Available online at www.sciencedirect.com

SCIENCE DIRECT®

Reliability Engineering and System Safety 87 (2005) 253–264

RELIABILITY ENGINEERING & SYSTEM SAFETY

www.elsevier.com/locate/ress

A Monte-Carlo simulation approach for approximating multi-state two-terminal reliability

Jose E. Ramirez-Marquez, David W. Coit*

Department of Industrial and Systems Engineering, Rutgers University, 96 Frelinghuysen Road, Piscataway, NJ 08854-8018, USA

Received 28 February 2004; accepted 11 May 2004

Abstract

This paper describes a Monte-Carlo (MC) simulation methodology for estimating the reliability of a multi-state network. The problem under consideration involves multi-state two-terminal reliability (M2TR) computation. Previous approaches have relied on enumeration or on the computation of multi-state minimal cut vectors (MMCV) and the application of inclusion/exclusion formulae. This paper discusses issues related to the reliability calculation process based on MMCV. For large systems with even a relatively small number of component states, reliability computation can become prohibitive or inaccurate using current methods. The major focus of this paper is to present and compare a new MC simulation approach that obtains accurate approximations to the actual M2TR. The methodology uses MC to generate system state vectors. Once a vector is obtained, it is compared to the set of MMCV to determine whether the capacity of the vector satisfies the required demand. Examples are used to illustrate and validate the methodology. The estimates of the simulation approach are compared to exact and approximation procedures from solution quality and computational effort perspectives. Results obtained from the simulation approach show that for relatively large networks, the maximum absolute relative error between the simulation and the actual M2TR is less than 0.9%, yet when considering approximation formulae, this error can be as large as 18.97%. Finally, the paper discusses that the MC approach consistently yields accurate results while the accuracy of the bounding methodologies can be dependant on components that have considerable impact on the system design.

Keywords: Monte-Carlo simulation; Multi-state minimal cut vector; Multi-state reliability computation

1. Introduction

The computation of two-terminal reliability (2TR) is a classical network reliability problem when considering binary and system states, i.e. both network and components can only be in fully working or fully failed states. Numerous approaches and methodologies have been proposed to solve this difficult problem [1–5], but these approaches operate under the assumption that a system and its components can only be in working or failed states. However, researchers have indicated that in some cases, binary state theory fails to characterize the actual system reliability behavior [6–10], which is multi-state. Misrepresenting system behavior as a binary event can be problematic since systems can have a range of intermediate states that are not accounted for in the reliability computation. For some systems, such erroneous appraisal of system reliability could translate into: (1) incorrect system modeling, (2) incorrect system reliability computation, and/or (3) incorrect conjectures regarding reliability-dependant measures.

Multi-state reliability has been proposed as a complementary theory to cope with the problem of analyzing systems where traditional binary reliability theory and models become insufficient [6–9]. For systems such as water distribution systems, telecommunication systems, oil and gas supply systems, and power generation and transmission systems, an analysis of 2TR from a multi-state view may be the preferred approach [10,27]. For these types of networked systems, it may be insufficient to consider a binary state behavior of the components. Generally, the components of these systems follow a degradation pattern that reduces the ability of the system to provide some required service.

* Corresponding author. Fax: +1-732-445-5467.
E-mail address: coit@rci.rutgers.edu (D.W. Coit).

0951-8320/$ - see front matter
doi:10.1016/j.ress.2004.05.002

Recently the multi-state extension of 2TR has received considerable attention [11–17]. For this extension, multi-state two-terminal reliability at demand level d (M2TR$_d$) is defined as the probability that a demand of d units can be supplied from source to sink nodes through multi-state arcs. Most of these approaches solve the problem by providing multi-state minimal cut or path vectors (MMCV, MMPV), which are the multi-state equivalent of minimal path or cut sets in 2TR.

Previous research [13–16] assumes or implies that once MMCV or MMPV are obtained, numerical reliability computation is straightforward or even trivial. The preferred approach used for obtaining M2TR$_d$ is the classical inclusion/exclusion formula. For small networks, this may be a simple approach. However, as the size of the network or the number of component states increase, using such formula may be computationally inefficient, and thus, not trivial at all. Therefore, alternative reliability computation approaches are needed.

Actual systems, where M2TR$_d$ is the most appropriate reliability metric, are complex. Reliability computation may be too costly to obtain through traditional techniques. Simulation has been effectively used for analyzing 2TR for relatively large systems [18]. However, simulation has been underutilized for approximating M2TR$_d$. In this paper, a Monte-Carlo (MC) simulation methodology for estimating numerical M2TR is proposed based on identified MMCV [13–17]. The methodology consists of using MC simulation to generate system state vectors. Once a vector is obtained it is compared to the set of MMCV to decide if the capacity of such a vector satisfies the required demand.

The remainder of the paper is organized as follows: the rest of this section defines the M2TR problem. Section 2 illustrates different methodologies that have been developed for analyzing M2TR$_d$ and techniques used for approximating reliability. Section 3 presents the new methodology, and in Section 4, examples are used to illustrate and validate the methodology. These examples are used to compare the proposed methodology against other approximation methodologies. Finally, Section 5 presents conclusions.

1.1. Problem description

Let $G=(N,A)$ represent a stochastic capacitated network with a specified required demand d from source node s to sink node t. N represents the set of nodes, $A=\{a_i|1\leq i\leq|A|\}$ represents the set of arcs. The current state (capacity) of arc a_i, is represented by x_i. The current state (capacity) of arc a_i, represented by $x_i\in\mathbf{b}_i$, takes values $b_{i1}=0, b_{i2},\ldots,b_{il}=M_i$, where $b_{ij}\in\Re^+$ and $\mathbf{b}_i$ represents arc a_i state space vector. The vector $\mathbf{p}_i$ represents the probability associated with each of the values taken by x_i. The system state vector $\mathbf{x}=(x_1,x_2,\ldots,x_{|A|})$ denotes the current state of all the arcs of the network. The function φ: $\Re^{|A|}\rightarrow\Re^+$ maps the system state vector into a system state. That is, $\varphi(\mathbf{x})$ is the available capacity from source to sink under system state vector $\mathbf{x}$. M2TR$_d$ is understood as the probability that a demand of d units can be supplied from source to sink through the multi-state arcs, i.e. M2TR$_d=P(\varphi(\mathbf{x})\geq d)$.

Definitions:

Vector dominance. A vector $\mathbf{y}$ is said to be less than $\mathbf{x}$, $\mathbf{y}<\mathbf{x}$, iff for every $x_i\in\mathbf{x}$ and every $y_i\in\mathbf{y}$, $y_i\leq x_i$ and for some $x_k\in\mathbf{x}$, $y_k\in\mathbf{y}$, $y_k<x_k$. A vector $\mathbf{y}$ is said to be dominated by a vector $\mathbf{x}$ if $\mathbf{y}<\mathbf{x}$.

Minimal path vector at level *d*. A vector $\mathbf{x}$ is said to be a minimal path vector to level d if $\varphi(\mathbf{x})\geq d$ and for every other $\mathbf{y}<\mathbf{x}$, $\varphi(\mathbf{y})<d$ [6].

Minimal cut vector at level *d*. A vector $\mathbf{x}$ is said to be a minimal cut vector to level d if $\varphi(\mathbf{x})<d$ and for every other $\mathbf{y}>\mathbf{x}$, $\varphi(\mathbf{y})\geq d$ [6].

Assumptions:

(1) Component states are statistically independent.
(2) The structure function $\varphi(\mathbf{x})$ is coherent. That is, improvement of component states cannot damage the system.
(3) Component states and associated probabilities are known.

Acronyms:

MMCV	multi-state minimal cut vector
MMPV	multi-state minimal path vector
M2TR$_d$	multi-state two-terminal reliability
MC	Monte-Carlo

2. Literature review

2.1. Multi-state two-terminal reliability

For systems where binary state analysis is insufficient, incorrect reliability assessment can lead to faulty decision-making regarding network performance. Unnecessary expenditures, incorrect maintenance scheduling and reduction of safety standards are problems that can potentially be related to unsatisfactory reliability assessments. Thus, there is a real need to incorporate a more realistic view of system performance concerning multi-state behavior.

A first approach to solve this problem has been to use complete enumeration. That is, it is necessary to relate all possible combinations of component states to a system state. It is understandable that enumerative procedures can only be applied to relatively small systems.

Patra and Misra [11,12] propose a methodology that uses all the binary cut sets of a network to obtain M2TR. Arc capacity is considered to be a function of the degradation process that the arcs experience with time. Through an enumerative procedure, the binary cuts are employed to develop multi-state system state vectors that potentially satisfy system demand. Once this step is completed, each of these vectors is analyzed, filtering all the infeasible system state vectors that do not fulfill the required source-to-sink demand. However, the methodology is still an implicit enumerative approach and does not provide either paths or cuts at the multi-state level. Furthermore, in some cases this approach may lead to a complete enumeration procedure.

Lin [13] and Ramirez-Marquez and Coit [14] have developed approaches for computing the exact M2TR by resorting to the multi-state version of minimal path sets. The algorithm presented by Lin [13] considers weakly homogeneous components. That is, components can have different number of states yet for any two components, a_h and a_k with $|\mathbf{b}_h|=l_h$ and $|\mathbf{b}_k|=l_k$ and $l_h>l_k$, the first l_k component states must be equal. Thus, for systems where components are heterogeneous, the methodology may not be suitable. The methodology described by Ramirez-Marquez and Coit [14] relies on a network reduction technique to obtain all possible MMPV. It is interesting to note that this methodology, when considering binary components, reduces to the approach developed by Kuo et al. [4] for the binary case. Both of these approaches [13,14] have been applied to relatively small networks.

Lin [15] and Yeh [16] presented the first approaches to identify the multi-state version of minimal binary cut sets for the M2TR problem. These approaches compute MMCV in a similar form. For each binary cut, they enumerate all different state combinations of the components in the cut to obtain the multi-state levels that guarantee a MMCV. These methods consider weakly homogeneous components with the added constraint that the capacity of arc i is an integer-valued random variable. This research represented an important contribution. However, the algorithms cannot be applied to problems of the type Patra and Misra [11,12] present.

The algorithm presented by Ramirez-Marquez et al. [17] reduces the computational burden inherent in previous approaches [11–16]. The approach is based on two ideas. First, that all MMCV can be obtained from the set containing all minimal cuts [15,16], and second, that a select number of MMCV called offspring cuts inherit information from a select number of MMCV called parent cuts. This information sharing approach significantly reduces the number of vector enumerations needed to obtain all MMCV.

2.2. *M2TR$_d$ approximation methodologies*

For complex systems, exact reliability computation methods are not generally attractive. Alternatives are to approximate network reliability. One approximation approach is to use bounds, and another is to use simulation.

One of the preferred approaches for approximating 2TR is bounds. In the multi-state case, when considering MMCV, the bounds developed by Ramirez-Marquez et al. [19] can be used to approximate numerical reliability. They propose several bounds. The first bound, called MESP, is the multi-state extension of the binary bound originally proposed by Esary and Proschan [20]. It assumes the MMCV can be transformed into an equivalent series-parallel system to approximate M2TR$_d$. The second bound, called MLQ, is an extension of the binary LQ bound proposed by Jin and Coit [21] and it approximates reliability by considering the linear and quadratic unreliability of each MMCV. When considering MMPV, Ramirez et al. [19] present an extension of the 2TR bound proposed by Colbourn [22]. A lower bound on 2TR is constructed by considering a parallel-series system developed from disjoint minimal path sets. That is, each disjoint set is connected in parallel and each component within the set is considered as a component working in series. The extension relies on defining an equivalent definition of disjoint sets for the multi-state case.

Meng [23] developed multi-state extensions of the path-cut and mini-max binary lower bounds. However, these lower bounds impose restrictions on the multi-state structure function. For general multi-state systems, the main research issue addressed by Meng [23] is a comparison of the path-cut lower bound and the mini-max lower bound when system components are all highly reliable.

Prekopa et al. [24] approximate the probability of the union of n events with a linear programming model. They assume that the individual probabilities, as well as, some or all joint probabilities of up to m events, are known, where $m<n$. With this information, lower and upper bounds are proposed for the probability associated with the union of the n events. The bounds are obtained as optimum values of linear programming problems or objective function values corresponding to feasible solutions of the dual problems.

3. Monte-Carlo simulation for M2TR$_d$

As previously discussed, most other methodologies [13–16] assume the reliability computation for the M2TR$_d$ is straightforward once MMCV (or MMPV) have been determined. For large systems with even a relatively small number of states, this may not be the case. Although bounding methodologies have been proposed for this problem [19], numerical examples show that, in some cases, the approximation to the actual M2TR$_d$ may not be sufficiently accurate. Even when the bounds do provide accurate approximations, MC simulation provides an opportunity for even better approximations.

Simulation has been effectively used for approximating 2TR in relatively large systems [18]. As mentioned by Fishman [18] and Yeh [28], most methodologies for approximating system reliability in the binary case are based on the knowledge of minimal cut sets a priori. Yeh [28] mentions that relying on these sets becomes an additional issue to that of obtaining an accurate approximation. Because of this drawback, Yeh [28] extended the work done by Lu [29] to develop a simulation approach independent of the minimal cut sets. The application or extension of these tools to the multi-state case has not been reported in the literature. In this paper, information generated by other research results [13–18,28,29] are used to present a simulation methodology that accurately approximates M2TR$_d$.

In the proposed methodology, at each run of the simulation, a system state vector is generated based on the state occupancy probabilities dictated by vector $\mathbf{p}_i$. Once this state vector has been generated, it is compared against the set of MMCV to decide if such a vector corresponds to a system success, $\varphi(\mathbf{x})\geq d$, or to a system failure $\varphi(\mathbf{x})<d$. If the simulated vector is dominated or equal to even one of the MMCV, then it represents a system failure. At each run, the indicator variable Q (system failure count) is updated. Once a pre-selected number of runs have been exhausted, M2TR$_d$ can be approximated based on the current value of Q. The number of simulation runs, L, is dictated by desired accuracy. The methodology is as follows.

3.1. Procedure

```
Initialize:
  Determine MMCV with current methods [15–17],
  MMCV_t, t=1,...,u. s=0, L=number of simulation
  runs
  While s≤L
    Q_s→0, r=0, i=1, t=1;
    While i≤|A| {
      j=1, l=|b_i|;
      Generate a random number, r, from a uniform
      (0,1) distribution;
      If p_ij>r {
        x_i→b_ij;
        j→l+1;
        }
      Else {
        j→j+1;
        }
      While j≤l {
        If Σ_{k=1}^{j−1} p_ik < r ≤ Σ_{k=1}^{j} p_ik {
          x_i→b_ij;
          j→l+1;
        }
        Else {
          j→j+1;
        }
      }
      i→i+1;
    }
    x_s=(x_1,x_2,...,x_|A|)
    While t≤u {
      If x_s≤MMCV_t (x_s<MMCV_t or x_s=MMCV_t)
        Q_s→1;
        t→u+1;
      Else t→t+1;
    }Q→Q+Q_s;
    s→s+1;
    M2̂TR_d = 1 − Q/L;
```

3.2. Variance of M2TR$_d$ estimate

The simulation approach presented in Section 3.1 generates an estimate of the system unreliability. Let U be defined as the system unreliability. At each run s, the indicator variable Q_s tracks the system state corresponding to system state vector $\mathbf{x}_s$. Thus, the estimate of the system unreliability is given by [25]:

$$\hat{U}=\frac{\sum_{s=1}^{L}Q_s}{L}=\frac{Q}{L},$$

where L is the total number of generated system state vectors.

Since $\hat{U}$ generates an estimate of the actual M2TR$_d$ value, it is necessary to obtain the uncertainty related to such estimate. When the sample is large enough, the variance associated to $\hat{U}$ can be approximated by:

$$\mathrm{Var}(\hat{U})=\mathrm{Var}(\mathrm{M\hat{2}TR}_d)\approx\frac{\hat{U}-\hat{U}^2}{L} \tag{1}$$

The coefficient of variation of the distribution can be defined as in Eq. (2) Coefficient δ can be used to obtain the accuracy level of the estimate generated by MC simulation.

$$\delta\approx\sqrt{\mathrm{Var}(\hat{U})}\hat{U}=\sqrt{\frac{1-\hat{U}}{L\hat{U}}} \tag{2}$$

4. Experimental results and comparison of existing methodologies

In this section, three examples are used to obtain experimental results using the proposed simulation approach. The value of M2TR$_d$ obtained through the simulation is compared to other approximation methodologies available in the literature [19]. The network in Example 1 is used to validate the methodology. Examples 2 and 3 consider medium-sized and relatively large networks where the accuracy of traditional computational approaches may not be sufficiently accurate. The results presented in this section are then subjected to further analysis from two perspectives, computational effort and quality of results, in Section 5.

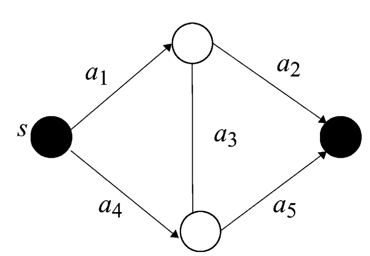

Fig. 1. Example 1 network.

Table 1a
Example 1 arc state data for cases 1–4

Arc	States				Probability (p_{ij})				Arc	States				Probability (p_{ij})			
Case 1									*Case 2*								
1	0	1	2	3	0.050	0.025	0.025	0.900	1	0	1	2	3	0.100	0.050	0.050	0.800
2	0	1	2		0.025	0.025	0.950		2	0	1	2		0.050	0.050	0.900	
3	0	1			0.050	0.950			3	0	1			0.100	0.900		
4	0	1			0.020	0.980			4	0	1			0.025	0.975		
5	0	1	2		0.075	0.025	0.900		5	0	1	2		0.150	0.050	0.800	
Case 3									*Case 4*								
1	0	1	2	3	0.150	0.075	0.075	0.700	1	0	1	2	3	0.100	0.150	0.150	0.600
2	0	1	2		0.075	0.075	0.850		2	0	1	2		0.100	0.100	0.800	
3	0	1			0.150	0.850			3	0	1			0.200	0.800		
4	0	1			0.050	0.950			4	0	1			0.100	0.900		
5	0	1	2		0.200	0.100	0.700		5	0	1	2		0.200	0.200	0.600	

4.1. Example 1

This first example [16] involves the identification of all MMCV guaranteeing that a demand greater than or equal to three units could not be delivered from source to sink. Fig. 1 illustrates the network. Table 1a presents four cases, with different arc state probabilities, that have been considered for analyzing M2TR$_d$. Table 1b presents the MMCV obtained by available methodologies [15–17]. Table 2 compares for each case the exact M2TR$_d$, obtained through inclusion/exclusion formula, against the different approximation schemes; MESP, MLQ and simulation. In this table and others, the number of significant digits is to demonstrate computation results, and not to indicate that the approximations can be expressed with that degree of precision. For the simulation approach 50,000 state vectors were randomly generated. As can be observed in the table, simulation provides a better estimate than the bounds in all cases.

4.2. Example 2

The network depicted in Fig. 2, called the ARPA network, is frequently used as an illustrative example in binary state reliability [21]. The ARPA network has loose connectivity and is presented here to demonstrate the simplicity of the methodology to obtain an estimate of M2TR$_d$. It is desired to obtain the probability that a demand of 10 units can be supplied from source to sink. Table 3 presents problem data for four cases that have been considered for analyzing M2TR$_d$ assuming different arc state probabilities. Table 4 presents the MMCV obtained by the Ramirez-Marquez et al. algorithm [17]. Finally, Table 5 presents the approximation results considering three different approaches.

4.3. Example 3

Fig. 3 shows a relatively large network, with 110 minimal cuts in the binary case. The problem is to find the probability that the capacity of the network from source to sink is greater or equal to five units. In the multi-state case, all arcs have the same three states, namely (0, 3, 5). Network 3 has 669 MMCV using Ramirez-Marquez et al. [17]. Clearly, the application of inclusion/exclusion formulae for this size of network is an inefficient and burdensome task. The procedure presented in Section 3 has been applied to obtain an estimate of M2TR$_5$. Table 6 presents problem data for four cases that have been considered for analyzing M2TR$_5$ assuming different arc state probabilities. Finally, Table 7 presents the approximation results considering three different approaches.

5. Performance and comparisons

The results obtained for the three examples presented in Section 4 have been analyzed considering two perspectives: computational effort and quality of results. For each of these perspectives, figures of merit are presented considering the approximation methodologies and the MC simulation approach. These analyses intend to clarify and illustrate when simulation-based approaches can be preferable to analytical approximation formulae.

5.1. Computational effort

For testing the MC simulation approach, statistics related to the computational effort have been recorded. The statistics presented are proportion of physical memory

Table 1b
Example 1 MMCV

$\mathbf{x}$	x_1	x_2	x_3	x_4	x_5
$\mathbf{x}_1$	3	0	1	1	2
$\mathbf{x}_2$	3	1	1	1	1
$\mathbf{x}_3$	3	2	1	1	0
$\mathbf{x}_4$	1	2	1	1	2
$\mathbf{x}_5$	2	2	1	0	2
$\mathbf{x}_6$	3	2	0	0	2
$\mathbf{x}_7$	3	1	0	1	2
$\mathbf{x}_8$	3	1	1	0	2

Table 2
$M2TR_3$ computation for Example 1

Case No.	Exact $M2TR_3$	Bounds		MC simulation	
		MESP	MLQ	$M2TR_3$	Variance $M2TR_3$
1	0.830990	0.824680	0.824935	0.830360	2.81725×10^{-6}
2	0.677599	0.659283	0.662031	0.676740	4.37526×10^{-6}
3	0.553735	0.519458	0.531444	0.551900	4.94613×10^{-6}
4	0.495120	0.439720	0.469800	0.493660	4.99920×10^{-6}

used, proportion of CPU usage, and CPU time. These same figures-of-merit have been obtained for the approximation formulae. Table 8 presents the results associated with 100,000 MC runs using a Pentium III laptop at 1000 MHz and 256 MB of RAM.

From these statistics it is evident that the MC simulation presents the highest computational effort of the estimation approaches. It must be noted that, as the probabilities of residing in the worst state increase, CPU time associated with the MC significantly reduces. This behavior can be related to a decrease in the number of comparisons done at the MMCV level.

5.2. Quality of results

The computational effort of the MC approach is considerable in comparison with approximation formulae. However, the accuracy of the results presented by each of the methodologies has been evaluated to provide evidence of the effectiveness of the MC approach.

Table 9 presents the absolute relative error of the methodologies for each of the cases of the first two examples. For both of these examples, the MLQ bound [9] presents the best estimate when considering bounding methodologies; the minimum and maximum error related to this formulae equals (0.2389 and 5.1139%). In contrast, the minimum and maximum error associated with the MC approach equals (0.0214 and 0.8665%). Table 9 provides evidence that for these relatively small networks, the simulation-based methodology consistently outperforms approximation methodologies from a solution quality perspective.

For the network presented in Example 3, the exact M2TR based on all 669 MMCV is not known. To further evaluate the accuracy of the approximations for Example 3, the MC approximation is computed and the values of both the MESP and MLQ are compared to obtain respective absolute relative errors. Table 10 presents the results of this comparison. For each of the cases, results are obtained through the proposed simulation approach, considering 500,000 runs. The effectiveness of the proposed methodology is evident when considering relatively large networks. The minimum and maximum absolute relative errors considering the bounding methodologies equal 0.2433 and 20.277%, respectively.

Figs. 4–6 illustrate how the simulation estimate converges to the exact $M2TR_d$ and provides a graphical view of how the approximations deviate at each run. Each of these figures present four graphs corresponding to the four cases run for each example in Section 4. They have been constructed by starting the number of simulation runs at 1000 and then subsequently incrementing this number by 1000 and conducting an independent simulation (with 2000 runs) until 100,000 is reached. As shown by the graphs the simulation approximates the M2TR value early on considering the number of replications.

Finally, it must be noted that in both Tables 9 and 10 the bounding methodologies exhibit an increase of the absolute relative error as the probability of residing in the highest state decreases. This behavior can be related to the mathematical form of the bounds and also to the impact of individual components.

Both the MESP and MLQ bounds [9] approximate the probability associated with the intersection of events as the product of the probability associated with each event. Consider the case of MMCVs $\mathbf{x}_k$ and $\mathbf{x}_h$. From the MLQ perspective:

$$P(\mathbf{x}\le\mathbf{x}_k \wedge \mathbf{x}\le\mathbf{x}_h) = P(\mathbf{x}\le\mathbf{x}_k)P(\mathbf{x}\le\mathbf{x}_h)$$

The actual expression can be written as:

$$P(\mathbf{x}\le\mathbf{x}_k \wedge \mathbf{x}\le\mathbf{x}_h) = P(\mathbf{x}\le\mathbf{x}_k|\mathbf{x}\le\mathbf{x}_h)P(\mathbf{x}\le\mathbf{x}_h)$$

That is $P(\mathbf{x}\le\mathbf{x}_k)$ approximates $P(\mathbf{x}\le\mathbf{x}_k|\mathbf{x}\le\mathbf{x}_h)$. When the probabilities of residing at the highest performance states are high, this approximation is often accurate because the probability associated to the shared state vectors is low.

The impact of components in the network system design is also important to explain the differences in the magnitude of the error associated with the bounding methodologies.

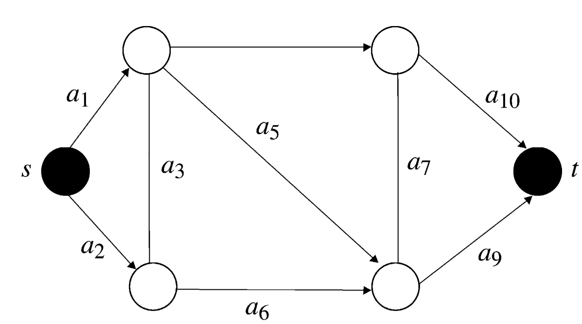

Fig. 2. ARPA network.

Table 3
ARPA network data

Arc	States				Probability (p_{ij})			
Case 1								
1	0	3	4	8	0.005	0.005	0.01	0.98
2	0	3	4	6	0.02	0.01	0.015	0.955
3	0	3			0.02	0.98		
4	0	3	4		0.01	0.015	0.975	
5	0	3			0.02	0.98		
6	0	3	6	0.005	0.02	0.975		
7	0	3	0.01	0.99				
8	0	3	4	6	0.01	0.015	0.005	0.97
9	0	3	4	8	0.02	0.01	0.01	0.96
Case 3								
1	0	3	4	8	0.05	0.025	0.025	0.9
2	0	3	4	6	0.03	0.02	0.05	0.9
3	0	3			0.1	0.9		
4	0	3	4		0.025	0.025	0.95	
5	0	3			0.1	0.9		
6	0	3	6		0.075	0.025	0.9	
7	0	3			0.025	0.975		
8	0	3	4	6	0.05	0.03	0.02	0.9
9	0	3	4	8	0.03	0.025	0.025	0.92

Arc	States				Probability (p_{ij})			
Case 2								
1	0	3	4	8	0.01	0.02	0.02	0.95
2	0	3	4	6	0.025	0.025	0.025	0.925
3	0	3			0.05	0.95		
4	0	3	4		0.025	0.025	0.95	
5	0	3			0.05	0.95		
6	0	3	6		0.01	0.04	0.95	
7	0	3			0.025	0.975		
8	0	3	4	6	0.02	0.02	0.01	0.95
9	0	3	4	8	0.04	0.01	0.01	0.94
Case 4								
1	0	3	4	8	0.1	0.05	0.05	0.8
2	0	3	4	6	0.05	0.075	0.075	0.8
3	0	3			0.2	0.8		
4	0	3	4		0.05	0.05	0.9	
5	0	3			0.2	0.8		
6	0	3	6		0.01	0.14	0.85	
7	0	3			0.05	0.95		
8	0	3	4	6	0.02	0.03	0.05	0.9
9	0	3	4	8	0.04	0.1	0.01	0.85

Table 4
ARPA network MMCV at level 10

MMCV	State vector								
	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9
1	8	6	3	4	3	6	3	0	8
2	8	6	3	4	3	6	3	4	4
3	8	6	3	4	3	6	3	6	3
4	8	6	3	4	3	6	0	6	4
5	8	6	3	0	3	6	3	6	8
6	8	6	3	3	3	3	3	6	8
7	8	6	3	3	0	6	3	6	8
8	8	6	3	4	0	3	3	6	8
9	8	6	3	4	3	0	3	6	8
10	4	6	0	4	3	3	3	6	8
11	8	3	0	3	3	6	3	6	8
12	8	6	3	4	0	6	0	3	8
13	8	6	3	4	3	3	0	3	8
14	8	3	0	4	3	6	0	3	8
15	8	3	0	4	0	6	3	3	8
16	8	4	0	4	0	6	0	4	8
17	8	3	0	4	0	6	0	6	8
18	3	6	3	4	3	6	3	6	8
19	4	4	3	4	3	6	3	6	8
20	8	0	3	4	3	6	3	6	8

As illustrated by the examples, two particular behaviors are evident:

(1) When the components probability of residing in a perfect performance state is high, the approximation formulae yield accurate results. For these cases, approximation expressions may be preferable to MC simulation approach from a computational efficiency perspective.

Table 5
Approximation results for Example 2

Case No.	Bounds		MC simulation	
	MESP	MLQ	$M2TR_{10}$	Variance $M2TR_{10}$
1	0.913941	0.9139989	0.915420	1.54852×10^{-6}
2	0.836295	0.8367996	0.844620	2.62474×10^{-6}
3	0.696673	0.7008271	0.718620	4.04411×10^{-6}
4	0.556872	0.5723886	0.604260	4.7826×10^{-6}

(2) When the components probability of residing in a perfect performance state is low, the approximation formulae achieves poorly. For these cases, MC simulation approach is preferable.

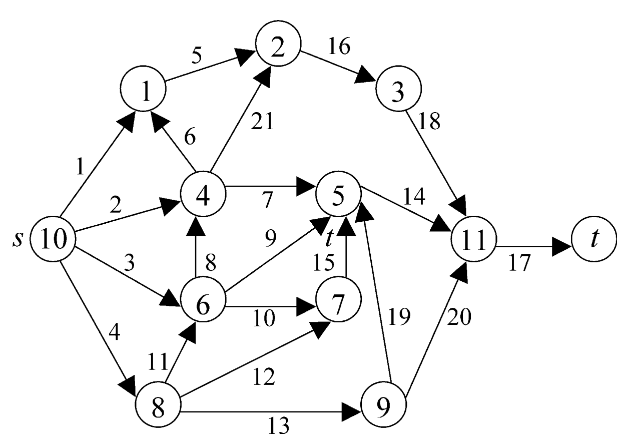

Fig. 3. Network 3.

Table 6
Network 3 data

Arc	States			Case 1			Case 2			Case 3			Case 4		
				State probability (p_{ij})			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	0.0539	0.1442	0.8019
2	0	3	5	0.1624	0.1224	0.7152	0.0199	0.0017	0.9784	0.1614	0.0474	0.7913	0.2781	0.0867	0.6352
3	0	3	5	0.2014	0.0900	0.7086	0.0294	0.0208	0.9499	0.0448	0.0729	0.8823	0.2213	0.2705	0.5082
4	0	3	5	0.0689	0.1155	0.8156	0.0606	0.0417	0.8977	0.0982	0.0039	0.8979	0.2142	0.0008	0.7850
5	0	3	5	0.1863	0.1366	0.6771	0.0188	0.0076	0.9736	0.1820	0.1068	0.7112	0.0065	0.0406	0.9529
6	0	3	5	0.2244	0.0214	0.7542	0.0995	0.0328	0.8676	0.0190	0.1769	0.8041	0.2920	0.0039	0.7041
7	0	3	5	0.2220	0.1334	0.6445	0.0072	0.0325	0.9602	0.0159	0.1230	0.8611	0.2251	0.0984	0.6764
8	0	3	5	0.1265	0.0762	0.7973	0.0698	0.0337	0.8965	0.1198	0.0032	0.8770	0.2562	0.1775	0.5663
9	0	3	5	0.2993	0.0343	0.6664	0.0852	0.0483	0.8665	0.1082	0.0007	0.8911	0.1731	0.2941	0.5327
10	0	3	5	0.3016	0.0813	0.6171	0.0540	0.0498	0.8962	0.0153	0.1136	0.8711	0.2603	0.1906	0.5491
11	0	3	5	0.2385	0.0785	0.6830	0.0572	0.0352	0.9076	0.0563	0.0778	0.8660	0.0335	0.0882	0.8783
12	0	3	5	0.3460	0.0269	0.6272	0.0191	0.0375	0.9434	0.0673	0.1711	0.7616	0.2170	0.2176	0.5654
13	0	3	5	0.3512	0.0441	0.6048	0.0690	0.0272	0.9038	0.1830	0.1752	0.6418	0.2514	0.1265	0.6221
14	0	3	5	0.0326	0.0182	0.9492	0.0672	0.0017	0.9310	0.0411	0.1582	0.8008	0.2570	0.0460	0.6970
15	0	3	5	0.0231	0.1268	0.8501	0.0197	0.0117	0.9685	0.0309	0.0766	0.8924	0.0710	0.1429	0.7861
16	0	3	5	0.0373	0.0830	0.8797	0.0457	0.0066	0.9477	0.0244	0.1499	0.8256	0.1095	0.2356	0.6549
17	0	3	5	0.0222	0.0192	0.9586	0.0490	0.0378	0.9132	0.0131	0.0753	0.9116	0.1099	0.1157	0.7744
18	0	3	5	0.0052	0.0411	0.9537	0.0243	0.0082	0.9675	0.0347	0.0174	0.9478	0.1946	0.1147	0.6908
19	0	3	5	0.3935	0.0625	0.5440	0.0830	0.0465	0.8705	0.0559	0.1944	0.7498	0.1965	0.1340	0.6695
20	0	3	5	0.0651	0.0457	0.8893	0.0314	0.0137	0.9549	0.1740	0.1877	0.6384	0.2618	0.2880	0.4502
21	0	3	5	0.1260	0.0495	0.8245	0.0503	0.0356	0.9141	0.1905	0.0572	0.7523	0.0280	0.1524	0.8197

Table 7
Approximation results for Example 3

Case No.	Bounds		MC simulation	
	MESP	MLQ	$M2TR_5$	Variance $M2TR_5$
1	0.930707	0.9307282	0.94692000	0.00000101
2	0.9104612	0.9104615	0.91200000	0.00000161
3	0.8909431	0.8909669	0.89992000	0.00000180
4	0.536264104	0.555777027	0.68354000	0.00000433

Table 8
Computing performance and comparisons

Case	Physical memory (%)			CPU usage (%)			CPU time (10^{-3} s)		
	Bounds		MC	Bounds		MC	Bounds		MC
	MESP	MLQ		MESP	MLQ		MESP	MLQ	
Example 1									
1	0.0023	0.1984	0.1842	1	10	30	0	0	190
2	0.0019	0.2322	0.1584	1	11	27	0	0	180
3	0.0023	0.1792	0.1938	1	11	28	0	0	170
4	0.0031	0.2199	0.1676	1	10	22	0	0	160
Example 2									
1	0.0096	0.1715	0.4137	2	8	59	0	0	560
2	0.0123	0.2295	0.4768	3	10	65	0	0	540
3	0.0077	0.2307	0.4922	2	12	58	0	0	520
4	0.0065	0.1569	0.4276	3	9	58	0	0	470
Example 3									
1	0.0165	0.2007	0.4583	4	9	98	0	10	23994
2	0.0246	0.2184	0.8378	4	11	99	0	10	23053
3	0.0192	0.3161	0.5122	4	10	99	0	10	22762
4	0.0231	0.2230	0.6690	4	8	99	0	10	17445

Table 9
Quality performance for Examples 1 and 2

Case	Exact M2TR3	Bounds		MC	Relative error (%)			CPU time (10^{-3} s)
		MESP	MLQ		MESP	MLQ	MC	
Example 1								
1	0.830990	0.824680	0.824935	0.830360	0.7593	0.7286	0.0758	560
2	0.677599	0.659283	0.662031	0.676740	2.7031	2.2975	0.1267	540
3	0.553735	0.519458	0.531444	0.551900	6.1901	4.0256	0.3314	520
4	0.495120	0.439720	0.469800	0.493660	11.1892	5.1139	0.2949	470
Example 2								
1	0.916188	0.913941	0.913999	0.91542	0.2452	0.2389	0.0838	560
2	0.84439	0.836295	0.8368	0.84462	0.9587	0.8989	0.0273	540
3	0.718466	0.696673	0.700827	0.71862	3.0333	2.4551	0.0214	520
4	0.599069	0.556872	0.572389	0.60426	7.0438	4.4537	0.8665	470

Table 10
Quality performance for approximation formulae in Example 3

Case	MC	Bounds		Relative error (%)		CPU time (10^{-3} s)
		MESP	MLQ	MESP	MLQ	
1	0.946396	0.930758	0.93078	1.6523	1.6501	129325
2	0.91272	0.9105	0.9105	0.2433	0.2432	130868
3	0.900228	0.891111	0.891134	1.0128	1.0102	127152
4	0.678196	0.540677	0.55924	20.2772	17.5400	99112

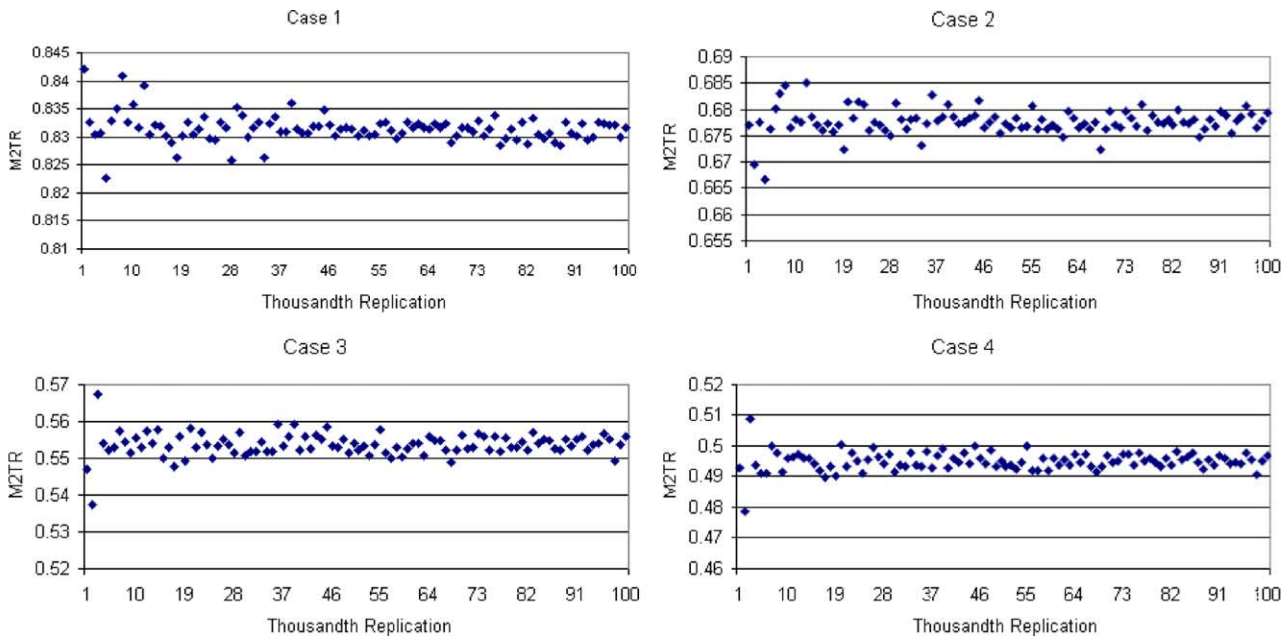

Fig. 4. Example 1 M2TR convergence.

The previous cases consider occupancy probabilities in the same range for all components. However, for cases when some of the component probabilities significantly differ, the MC approach offers considerably better results than the approximation schemes. This behavior is related to the impact components have on the system design with regard to multi-state reliability. To illustrate this discussion, consider Example 2 presented in Section 4, based on the methods presented by Ramirez-Marquez and Coit [17,26]. M2TR is recomputed assuming high and low reliability for particular components.

Table 11 presents bounding and MC results obtained by assuming that for each of the components $p(b_{il})=0.98$ (l being the highest state) and the remaining probability is equally shared by the rest of the states, i.e. $1,2,\ldots,l-1$. Moreover, different values for the probabilities associated to the states of component three and component nine are considered while maintaining all other components at the high probability of residing in the best performance state. Table 11 illustrates that the probability associated with component nine has the greater impact on the accuracy of the bounds. In contrast, the absolute relative error exhibits

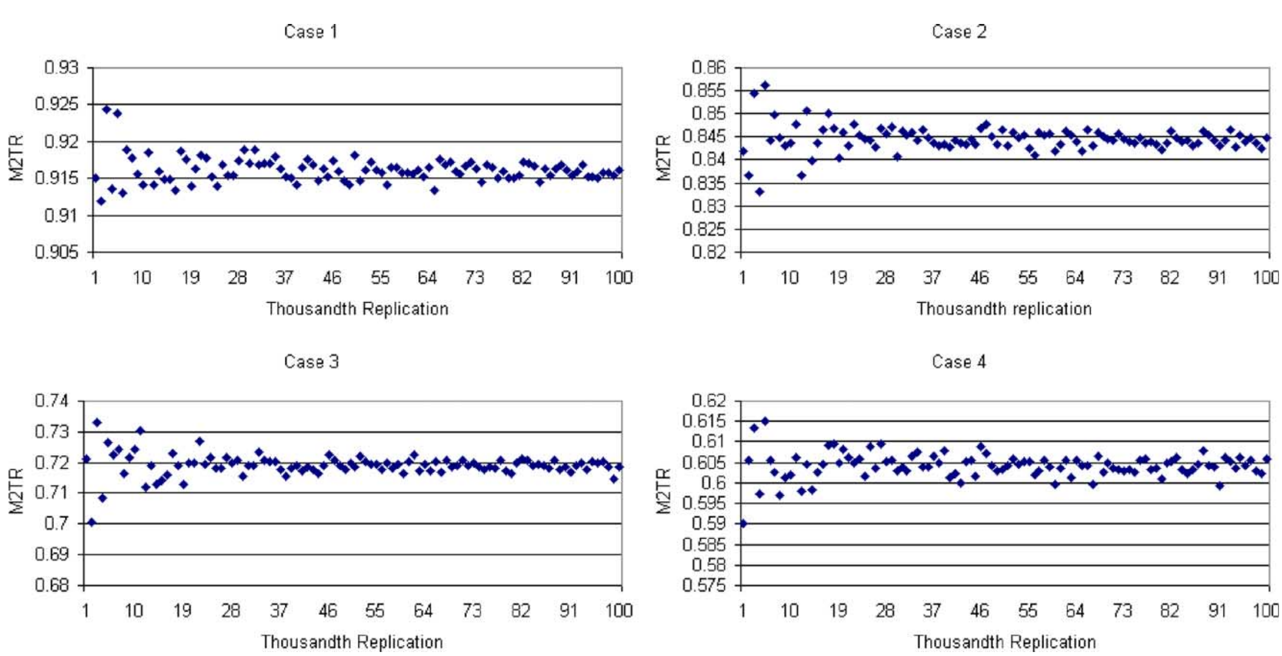

Fig. 5. Example 2 M2TR convergence.

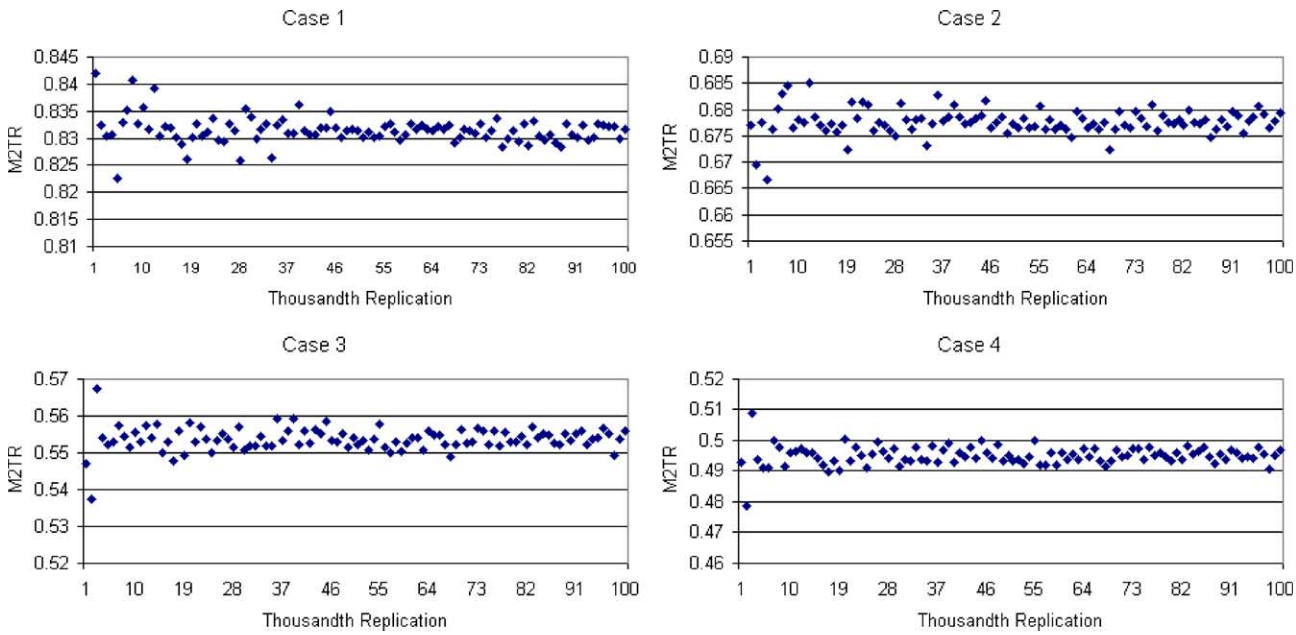

Fig. 6. Example 3 M2TR convergence.

a constant behavior when considering different values of component three.

Table 12 considers the same approach illustrated in Table 11 but considering two sets of components namely, set 1 = {3, 5, 7} and set 2 = {2, 4, 9}. The variation of component probabilities in set 1 generates an associated maximum error less than 1.3% for the bounding methodologies, yet when set 2 is considered, the maximum error increases significantly to 7.3849%.

The accuracy of the approximation approaches cannot be regarded as only a function of the components probability. As shown in these examples, approximation methodologies accuracy will be highly dependant on the component impact on system design with respect to multistate reliability.

Table 11
Component 3 and 9 impact on approximation approaches

Case	Bounds		MC	Relative error (%)	
	MESP	MLQ		MESP	MLQ
$p(b_{il})=0.98$	0.942613	0.942632	0.94368	0.1130	0.1110
$p(b_{3l})=0.9$	0.942562	0.942581	0.94368	0.1184	0.1164
$p(b_{3l})=0.8$	0.942504	0.942523	0.94364	0.1204	0.1183
$p(b_{3l})=0.7$	0.942435	0.942454	0.94365	0.1288	0.1267
$p(b_{9l})=0.9$	0.850217	0.850336	0.85335	0.3672	0.3532
$p(b_{9l})=0.8$	0.823251	0.823419	0.82786	0.5568	0.5364
$p(b_{9l})=0.7$	0.654617	0.655062	0.6633	1.3091	1.2419

Table 12
Set 1 and set 2 impact on approximation approaches

Case	Bounds		MC	Absolute error (%)	
	MESP	MLQ		MESP	MLQ
$p(b_{il})=0.98$	0.942613	0.942632	0.94368	0.1130	0.1110
Set 1					
$p(b_{\mathrm{Set1}l})=0.9$	0.931013	0.931049	0.93517	0.4445	0.4406
$p(b_{\mathrm{Set1}l})=0.8$	0.931748	0.931786	0.93900	0.7723	0.7683
$p(b_{\mathrm{Set1}l})=0.7$	0.917658	0.917728	0.92962	1.2868	1.2792
Set 2					
$p(b_{\mathrm{Set2}l})=0.9$	0.826675	0.827127	0.83151	0.5814	0.5271
$p(b_{\mathrm{Set2}l})=0.8$	0.695743	0.698193	0.70406	1.1812	0.8333
$p(b_{\mathrm{Set2}l})=0.7$	0.577279	0.584475	0.62331	7.3849	6.2304

Because this impact can only be assessed once the system reliability is obtained, the MC simulation methodology can be considered as a valid alternative when component probabilities differ.

6. Conclusions

A Monte-Carlo simulation approach has been developed for estimating $M2TR_d$. $M2TR_d$ is defined as the probability that a demand of d units can be supplied from source to sink through multi-state arcs. The new methodology effectively estimates $M2TR_d$ if MMCV or MMPV is known. Additionally, it has been discussed that most methodologies [13–16] assume the reliability computation for the $M2TR_d$ based on MMCV (MMPV) is straightforward. For large systems, even with a relatively small number of states, this may not be true. Although bounding methodologies have been proposed for this problem [19], numerical examples show that in some cases, the approximation to the actual $M2TR_d$ may result in an absolute error as large as 20%. Moreover, it has been shown that a particular component can have a significant impact on the accuracy of approximation formulae. The results presented show that if important components state probabilities are significantly different, the approximation results may be far from accurate. In contrast the proposed approach is not dependant on these distinct probabilities, and as the results indicate, it consistently yields accurate results. As evidenced by the results, simulation is a valid approach to obtain fairly accurate approximations to the actual reliability in a reduced computational time.

References

[1] Hansler E. A fast recursive algorithm to calculate the reliability of a communication network. IEEE Trans Commun 1972;20(3):637–40.

[2] Dotson W, Gobien J. A new analysis technique for probabilistic graphs. IEEE Trans Circuits Syst 1979;26(10):855–65.

[3] Torrierri D. Calculation of node-pair reliability in large networks with unreliable nodes. IEEE Trans Reliab 1994;43(3):375–9.

[4] Kuo S, Lu S, Yeh F. Determining terminal pair reliability based on edge expansion diagrams using OBDD. IEEE Trans Reliab 1999; 48(3):234–46.

[5] Elsayed E. In: Reliability engineering. Addison Wesley Longman Inc; 1996.

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

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

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

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

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

[11] Patra S, Misra B. Reliability evaluation of flow networks considering multistate modeling of network elements. Microelectron Reliab 1993; 33(14):2161–4.

[12] Patra S, Misra B. Evaluation of probability mass function of flow in a communication network considering a multistate model of network links. Microelectron Reliab 1996;36(3):415–21.

[13] Lin Y. A simple algorithm for reliability evaluation of a stochastic-flow network with node failure. Comput Oper Res 2001;28:1277–85.

[14] Ramirez-Marquez JE, Coit D. Alternative approach for analyzing multistate network reliability. Proceedings of the Industrial Engineering Research Conference (IERC), Portland, OR; May 2003.

[15] Lin Y. Using minimal cuts to evaluate the system reliability of a stochastic-flow network with failures at nodes and arcs. Reliab Eng Syst Saf 2002;75:41–6.

[16] Yeh W. A simple MC-based algorithm for evaluating reliability of a stochastic-flow network with unreliable nodes. Reliab Eng Syst Saf 2004;83(1):47–55.

[17] Ramirez-Marquez JE, Coit D, Tortorella M. Multistate two-terminal reliability—a cut approach. Rutgers University IE Working Paper 03-135 (under review, IEEE Trans Reliab); 2003.

[18] Fishman G. A comparison of four Monte Carlo methods for estimating the probability of s–t connectedness. IEEE Trans Reliab 1986; 35(2):145–54.

[19] Ramirez-Marquez JE, Coit D, Tortorella M. Bounds for multistate network two-terminal reliability. Rutgers University IE Working Paper 03-121 (under review, IEEE Trans Reliab); 2003.

[20] Esary J, Proschan F. Coherent structures of non-identical components. Technometrics 1963;5(2):191–209.

[21] Jin T, Coit D. Network reliability estimates using linear and quadratic unreliability of minimal cuts. Reliab Eng Syst Saf 2003;82(1):41–8.

[22] Colbourn C. Edge-packings of graphs and network reliability. Discrete Math 1988;72:49–61.

[23] Meng FC. A note on some reliability bounds for multistate systems. J Acad Sinica 2001;.

[24] Prekopa A, Vizvari B, Regos G, Gao L. Bounding the probability of the union of events by the use of aggregation and disaggregation in linear programs. RUTCOR Research Report 4-2001; 2001.

[25] Billinton R, Li W. Reliability assessment of electric power systems using Monte Carlo methods. New York: Plenum Press; 1994.

[26] Ramirez-Marquez JE, Coit D. Composite importance measures for multistate systems with multistate components. Rutgers University IE Working Paper 04-010; 2004.

[27] Lisnianski A, Levitin G. Multi-state system reliability. Assessment, optimization and applications. Singapore: World Scientific; 2003.

[28] Yeh W. A new Monte Carlo method for estimating network reliability. Proceedings of the 16th International Conference on Computers and Industrial Engineering, Ashikaga, Japan; 1994.

[29] Lu J. Studies on the Monte Carlo simulation methods for estimating network reliability. Master dissertation of the National TsingHua University, ROC, 1992.