IOSR Journal of Electronics and Communication Engineering (IOSR-JECE)
e-ISSN: 2278-2834,p- ISSN: 2278-8735.Volume 14, Issue 1, Ser. II (Jan.-Feb. 2019), PP 17-25
www.iosrjournals.org

Reliability Evaluation of Communication Flow Network Considering a Multi-State System

Mohamed AbdulRahman Al-Amoudi
Department of Electrical and Computer Engineering, Faculty of Engineering,
King Abdulaziz University, P. O. Box 108240, Jeddah, 21351, Saudi Arabia
Corresponding Author: Mohamed AbdulRahman Al-Amoudi

Abstract: *This paper presents a comparison between some methods for evaluation the reliability of a multi-state communication flow networks systems. A system is considered reliable only if successfully transmits at least the required system capacity from the transmitter to the receiver station. A main method used is the Brute Force Solution Exhaustive (BFSE) which gives the reliability at several required cases (By using Matlab code), the second method used all successful states are obtained from the knowledge of the max-flow min-cuts of the system graph. Examples are solved to demonstrate the applicability of proposed methods. It shows the matching results and full data.*

Keywords: *Brute Force Solution Exhaustive (BFSE), Multi-State System (MSS),* Artificial Neural Networks (ANN), *Reliability analysis, Sink Node.*

Date of Submission: 13-02-2019 Date of acceptance:28-02-2019

I. Introduction

Reliability evaluation of flow-communication networks considering two states of its elements (branches as well as nodes) has been well covered in the literature [1-4]. In evaluating reliability considering two states for the elements (branches as well as nodes), each element is assigned only one fixed capacity, i.e., the maximum flow capacity of that element. Whenever a branch or a node cannot transmit the maximum flow it is said to be in a failed state, although in such a state, the branch or node may still be able to transmit an appreciable amount of flow. Hence, many intermediate states can be introduced, assigning capacities to each state less than the maximum and, therefore, multistate modelling of the network elements is necessary to simulate the practical conditions. Thus, in a multistate model a network element has many states, where each state is assigned a capacity less than the maximum flow capacity of that element. For example, in a telecommunication network, the exchanges (represented as nodes in a reliability graph) may be completely or partly healthy. They are said to be partly healthy when one or two selectors are down, out of say 20 in a 1000 line exchange, consequently reducing the capacity of the exchange. It is undesirable to consider the exchange to be in a failed state when the capacity is less than the maximum. Similarly, all the toll-connecting trunks or inter-toll trunks (referred to as branches in a reliability graph) in a telecommunication network may not be healthy all the time, though serving the purpose partially. There are many methods to study the evaluation reliability for multistate as in [5- 7].

For future work, Artificial neural networks (ANN) can be used to solve multi-state system as [8], (ANN) could surpass the capabilities of conventional computer-based pattern recognition systems. An artificial neural network seeks to emulate the function of the biological neural network that makes up the brains found in nearly all higher life forms found on Earth. Neural networks are made up of neurons. The original goal of the ANN approach was to solve problems in the same way that a human brain would. However, over time, attention moved to performing specific tasks, leading to deviations from biology. Artificial neural networks (ANN) have been used on a variety of tasks, including computer vision, speech recognition, machine translation, social network filtering, playing board and video games and medical diagnosis. Neurons are the most important units in the nervous system. There are approximately 100 billion neurons in the brain, each of which is amazingly complex in itself. From a simplistic viewpoint, a neuron is a basic processing unit. A neuron receives input from other neurons, processes and integrates it, then bases its output (or lack thereof) on this integration, Training using BPNN is suggested to be performed each 100 DVB-H super-frames and training is performed by sending training data over pilots of higher power and then the receiver performs the training process of the BPNN which identify the channel model. The receiver uses the pilots of 1 super-frame for the next 99 super-frames and training is repeated again each 100 super-frames by using the resulting weights, as best estimate weights, for the estimation of the next 100 frames. The receiver multiplies the data by the inverse channel model to perform the equalization.

II. Brute Force Solution Exhaustive (BFSE)

In computer science, brute-force search or exhaustive search, also known as generate and test, is a very general problem-solving technique and algorithmic paradigm that consists of systematically enumerating all possible candidates for the solution and checking whether each candidate satisfies the problem's statement.

A brute-force algorithm to find the divisors of a natural number *n* would enumerate all integers from 1 to n, and check whether each of them divides *n* without remainder. A brute-force approach for the eight queens puzzle would examine all possible arrangements of 8 pieces on the 64-square chessboard, and, for each arrangement, check whether each (queen) piece can attack any other.

While a brute-force search is simple to implement, and will always find a solution if it exists, its cost is proportional to the number of candidate solutions – which in many practical problems tends to grow very quickly as the size of the problem increases (combinatorial explosion). Therefore, brute- Force search is typically used when the problem size is limited, or when there are problem-specific heuristics that can be used to reduce the set of candidate solutions to a manageable size. The method is also used when the simplicity of implementation is more important than speed.

This is the case, for example, in critical applications where any errors in the algorithm would have very serious consequences; or when using a computer to prove a mathematical theorem. Brute-force search is also useful as a baseline method when benchmarking other algorithms or metaheuristics. Indeed, brute- Force search can be viewed as the simplest metaheuristic. Brute force search should not be confused with backtracking, where large sets of solutions can be discarded without being explicitly enumerated (as in the textbook computer solution to the eight queens problem above). The brute-force method for finding an item in a truth table – namely, check all entries of the latter, sequentially – is called linear search.

III. Assumptions

The proposed method is based on the following assumptions:
(i) The telecommunication network is modelled by a graph.
(ii) Each element (branches and nodes) can have multistate, i.e., a failed state with zero capacity, a state with maximum capacity and other states with capacity less than the maximum capacity.
(iii) The probability of a network element (branches as well as nodes) being in a particular state is known.

IV. Examples

The proposed method evaluates the reliability of a flow network considering the multistate model of the network elements. It assumes the knowledge of the following.
(i) Graph structure, *i.e.*, the number of nodes, number of branches and the identification of each branch with an ordered pair of nodes; the starting node and the terminating node.
(ii) The source (initial) can be as main Exchange node and the sink (terminal) node can be as branch Exchange node.
(iii) The required system capacity.
(iv) The number of states of each branch and node.
(v) Capacities of all states of each branch and node.
(vi) states Probabilities of assigned units of flow for all of each branch and node.

Example A

As Fig.1 shows the Graph Network, connect source to terminal , with five differents cables a, b, c, d and e , with differents capacity and probability as showing in Table. 1A

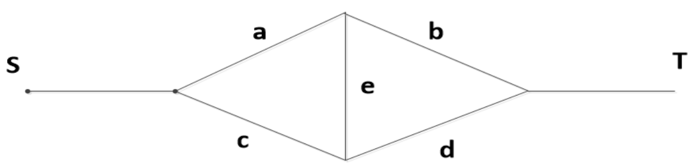


Fig. 1. Grpah Network Example A

In each exapmple will solve it by two methods, first will work with evaluation of multistate flow networks using the state enumeration technique. The proposed method considers cutsets of the network and examine coressponding capacity vectors for the required flow capacity from the source node to the sink node. Reliability is evaluted through enumeration of selective states from enumerated capacity vectors .

Table 1A. cables Network capacity with their probabilities

Cable Line	C	P(C)
a	0	.1
	2	.3
	4	.6
b	0	.2
	3	.8
c	0	.3
	1	.3
	3	.4
d	0	.3
	4	.7
e	0	.3
	3	.7

There are 5 Minimal cuts:(a+c, b+d, a+e+d, b+e+c)

Case 1 : if The required system capacity is Cs= 6 units (Mega bits/second).

Table 2AC1. Listing of all the possible states of cuts and capacity of cut for the system graph of Fig. 1A

a	b	c	d	e	capacity	Successful states
0	—	0	—	—	0	No
0	—	1	—	—	1	No
0	—	3	—	—	3	No
2	—	0	—	—	2	No
2	—	1	—	—	3	No
2	—	3	—	—	5	No
4	—	0	—	—	4	No
4	—	1	—	—	5	No
4		3			7	Yes
—	0	—	0	—	0	No
—	0	—	4	—	4	No
—	3	—	0	—	3	No
—	3	—	4	—	7	Yes
4	—	—	4	0	8	Yes
4	—	—	4	3	11	Yes
—	3	3	—	0	6	Yes
—	3	3	—	3	9	Yes

Table 3AC1. Listing of all of the possible states of capacity vectors of branches, maximum capacity and probability for the system graph of Fig. 1A

All state					Min (capacity of all cuts)	Successful states	Probability
a	b	c	d	e			PaPbPcPdPe
4	3	3	4	0	6	yes	.04032
4	3	3	4	3	7	yes	.09408
Total reliability =							.1344

Case 2: The required system capacity is Cs= 7 units (Mega bits /second).

Table 2AC2. Listing of all the possible states of cuts and capacity of cut for the system graph

a	b	c	d	e	capacity	Successful states
0	—	0	—	—	0	No
0	—	1	—	—	1	No
0	—	3	—	—	3	No
2	—	0	—	—	2	No
2	—	1	—	—	3	No
2	—	3	—	—	5	No
4	—	0	—	—	4	No
4	—	1	—	—	5	No
4		3			7	Yes
—	0	—	0	—	0	No
—	0	—	4	—	4	No
—	3	—	0	—	3	No
—	3	—	4	—	7	Yes
4	—	—	4	0	8	Yes
4	—	—	4	3	11	Yes
—	3	3	—	0	6	No
—	3	3	—	3	9	Yes

Table 3AC2. Listing of all of the possible states of capacity vectors of branches, maximum capacity and probability for the system graph of Fig. 1

All state					Min (capacity of all cuts)	Successful states	Probability
a	b	c	d	e			PaPbPcPdPe
4	3	3	4	3	7	yes	.09408
Total reliability =							.09408

Case 3: The required system capacity is Cs= 8 units (Mega bits/second).

Table 2AC3. Listing of all the possible states of cuts and capacity of cut for the system graph

a	b	c	d	e	capacity	Successful states
0	—	0	—	—	0	No
0	—	1	—	—	1	No
0	—	3	—	—	3	No
2	—	0	—	—	2	No
2	—	1	—	—	3	No
2	—	3	—	—	5	No
4	—	0	—	—	4	No
4	—	1	—	—	5	No
4		3			7	No
—	0	—	0	—	0	No
—	0	—	4	—	4	No
—	3	—	0	—	3	No
—	3	—	4	—	7	No
4	—	—	4	0	8	Yes
4	—	—	4	3	11	Yes
—	3	3	—	0	6	No
—	3	3	—	3	9	Yes

Table 3AC3. Listing of all of the possible states of capacity vectors of branches, maximum capacity and probability for the system graph of Fig. 1

All state					Min (capacity of all cuts)	Successful states	Probability
a	b	c	d	e			PaPbPcPdPe
4	3	3	4	3	7	No	-
4	3	3	4	0	6	No	-
Total reliability =							0

Case 4 : The required system capacity is Cs= 5 units(Mega bits/second) .

Table 2AC4. Listing of all the possible states of cuts and capacity of cut for the system graph

a	b	c	d	e	capacity	Successful states
0	—	0	—	—	0	No
0	—	1	—	—	1	No
0	—	3	—	—	3	No
2	—	0	—	—	2	No
2	—	1	—	—	3	No
2	—	3	—	—	5	No
4	—	0	—	—	4	No
4	—	1	—	—	5	Yes
4		3			7	Yes
—	0	—	0	—	0	No
—	0	—	4	—	4	No
—	3	—	0	—	3	No
—	3	—	4	—	7	Yes
2	—	—	4	0	6	Yes
2	—	—	4	3	9	Yes
4	—	—	4	0	8	Yes
4	—	—	4	3	11	Yes
—	3	1	—	0	4	No
—	3	3	—	0	6	Yes
—	3	1	—	3	7	Yes
—	3	3	—	3	9	Yes

Table 3AC4. Listing of all of the possible states of capacity vectors of branches, maximum capacity and probability for the system graph of Fig. 1

All state					Min (capacity of all cuts)	Successfu l states	Probability
a	**b**	**c**	**d**	**e**			**PaPbPcPdPe**
2	**3**	**1**	**4**	**0**	**3**	**No**	**—**
2	**3**	**3**	**4**	**0**	**5**	**Yes**	**.02016**
2	**3**	**1**	**4**	**3**	**3**	**No**	**-**
2	**3**	**3**	**4**	**3**	**5**	**Yes**	**.04704**
4	**3**	**1**	**4**	**0**	**4**	**No**	**-**
4	**3**	**3**	**4**	**0**	**6**	**Yes**	**.04032**
4	**3**	**1**	**4**	**3**	**5**	**Yes**	**.07056**
4	**3**	**3**	**4**	**3**	**7**	**Yes**	**.09408**
Total reliability =							**0.27216**

Cs(required capacity)	**5**	**6**	**7**	**8**
Reliability	**0.27216**	**.1344**	**.09408**	**0**

Second method is the Brute Force Solution Exhaustive (BFSE) which is working by calculating the truth table for all cases for Table.1A, will be 72(=3*2*3*2*2) cases, by using Matlab code to calculate all these 72 cases.
Flow = min{cuts sets}= min{ a+c, b+d, a+e+d, b+e+c }

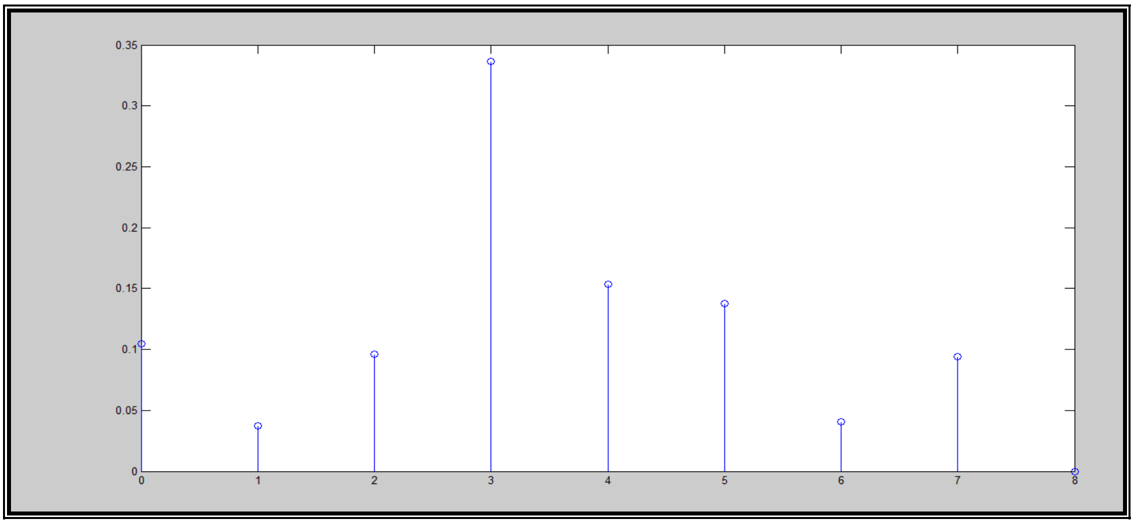
Table 4A. all 72 cases (states) Truth Table

a	b	c	d	e	Flow	Prob	a	b	c	d	e	Flow	Prob
0	0	0	0	0	0	0.00054	2	3	0	0	0	2	0.00648
0	0	0	0	3	0	0.00126	2	3	0	0	3	2	0.01512
0	0	0	4	0	0	0.00126	2	3	0	4	0	2	0.01512
0	0	0	4	3	0	0.00294	2	3	0	4	3	2	0.03528
0	0	1	0	0	0	0.00054	2	3	1	0	0	2	0.00648
0	0	1	0	3	0	0.00126	2	3	1	0	3	3	0.01512
0	0	1	4	0	1	0.00126	2	3	1	4	0	3	0.01512
0	0	1	4	3	1	0.00294	2	3	1	4	3	3	0.03528
0	0	3	0	0	0	0.00072	2	3	3	0	0	2	0.00864
0	0	3	0	3	0	0.00168	2	3	3	0	3	3	0.02016
0	0	3	4	0	3	0.00168	2	3	3	4	0	5	0.02016
0	0	3	4	3	3	0.00392	2	3	3	4	3	5	0.04704
0	3	0	0	0	0	0.00216	4	0	0	0	0	0	0.00324
0	3	0	0	3	0	0.00504	4	0	0	0	3	0	0.00756
0	3	0	4	0	0	0.00504	4	0	0	4	0	0	0.00756
0	3	0	4	3	0	0.01176	4	0	0	4	3	3	0.01764
0	3	1	0	0	0	0.00216	4	0	1	0	0	0	0.00324
0	3	1	0	3	1	0.00504	4	0	1	0	3	0	0.00756
0	3	1	4	0	1	0.00504	4	0	1	4	0	1	0.00756
0	3	1	4	3	1	0.01176	4	0	1	4	3	4	0.01764
0	3	3	0	0	0	0.00288	4	0	3	0	0	0	0.00432
0	3	3	0	3	3	0.00672	4	0	3	0	3	0	0.01008
0	3	3	4	0	3	0.00672	4	0	3	4	0	3	0.01008
0	3	3	4	3	3	0.01568	4	0	3	4	3	4	0.02352
2	0	0	0	0	0	0.00162	4	3	0	0	0	3	0.01296
2	0	0	0	3	0	0.00378	4	3	0	0	3	3	0.03024
2	0	0	4	0	0	0.00378	4	3	0	4	0	3	0.03024
2	0	0	4	3	2	0.00882	4	3	0	4	3	4	0.07056
2	0	1	0	0	0	0.00162	4	3	1	0	0	3	0.01296
2	0	1	0	3	0	0.00378	4	3	1	0	3	3	0.03024
2	0	1	4	0	1	0.00378	4	3	1	4	0	4	0.03024
2	0	1	4	3	3	0.00882	4	3	1	4	3	5	0.07056
2	0	3	0	0	0	0.00216	4	3	3	0	0	3	0.01728
2	0	3	0	3	0	0.00504	4	3	3	0	3	3	0.04032
2	0	3	4	0	3	0.00504	4	3	3	4	0	6	0.04032
2	0	3	4	3	4	0.01176	4	3	3	4	3	7	0.09408

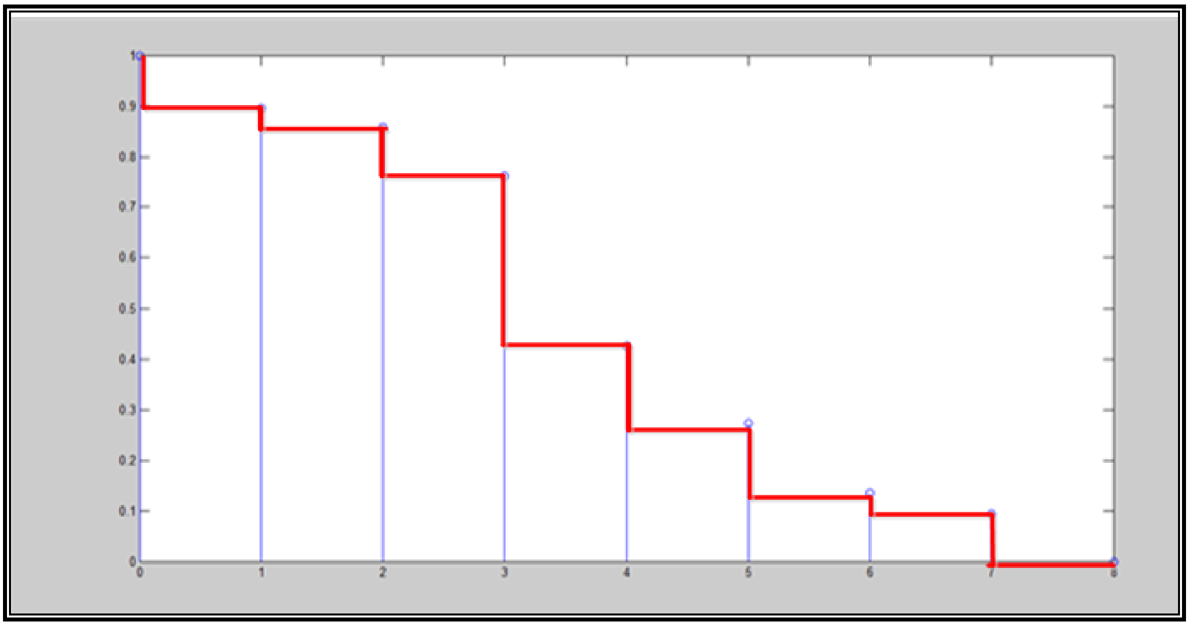
Table.5A . Probability mass function(PMF) and complementary cumulative Distribution Function (CCDF)

Flow	pmf	case	CCDF
0	**0.10458**	**F> -n**	**1**
1	**0.03738**	**F> 0**	**0.89542**
2	**0.09594**	**F> 1**	**0.85804**
3	**0.33622**	**F> 2**	**0.7621**
4	**0.15372**	**F> 3**	**0.42588**
5	**0.13776**	**F> 4**	**0.27216**
6	**0.04032**	**F> 5**	**0.1344**
7	**0.09408**	**F> 6**	**0.09408**
8	**0**	**F> 7**	**0**

Figure. 2. Two curves shows that the plot of Probability mass function(PMF) and complementary cumulative Distribution Function (CCDF) for Example A



PMF



CCDF

Figure 2. plots PMF and CCDF versus the all required cases

Example B

As Fig.3 shows the Graph Network [4], connect source to terminal , with seven differents cables a, b, c, d, e, f and g , with differents capacity and probability as showing in Table. 1B.

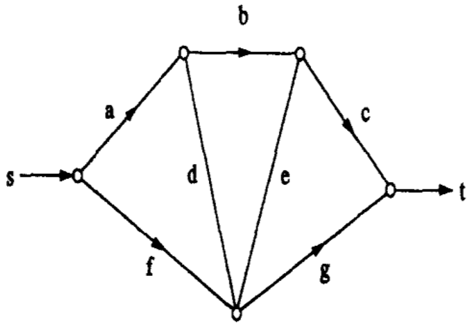


Fig. 3. Grpah Network Example B

Table 1B. cables Network capacity with their probabilities

symbol	Capacity	Probability (C)
a	0	.1
	2	.3
	4	.6
b	0	.2
	3	.8
c	0	.3
	1	.3
	3	.4
d	0	.3
	4	.7
e	0	.3
	3	.7
f	0	.2
	2	.3
	3	.5
g	0	.2
	3	.2
	4	.6

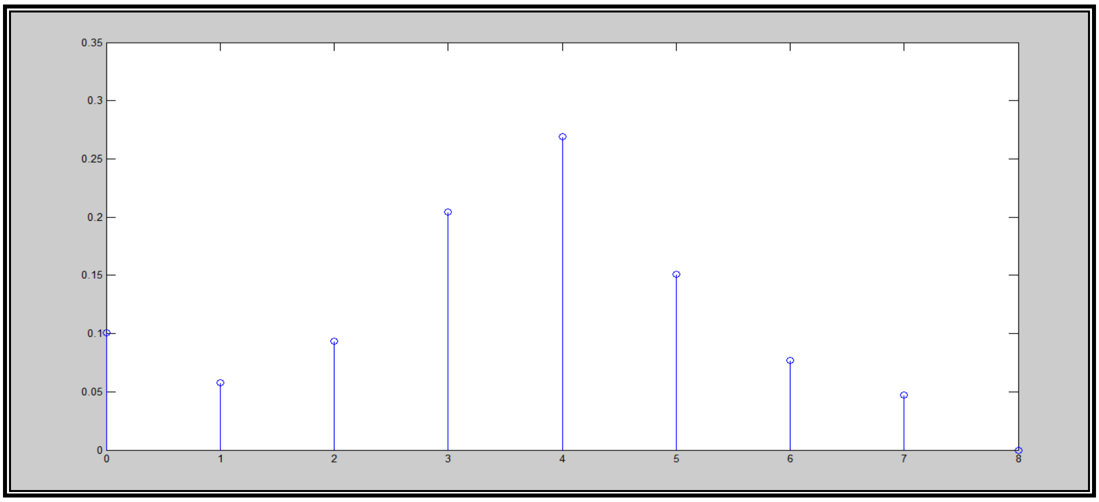
There are 6 Minimal cuts:(a+f, c+g, b+d+f, b+e+g, a+d+e+g , c+d+e+f)

As [4] describe that the required system capacity is Cs= 6 units , the reliability was =.1241088
Here, We have 648 states (cases)= 3*2*3*2*2*3*3 states

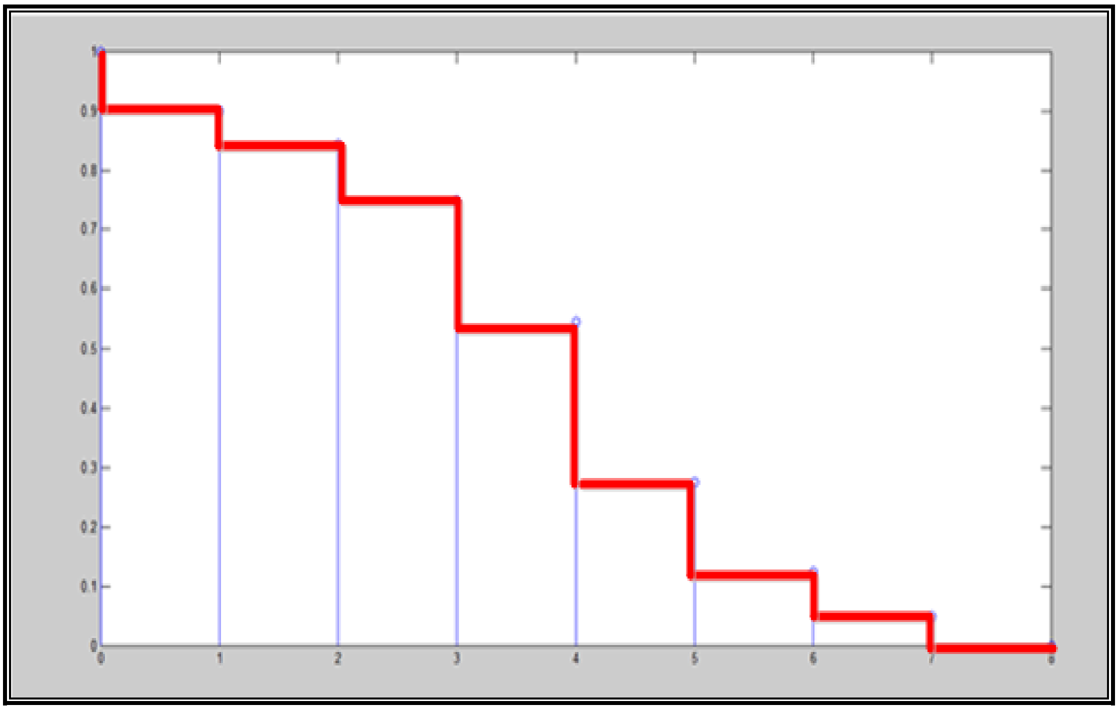
Table.2B . Probability mass function(PMF) and complementary cumulative Distribution Function (CCDF)

Flow	pmf	case	CCDF
0	0.10065	**F> -n**	1.00
1	0.05758	**F> 0**	0.899357
2	0.09336	**F> 1**	0.841774
3	0.20436	**F> 2**	0.748419
4	0.26935	**F> 3**	0.544059
5	0.1506	**F> 4**	0.274709
6	0.07673	**F> 5**	0.124109
7	0.04738	**F> 6**	0.047376
8	1.00	**F> 7**	0.00

It shows that, matching the results between [4] at required system capacity Cs=5 units (Mega /second) (Mb/S), but clear that , this method (BFSE) gives result at any required system capacity.
Figure. 4. Two curves shows that the plot of Probability mass function(PMF) and complementary cumulative Distribution Function (CCDF) for Example B



Pmf



CCDF

Figure 4. plots PMF and CCDF versus the all required cases

V. Conclusion

This paper presents a comparison between two methods of reliability evaluation of multistate flow communication network with required system capacity constraints . first, Obtained by max-flow min cut sets theorem which gives reliability at only certain required system capacity , but the second method the brute force solution exhaustive (BFSE) which shows the reliability at any required and all system capacity .
These methods brings to light the multistate modelling of a communication network. In practice, in a telecommunication network the exchanges (referred to as nodes in a reliability graph) and the communication links (referred to as branches in a reliability graph) do have multistate. We can see the increase in reliability of the network with multistate modelling, which is the actual reliability value. The increase in reliability with multistate modelling as compared with two-state modelling is clearly due to the fact that we ignored the intermediate states in the conventional two-state modelling. In the two-state model we considered a network element to be good when it could carry its maximum specified capacity and to be in its failed state when it carried less than the maximum specified capacity. However, in a capacity limited network it is unwise to consider only two states (good and failed) because when a network element transmits less than the maximum

specified units it can satisfy the required system flow condition in combination with other elements of the network, thus raising the system.

Acknowledgements

The author would like to thank his supervisor Prof. Dr. Ali M. Rushdi at King Abdulaziz University, Jeddah, Saudi Arabia for his kind and proficient help , support and guiding.

References

[1]. Lee, S. H. (1980). Reliability evaluation of a flow network. IEEE Transactions on Reliability, 29(1), 24-26.
[2]. Misra, K. B., & Prasad, P. (1982). Comment on" Reliability Evaluation of a Flow Network. IEEE Transactions on Reliability, 31(2), 174-176.
[3]. Sharma, U., Panigrahi, S. C., & Misra, R. B. (1990). Reliability evaluation of a communication system considering a multistate model. Microelectronics Reliability, 30(4), 701-704.
[4]. Patra, S., & Misra, R. B. (1993). Reliability evaluation of flow networks considering multistate modelling of network elements. Microelectronics Reliability, 33(14), 2161-2164.
[5]. Rushdi, A. M. A., & Al-Amoudi, M. A. (2018). Recursively-Defined Combinatorial Functions: The Case of Binomial and Multinomial Coefficients and Probabilities. Journal of Advances in Mathematics and Computer Science, 27(4), 1-16..
[6]. Rushdi, A. M. A., & Al-Amoudi, M. A. (2018). Switching-Algebraic Analysis of Multi-State System Reliability. Journal of Engineering Research and Reports, 1-22.
[7]. Rushdi, A. M. A., & Al-Amoudi, M. A. (2018). Reliability Analysis of a Multi-State System Using Multi-Valued logic. IOSR Journal of Electronics and Communication Engineering (IOSR-JECE). Volume 14, Issue 1, Ver. I (Jan.-Feb. 2019), PP 01-10.
[8]. Alamoudi, M. A., & Affandi, A. (2015). Simulation of Rayleigh channel and improve of channel estimation using Artificial Neural Network. Life Science Journal, 12(6).

IOSR Journal of Electronics and Communication Engineering (IOSR-JECE) is UGC approved Journal with Sl. No. 5016, Journal no. 49082.

Mohamed AbdulRahman Al-Amoudi. " Reliability Evaluation of Communication Flow Network Considering a Multi-State System." IOSR Journal of Electronics and Communication Engineering (IOSR-JECE) 14.1 (2019): 17-25.