analysis is found using a formula, its run time is very short (by ignoring the time required for the UGF deriving process), 0.0001249 second in the case considered here.

In fact, the proposed stochastic model of Figure 11 is a generalization of the model of Figure 5(b). The multiplexer in Figure 11 can be simplified as the multiplication module in Figure 5(b) if the state of a component is binary.

D. THE CONSECUTIVE-WEIGHTED MULTI-STATE MAJORITY VOTER

If the weight of the system is equal to the maximum sum of the weights of the consecutive working components, then the weighted multi-state majority voter is generalized to a consecutive-weighted multi-state voter. For a

TABLE 5. Comparison of the reliabilities of weighted multi-valued *k*-out-of-3 majority voters obtained by stochastic and UGF analysis, *k* ∈ [2, 7].

k		Sequence length L (bits)			UGF analysis [12]
		1,000	10,000	100,000	
2	R	0.9590	0.9627	0.9634	0.964
	Avg. (s)	0.0068	0.0095	0.0393	
3	R	0.8360	0.8504	0.8543	0.8540
	Avg. (s)	0.0080	0.0120	0.0483	
4	R	0.7970	0.7910	0.7894	0.7900
	Avg. (s)	0.0083	0.0108	0.0479	
5	R	0.5700	0.5680	0.5666	0.566
	Avg. (s)	0.0067	0.0095	0.0557	
6	R	0.4230	0.4236	0.4245	0.4260
	Avg. (s)	0.0078	0.0102	0.0494	
7	R	0.2670	0.2536	0.2556	0.256
	Avg. (s)	0.0085	0.0121	0.0640	

consecutive-weighted multi-state majority voter, every possible state of a specific component contributes to some extent to the system's output performance distribution.

The majority voter shown in Figure 11 is generalized to a consecutive-weighted multi-state voter. Depending on the state values, different weight contributions are selected; then, an n-input sum module is utilized to compute the total weight of the consecutive-working components. Finally, the status of the system is determined by the output sequence of the threshold logic module.

Example 2(b): The majority voter in Example 2 is generalized to a consecutive-weighted multi-state voter. A consecutive-weighted multi-state k-out-of-3: G system is considered to show the efficiency of the proposed stochastic model. The system is operational if and only if the total weight of the consecutive-working components is at least k. For simplicity, it is assumed that each component has three states.

The UGF of component i is a polynomial function that relates the probability of each state to the performance of the component. The UGF defines the OPD for the investigated system. For the multi-state consecutive-weighted-k-out-of-n system, the UGF of component i is given by $U_{\mathrm{i}}(z) = \sum p_{i,j} z^{wi,j}$; here, $p_{i,j}$ denotes the probability of component i being in state j, and $w_{i,j}$ represents the weight distributions for state j of component i. The operator $\Omega = \Omega(U_1(z), U_2(z), \ldots, U_n(z))$ in [12] can be utilized for describing the UGF of a system, where Ω is determined by the corresponding structure of the components.

The UGFs for the input components are obtained as:

$$U_1(z) = p_{1,1}z^0 + p_{1,2}z^1 + p_{1,3}z^2$$
$$U_2(z) = p_{2,1}z^0 + p_{2,2}z^2 + p_{2,3}z^3$$
$$U_3(z) = p_{3,1}z^0 + p_{3,2}z^2 + p_{3,3}z^4$$

TABLE 6. Comparison of the reliabilities of weighted multi-valued *k*-out-of-3 majority voters obtained by stochastic and UGF analysis, $k \in [2, 7]$.

k		Sequence length L (bits)			UGF analysis [12]
		1,000	10,000	100,000	
2	R	0.9650	0.9643	0.9641	0.964
	Avg. (s)	0.0027	0.0164	0.0439	
3	R	0.6820	0.6772	0.6744	0.674
	Avg. (s)	0.0104	0.0126	0.0548	
4	R	0.6330	0.6511	0.6507	0.650
	Avg. (s)	0.0097	0.0139	0.0769	
5	R	0.5070	0.4929	0.4943	0.494
	Avg. (s)	0.0093	0.0127	0.0588	
6	R	0.3560	0.3714	0.3697	0.370
	Avg. (s)	0.0110	0.0141	0.0503	
10	R	0.2640	0.2562	0.2561	0.256
	Avg. (s)	0.0098	0.0122	0.0487	

Then, the exact expression obtained by the UGF analysis is:

$$\begin{aligned} U_s(z) = {} & p_{1,1}p_{2,1}p_{3,1}z^0 + p_{1,2}p_{2,1}p_{3,1}z^1 \\ & + (p_{1,3}p_{2,1}p_{3,1} + p_{1,1}p_{2,2}p_{3,1} + p_{1,1}p_{2,1}p_{3,2})z^2 \\ & + (p_{1,2}p_{2,2}p_{3,1} + p_{1,1}p_{2,3}p_{3,1})z^3 \\ & + (p_{1,2}p_{2,3}p_{3,1} + p_{1,3}p_{2,2}p_{3,1} + p_{1,1}p_{2,2}p_{3,2} \\ & + p_{1,1}p_{2,1}p_{3,3})z^4 + (p_{1,2}p_{2,2}p_{3,2} + p_{1,3}p_{2,3}p_{3,1})z^5 \\ & + (p_{1,3}p_{2,2}p_{3,2} + p_{1,2}p_{2,3}p_{3,2} + p_{1,1}p_{2,2}p_{3,3})z^6 \\ & + (p_{1,2}p_{2,2}p_{3,3} + p_{1,3}p_{2,3}p_{3,2} + p_{1,1}p_{2,3}p_{3,3})z^7 \\ & + (p_{1,2}p_{2,3}p_{3,3} + p_{1,3}p_{2,2}p_{3,3})z^8 + z^9 \end{aligned} \tag{2}$$

Consider the input signal probabilities and the weight distributions in Table 4, the analysis of Example 2(b) is performed by using the stochastic and UGF methods. The simulation results for both approaches are presented in Table 6 for $k \in [2, 7]$. It can be seen that the stochastic analysis provides very accurate results using a reasonable sequence length compared with the UGF approach. As indicated by the simulation time in Table 6, the stochastic analysis can efficiently predict the output reliability.

L denotes the sequence length used in the stochastic analysis. The average run time (Avg. (s)) for the stochastic analysis is also provided. As evidenced by the simulation results for Examples 1 and 2 (in Tables 2, 3, 5, 6), the stochastic approach can accurately and efficiently predict the overall reliability. Moreover, the proposed model can also be applied to a majority voter with mixed states (i.e., different component has different number of states). The simulation time for the stochastic analysis only slightly varies as a function of the threshold value; so, the threshold k has no significant effect on the average run time.

TABLE 7. Reliability of binary component *i*, i.e., p_i, where $i \in \{A, B, C, \ldots, J\}$, and the corresponding weight w_i.

<table>
<tr><td></td><td colspan="7">MV_1</td></tr>
<tr><td>Component</td><td>A</td><td colspan="2">B</td><td colspan="2">C</td><td colspan="2">D</td></tr>
<tr><td>Reliability</td><td>p_A =0.8</td><td colspan="2">p_B =0.6</td><td colspan="2">p_C =0.9</td><td colspan="2">p_D =0.78</td></tr>
<tr><td></td><td colspan="7">MV_2</td></tr>
<tr><td>Component</td><td>E</td><td colspan="2">F</td><td colspan="2">G</td><td colspan="2">H</td></tr>
<tr><td>Reliability</td><td>p_E =0.8</td><td colspan="2">p_F =0.5</td><td colspan="2">p_G =0.6</td><td colspan="2">p_H =0.86</td></tr>
<tr><td>Weights</td><td>$w_E = 2$</td><td colspan="2">$w_F = 3$</td><td colspan="2">$w_G = 1$</td><td colspan="2">$w_H = 4$</td></tr>
<tr><td>Component</td><td>B</td><td colspan="2">D</td><td colspan="2">I</td><td colspan="2">J</td></tr>
<tr><td>Reliability</td><td>p_B =0.6</td><td colspan="2">p_D =0.78</td><td colspan="2">p_I =0.84</td><td colspan="2">p_J =0.55</td></tr>
<tr><td>Weights</td><td>$w_B = 4$</td><td colspan="2">$w_D = 7$</td><td colspan="2">$w_I = 5$</td><td colspan="2">$w_J = 2$</td></tr>
<tr><td colspan="2" rowspan="2"></td><td colspan="2">j=1</td><td colspan="2">j=2</td><td colspan="2">j=3</td></tr>
<tr><td>$p_{i,j}$</td><td>$w_{i,j}$</td><td>$p_{i,j}$</td><td>$w_{i,j}$</td><td>$p_{i,j}$</td><td>$w_{i,j}$</td></tr>
<tr><td rowspan="4">MV_3</td><td>K</td><td>0.1</td><td>0</td><td>0.2</td><td>2</td><td>0.7</td><td>3</td></tr>
<tr><td>L</td><td>0.4</td><td>0</td><td>0.2</td><td>1</td><td>0.4</td><td>3</td></tr>
<tr><td>M</td><td>0.3</td><td>0</td><td>0.4</td><td>3</td><td>0.3</td><td>5</td></tr>
<tr><td>N</td><td>0.6</td><td>0</td><td>0.2</td><td>2</td><td>0.2</td><td>3</td></tr>
<tr><td rowspan="2">MV_4</td><td>R</td><td>0.33</td><td>0</td><td>0.4</td><td>3</td><td>0.27</td><td>4</td></tr>
<tr><td>S</td><td>0.28</td><td>0</td><td>0.42</td><td>2</td><td>0.3</td><td>5</td></tr>
<tr><td colspan="2">O in MV_3</td><td rowspan="2">0.25</td><td>0</td><td rowspan="2">0.25</td><td>1</td><td rowspan="2">0.5</td><td>3</td></tr>
<tr><td colspan="2">O in MV_4</td><td>0</td><td>2</td><td>3</td></tr>
<tr><td colspan="2">P in MV_3</td><td rowspan="2">0.35</td><td>0</td><td rowspan="2">0.45</td><td>3</td><td rowspan="2">0.2</td><td>5</td></tr>
<tr><td colspan="2">P in MV_4</td><td>0</td><td>2</td><td>4</td></tr>
<tr><td colspan="2">Q in MV_3</td><td rowspan="2">0.6</td><td>0</td><td rowspan="2">0.1</td><td>2</td><td rowspan="2">0.3</td><td>3</td></tr>
<tr><td colspan="2">Q in MV_4</td><td>0</td><td>1</td><td>2</td></tr>
</table>

IV. CASE STUDIES

In this section, the analysis of a system consisting of several majority voters is presented to show the capability of the proposed stochastic models. All simulations are run on a computer with a 2.30 GHz i5-6200U microprocessor and an 8 GB memory.

Example 3: The system in this example is composed of 4 majority voters (i.e., MV_1, MV_2, MV_3, MV_4) and 17 components (A∼ H are components with binary states, while K∼S are components with ternary states). This system is shown in Figure 12(a).

In Figure 12(a), MV_1, MV_2, MV_3 and MV_4 are respectively a 3-out-of-4 majority voter, a weighted 20-out-of-8 binary majority voter, a weighted-10-out-of-7 multi-state majority voter and a consecutive-weighted-6-out-of-5 multi-state majority voter. B and D are common components for MV_1 and MV_2, while O, P and Q are common components for MV_3 and MV_4. The components in MV_4 are adjacent in the order of "O-P-Q-R-S". The common components are assumed to have different weights for different majority voters. The reliability of the main task (or event) is determined by the output of the 4 majority voters and the system topology.

The reliability and weight distributions for the components are reported in Table 7; the reliability is assumed to be a fixed value for each component. The reliability of component m, $m \in \{K, \ldots, S\}$, in state j and the corresponding weight are represented as $p_{m,j}$ and $w_{m,j}$ respectively.

A stochastic model is then constructed in Figure 12(b) by applying the proposed stochastic models of Figures 3(c), 5(b) and 11. By propagating non-Bernoulli and randomly permuted sequences through this model, a stochastic analysis

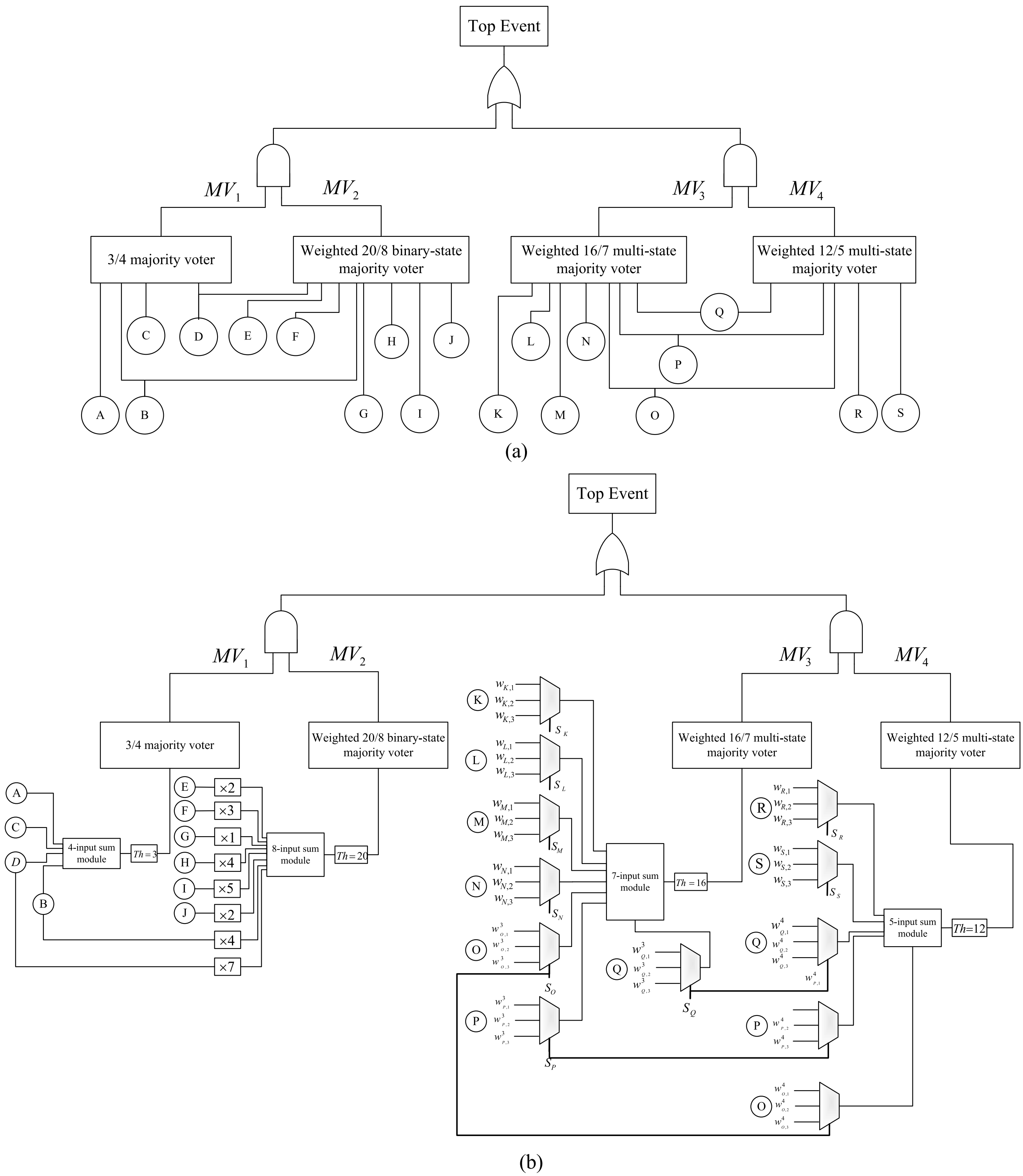


FIGURE 12. (a) An illustrative system. (b) Stochastic model for the system in (a). Here, S_i to the multiplexer is a combination of two-bit stochastic sequences.

is performed to find the reliability of the main event. The simulation results of Example 3, as well as the average run time for the stochastic approach, are reported in Table 8 for different sequence lengths.

As per the simulation results, the proposed stochastic approach can deal with reconvergent fanouts as caused by common components; furthermore, components with different failure distributions are efficiently taken into account by the encoding property of stochastic sequences.

In practice, if external events occur (such as when a neutron hits a chip, causing a single event upset), the overall reliability of the system is affected. To ensure the correct operation

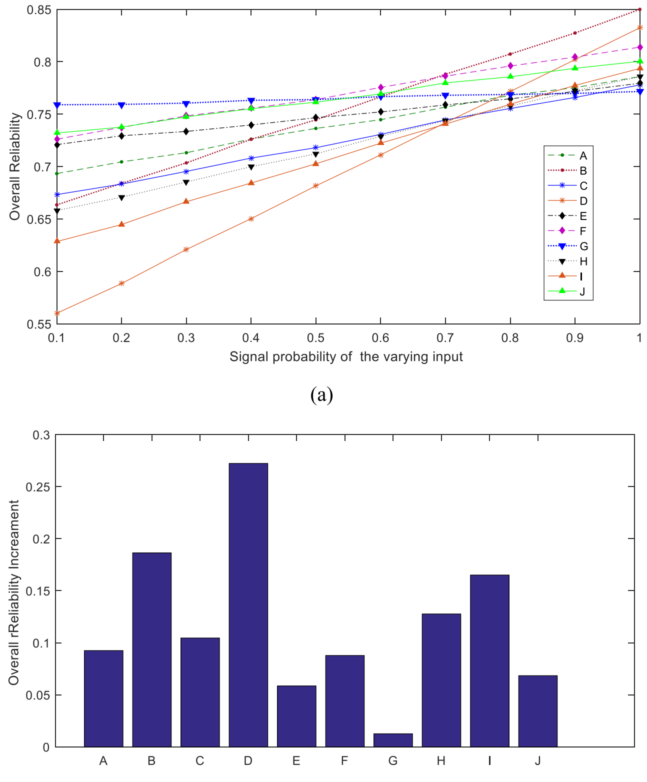
TABLE 8. **Reliabilities of majority voters in Figure 11(a) and the top event obtained by stochastic analysis, as well as the average run time.**

	Sequence length L (bits)		
	$L = 1{,}000$	$L = 10{,}000$	$L = 100{,}000$
Reliability for MV_1	0.7810	0.7787	0.7761
Reliability for MV_2	0.5950	0.6009	0.5989
Reliability for MV_3	0.7900	0.8041	0.8068
Reliability for MV_4	0.5660	0.5677	0.5671
Reliability for Top event	0.7550	0.7720	0.7663
Avg. (s)	0.0252	0.0489	0.4160

of a system, it is vital to identify the most important (critical) component because available resources, usually limited, can be directed for remedy. For the investigated system in Figure 12(a), we assume that the signal probability of a specific input for MV_1 and MV_2 varies from 0.1 to 1, while the parameters for the other components of MV_1 and MV_2 remain the same as in Table 7.

As shown in Figure 13(a), the component G is less important with respect to increasing the overall reliability than other components. However, if the reliability of component D changes, the reliability of the overall system will significantly change. Thus, to ensure a correct operation, the reliability of component D is critical. Furthermore, the relative relationship between the increasing overall system reliability and the reliability of component i in a range from 0.1 to 1 is illustrated in Figure 13(b); the sequence length equals to 10,000. When external factors affecting the reliability of a component occur, D is the most critical component, i.e., if the reliability of D decreases, the overall system reliability decreases most rapidly. The order of importance of the binary state components is given by D>B>I>H>C>A>F>J>E>G. Hence, the stochastic analysis is capable of predicting the importance order of components, providing further guidance in ensuring a correctly operating system.

For the stochastic architecture, a complex operation can be performed by employing a simple logic. The proposed stochastic scheme is independent of the input sequence; a unary sequence [39] can also be applied with the proposed models. However, it should be realized that an increase of sequence length will incur in a high latency and the generation of random sequence is costly. As indicated in [40], pseudorandom circuits account for approximately 90% of the area of a stochastic circuit design. A Deep Neural Network is analyzed through stochastic analysis in [41]. Though the existing stochastic algorithms are likely to incur long latencies, the stochastic computing is still desirable for low-power area-efficient hardware implementations [41]. Thus, in the future, it is of interest to investigate the application of new and different types of sequence, such as the Sobol sequence [42], a deterministic sequence [39] and time-encoded values [43] to overcome the limitations of current techniques. Furthermore, this technique is also capable of dealing with analysis of complex networks [44].



(b)

FIGURE 13. **(a) The relationship of overall system reliability; (b) Reliability increment given variation of the signal probability for certain component *i*.**

V. CONCLUSION

In this paper, a stochastic architecture is proposed for a binary-state majority system; this model can efficiently and accurately evaluate the corresponding system reliability. Furthermore, for a consecutive-weighted system (i.e., each component contributes differently to the total system), a stochastic analysis is pursued to predict the system reliability (both binary and multi-state properties can be investigated accordingly). The input signal probabilities are encoded as non-Bernoulli sequences of random permutations of fixed numbers of 1s and 0s for the Boolean case, or randomly permuted sequences of random permutations of fixed numbers of multi-values for the multi-state scenario. These stochastic sequences are propagated through the proposed stochastic model; the system reliability can then be obtained efficiently and accurately by analyzing the stochastic sequence at the output, as validated by the results for several case studies.

The evaluation accuracy can be improved by increasing the sequence length L for the stochastic approach, as indicated by the reduced absolute result difference at a longer sequence length; so, a trade-off between accuracy and efficiency is achieved by the selection of the sequence length. As shown by the simulation time of the investigated benchmarks, the stochastic approach is more efficient than MC simulation. It has been found that the average run time required for stochastic analysis is mostly dependent on the employed sequence length; furthermore, it has been shown that the threshold has little impact on the average run time.

The stochastic approach is capable of dealing with any failure distribution of the components as well as a system consisting of mixed states, for example by the simulation results of Example 1 with arbitrary input signal probabilities. This is made possible because for any failure distribution and specific mission time points, the failure probability can be readily obtained for any investigated mission time. This probability is then directly encoded into stochastic sequences. This is an advantageous feature of a stochastic analysis; nevertheless, the stochastic approach consumes more memory than an analytical approach. As shown in [29], with an increase of sequence length, a larger memory is required for the stochastic approach than for MC simulation. However, due to the faster convergence, the efficiency and accuracy of the stochastic approach are still highly desirable. The use of longer stochastic sequences considerably improves the accuracy of the evaluation, i.e., it is more efficient than the MC method by several orders of magnitude. Finally, its efficiency can be further improved through parallelization in the stochastic computation. Additionally, the importance of each component in the voting system can be efficiently assessed by the proposed models to provide further guidance for protective/corrective processes (such as maintenance) and the continued correct operation in critical applications.

In the future work, we will discuss the systems with and dynamically repairable components. The effect of common cause failures (CCF, caused by flood, hurricane, etc.) for system can be considered.

REFERENCES

[1] S. P. Jain and K. Gopal, "Recursive algorithm for reliability evaluation of k-out-of-n:G system," *IEEE Trans. Rel.*, vol. R-34, no. 2, pp. 144–150, Jun. 1985.
[2] S. P. York, "Engine placement for manned descent at mars considering single engine failures," M.S. thesis, Dept. Aeronaut. Astronaut., Massachusetts Inst. Technol., Cambridge, MA, USA, 2006.
[3] C. H. Kuo, C. M. Kuo, A. Leber, and C. Boller, "Vector thrust multi-rotor copter and its application for building inspection," in *Proc. Int. Micro Air Vehicle Conf. Flight Competition*, Toulouse, France, 2013, pp. 1–10.
[4] Z. Tian, G. Levitin, and M. J. Zuo, "A joint reliability–redundancy optimization approach for multi-state series–parallel systems," *Rel. Eng. Syst. Saf.*, vol. 94, no. 10, pp. 1568–1576, Oct. 2009.
[5] B. Cai, Y. Zhao, H. Liu, and M. Xie, "A data-driven fault diagnosis methodology in three-phase inverters for PMSM drive systems," *IEEE Trans. Power Electron.*, vol. 32, no. 7, pp. 5590–5600, Jul. 2017.
[6] T. Kim, S. J. Wright, D. Bienstock, and S. Harnett, "Analyzing vulnerability of power systems with continuous optimization formulations," *IEEE Trans. Netw. Sci. Eng.*, vol. 3, no. 3, pp. 132–146, Jul. 2016.
[7] A. Myers, *Complex System Reliability*. London, U.K.: Springer-Verlag, 2010.
[8] W. Kuo and M. J. Zuo, *Optimal Reliability Modeling: Principles and Applications*. Hoboken, NJ, USA: Wiley, 2003.
[9] A. Dembińska, "On reliability analysis of k-Out-of-n systems consisting of heterogeneous components with discrete lifetimes," *IEEE Trans. Rel.*, vol. 67, no. 3, pp. 1071–1083, Sep. 2018.
[10] H. Dui, "On the use of the importance measure for multi-state repairable k-out-of-n: G systems," *Commun. Statist.-Theory Methods*, vol. 43, no. 13, pp. 2766–2781, 2014.
[11] J. S. Wu and R. J. Chen, "An algorithm for computing the reliability of weighted-k-out-of-n systems," *IEEE Trans. Rel.*, vol. 43, no. 2, pp. 327–328, Jun. 1994.
[12] I. Uashakov, I. Ushakov, and I. B. Ushakov, "Optimal standby problem and a universal generating function," *Sov J. Comput. Syc Sci.*, vol. 25, no. 4, pp. 67–73, 1987.
[13] B. Jafary and L. Fiondella, "A universal generating function-based multi-state system performance model subject to correlated failures," *Rel. Eng. Syst. Saf.*, vol. 152, pp. 16–27, Aug. 2016.
[14] G. Levitin, A. Lisnianski, H. Ben-Haim, and D. Elmakis, "Redundancy optimization for series-parallel multi-state systems," *IEEE Trans. Rel.*, vol. 47, no. 2, pp. 165–172, Jun. 1998.
[15] W. Li and M. J. Zuo, "Reliability evaluation of multi-state weighted k-out-of-n systems," *Rel. Eng. Syst. Saf.*, vol. 93, pp. 160–167, Jan. 2008.
[16] S. Eryilmaz, "Capacity loss and residual capacity in weighted k-out-of-n:G systems," *Rel. Eng. Syst. Saf.*, vol. 136, pp. 140–144, Apr. 2015.
[17] G. Wang, R. Peng, and L. Xing, "Reliability evaluation of unrepairable k-out-of-n: G systems with phased-mission requirements based on record values," *Rel. Eng. Syst. Saf.*, vol. 178, pp. 191–197, Oct. 2018.
[18] A. F. Myers, "Achievable limits on the reliability of k-out-of-n: G systems subject to imperfect fault coverage," *IEEE Trans. Rel.*, vol. 57, no. 2, pp. 349–354, Jun. 2008.
[19] L. Xing, S. V. Amari, and C. Wang, "Reliability of k-out-of-n systems with phased-mission requirements and imperfect fault coverage," *Rel. Eng. Syst. Saf.*, vol. 103, pp. 45–50, Jul. 2012.
[20] Y. Ding, E. Zio, Y. Li, L. Cheng, and Q. Wu, "Definition of multi-state weighted k-out-of-n: F systems," *Int. J. Performability Eng.*, vol. 8, no. 2, pp. 217–219, 2012.
[21] S. Faghih-Roohi, M. Xie, K. M. Ng, and R. C. M. Yam, "Dynamic availability assessment and optimal component design of multi-state weighted k-out-of-n systems," *Rel. Eng. Syst. Saf.*, vol. 123, pp. 57–62, Mar. 2014.
[22] Y. Wang, L. Li, S. Huang, and Q. Chang, "Reliability and covariance estimation of weighted k-out-of-n multi-state systems," *Eur. J. Oper. Res.*, vol. 221, no. 1, pp. 138–147, Aug. 2012.
[23] X. Zhuang, T. Yu, and L. Shen, "On capacity evaluation for multi-state weighted k-out-of-n system," *Commun. Statist.-Simul. Comput.*, vol. 48, no. 7, pp. 2083–2098, Feb. 2018.
[24] Y. Mo, L. Xing, S. V. Amari, and J. Bechta Dugan, "Efficient analysis of multi-state k-out-of-n systems," *Rel. Eng. Syst. Saf.*, vol. 133, pp. 95–105, Jan. 2015.
[25] H. Abdollahzadeh and K. Atashgar, "Optimal design and imperfect opportunistic maintenance of multi-state weighted k-out-of-n systems considering uncertainty in supplier selection," *Comput. Ind. Eng.*, vol. 105, pp. 411–424, 2017.
[26] X. Song, Z. Zhai, Y. Guo, P. Zhu, and J. Han, "Approximate analysis of multi-state weighted k-Out-of-n systems applied to transmission lines," *Energies*, vol. 10, no. 11, p. 1740, Oct. 2017.
[27] 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. Comput.*, vol. 63, no. 6, pp. 1336–1350, Jun. 2014.
[28] P. Zhu, J. Han, L. Liu, and F. Lombardi, "A stochastic approach for the analysis of dynamic fault trees with spare gates under probabilistic common cause failures," *IEEE Trans. Rel.*, vol. 64, no. 3, pp. 878–892, Sep. 2015.
[29] P. Zhu and J. Han, "Stochastic multiple-valued gene networks," *IEEE Trans. Biomed. Circuits Syst.*, vol. 8, no. 1, pp. 42–53, Feb. 2014.
[30] N. F. Vaidya and D. K. Pradhan, "Fault-tolerant design strategies for high reliability and safety," *IEEE Trans. Comput.*, vol. 42, no. 10, pp. 1195–1206, Oct. 1993.
[31] T. Ban and L. Naviner, "Progressive module redundancy for fault tolerant designs in nanoelectronics," *Microelectron. Rel.*, vol. 51, no. 9, pp. 1489–1492, 2011.
[32] S. L. Hight and D. P. Petersen, "Dissent in a majority voting system," *IEEE Trans. Comput.*, vol. C-22, no. 2, pp. 168–171, Feb. 1973.
[33] D. P. Siewiorek, "Reliability modeling of compensating module failures in majority voted redundancy," *IEEE Trans. Comput.*, vol. C-24, no. 5, pp. 525–533, May 1975.
[34] J. Han, H. Chen, E. Boykin, and J. Fortes, "Reliability evaluation of logic circuits using probabilistic gate models," *Microelectron. Rel.*, vol. 51, no. 2, pp. 468–476, Feb. 2011.
[35] J. Liang, J. Han, and F. Lombardi, "New metrics for the reliability of approximate and probabilistic adders," *IEEE Trans. Comput.*, vol. 62, no. 9, pp. 1760–1771, Sep. 2013.
[36] B. R. Gaines, "Stochastic computing systems," in *Advances in Information Systems Science*, vol. 2. Boston, MA, USA: Springer, 1969, pp. 172–137.

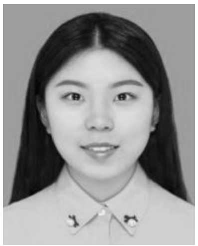
[37] Y. Chen and Q. Yang, "Reliability of two-stage weighted-k-out-of-n systems with components in common," *IEEE Trans. Rel.*, vol. 53, no. 3, pp. 431–440, Sep. 2005.
[38] *Fault Tree Handbook With Aerospace Applications*, NASA, Washington, DC, USA, 2002.
[39] D. Jenson and M. Riedel, "A deterministic approach to stochastic computing," in *Proc. Int. Conf. Comput.-Aided Design*, 2016, p. 102.
[40] W. Qian, X. Li, M. D. Riedel, K. Bazargan, and D. J. Lilja, "An architecture for fault-tolerant computation with stochastic logic," *IEEE Trans. Comput.*, vol. 60, no. 1, pp. 93–105, Jan. 2011.
[41] A. Ardakani, F. Leduc-Primeau, N. Onizawa, T. Hanyu, and W. J. Gross, "VLSI implementation of deep neural network using integral stochastic computing," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 25, no. 10, pp. 2688–2699, Oct. 2017.
[42] S. Liu and J. Han, "Energy efficient stochastic computing with Sobol sequences," in *Proc. Design, Autom. Test Eur. Conf. Exhib. (DATE)*, Mar. 2017.
[43] M. H. Najafi, S. Jamali-Zavareh, D. J. Lilja, M. D. Riedel, K. Bazargan, and R. Harjani, "Time-encoded values for highly efficient stochastic circuits," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 25, no. 5, pp. 1644–1657, May 2017.
[44] P. Zhu, X. Wang, S. Li, Y. Guo, and Z. Wang, "Investigation of epidemic spreading process on multiplex networks by incorporating fatal properties," *Appl. Math. Comput.*, vol. 359, pp. 512–524, Oct. 2019.

NING WANG received the B.S. degree in management science and engineering from the Zhongnan University of Economics and Law (ZUEL), Wuhan, China, in 2004, and the M.S. degree in management science and engineering, and the Ph.D. degree in management science and engineering from Northwestern Polytechnical University (NPU), Xi'an, China, in 2007 and 2012, respectively.

In 2011, with the support of China Scholarship Council, he joined The University of Texas at San Antonio (UTSA), San Antonio, TX, USA, as a Co-Supervised Ph.D. Student. He is an Associate Professor with the School of Automobile, Chang'an University. He has published more than 20 academic articles and articles in journals and conferences in the past five years. His research interests include reliability modeling, importance measures, maintenance management, and decision making support.

WEI FENG received the B.S. degree in software engineering from Yunnan University, Kunming, China, in 2019. He is currently pursuing the M.S. degree with Northwestern Polytechnical University (NPU), Xi'an, China.

His current research interests include detection/diagnosis/prognosis and reliability modeling.



HAILUN ZHANG received the B.S. degree in software engineering from the Xi'an University of Posts and Telecommunications, Xi'an, China, in 2019. She is currently pursuing the M.S. degree with Northwestern Polytechnical University (NPU), Xi'an.

Her current research interests include detection/diagnosis/prognosis, reliability modeling, complex networks, and decision making.

SHUMIN LI received the B.S. degree in industrial engineering, the M.S. degree in management science and engineering, and the Ph.D. degree in management science and engineering from Northwestern Polytechnical University (NPU), Xi'an, China, in 2007, 2010, and 2014, respectively.

She is currently a Lecturer with the School of Management Engineering, Zhengzhou University, Zhengzhou, China. Her current research interests include reliability modeling, importance measures, maintenance management, and decision diagrams.