Hierarchical Steady-State Availability Evaluation of Dynamic Fault Trees through Equal Markov Model

Z. Ramezani
PhD student
School of Electrical and Computer Engineering
Shahid Beheshti University
Tehran, Iran, 1658953571
Email: z_ramezani@sbu.ac.ir

G. Latif-Shabgahi
Faculty of Electrical and Computer Engineering
School of Electrical & Computer Engineering
Shahid Beheshti University
Tehran, Iran, 1658953571
Email: gh_latif@sbu.ac.ir

K. Aslansefat
MSc student
School of Electrical and computer Engineering
Shahid Beheshti University
Tehran, Iran, 1658953571
Email: k_aslansefat@sbu.ac.ir

Abstract— *Abstract*—**Evaluation of availability of systems is an important issue in many applications. One way to estimate the availability of dynamic systems is using their dynamic fault tree (DFT) model. Markov model is a method to solve the dynamic fault trees. However by increasing the tree size, the number of Markov states of is increased exponentially. This is why which cases, the solving of model complicated. The use of hierarchical methods is a technique to solve such this problem. This paper presents that and approach to address problem. Many published papers have used the Mont-Carlo approach for finding the availability of system from their DFT model. This paper uses the equal Markov model of DFT to estimate the system availability. First the equal Markov model is introduced, and the equal Markov model of existing dynamic gates will be constructed from their conventional Markov model. The equal Markov model of two systems is then constructed and their availability is computed. The results are them compared with those obtained from conventional Markov models to verify the correctness of our approach.**

Keywords- dynamical fault tree, Markov model, availability, equal failure rate, and equal repair rate.

I. INTRODUCTION

A. A review of evaluation methods:

One of the important issues in industry is evaluation of the availability of systems. Availability is the probability that a system performans its function correctly at any instance of time. It is in fact a nonstop operation index of a system. The use of self repair components, self diagnoses mechanisms, and online and offline repairs are mechanisms to improve a system availability. In the last three decades, a number of approaches including DFT, Mont-Carlo, Markov modeling and hybrid techniques have been used to calculate system's availability [1].

For constructing a dynamic fault tree, it is necessary fully to understand the topology of the system, and failure modes of its components [2-3]. In addition, the size of tree system is exponentially increased with increasing the number of components and subsystems. Fault tree is a graphical model that can be achieved to evaluate the system performance by accessing the failure paths [4].It is also divided into two categories, dynamic and static. A static fault tree can only model systems in which the relation between system components can be expressed by means of AND & OR gates. A dynamic fault tree, on the other hand, models many dynamic aspects and attributes of systems like priority, dependency, sequence, time, and reconfiguration by means of dynamics gates introduce in the last decade, (including PAND, Spare, FDEP, and etc) [5-7].

Research efforts have been looking for automated tools [8-11], able: to convert a given DFT into state-space models, to apply techniques to improve computational performance, to avoid the state-space explosion, like Stochastic Process Algebra, lumping or aggregation [12] or using local explorations of the state graph via sequences [13]. In [14], DFT has been solved via its conversion to a Bayesian network. This method mitigates the state space explosion, but like direct acyclic graph, the Bayesian network cannot model cyclic dependencies, so restoration is not allowed. Other approaches have used stochastic Petri nets as target formalism for solving DFTs [15].

The DFT formalism introduced in [16]; derives from a DFT model, its equivalent state space model in the form of continuous time Markov chain [17-18]. The state space analysis incorporates high computational costs. For this reason, a modular approach to analyze DFTs is proposed in [19].

Writer in [20] presented a transformation of DFTs to Stochastic Petri Nets [21], which were in turn analyzed by conversion to Markov Chains. Although this method was suffering from a combinatorial explosion when constructing the Markov Chain, the Petri Nets are much smaller and easier to understand and extend.

Reference [22] was introduced a software tool to analyze the reliability and availability of DFT by the use of I/O IMC methodology.

Monte Carlo does not require knowledge about the internal structure of system [23], it does not provide a closed formula for system reliability. However, due to its simplicity, many researchers in the field of power and communication sector have been used this methodology for finding the point reliability of systems. [24]

The Markov model is another approach to evaluate the reliability and availability of systems. Various extension of this state space based modeling approach like Semi Markov, hidden Markov, reward Markov have been proposed for basic Markov theory [24]. This approach has been widely used for the evaluation of the reliability, availability, sensitivity, safety, and perform ability of industrial systems, see for example [1, 2, 5, 17, 24_26]. In the past decades considerable advances, have been made in the hierarchical solution techniques methods of automated model generation, and the availability of software tools. A Markov chain consists of a set of states and a set of transitions between the states. A state can model various condition of interest in the system being studied. This could be the number of jobs of various types waiting to use which resource the number of modules, that have failed, the number of modules perform correctly and so on. After a sojourn in a state, the Markov model will make a transition to another state. Such transitions are labeled either with probabilities of transition or rates of transition. If the Markov model has appropriate structure, it is often possible to avoid the generation and solution of the underlying large state space (explosion).

One way to solve the problem of state explosion in large Markov models is the use of hierarchical approach [2]. This reference introduces a “reward Markov model”. In which, when the state is active, the reward rate is set to zero, and when the state is failed, the reward rate is set to one. Thus a given Markov model can be converted to a two states model, with two transition between the states [2]. One of transitions is assigned to “an equal failure rate” and the other is assigned to “an equal repair rate”.

B. Problem Stating

As stated in previous section (A), one of the widely, used methods, for availability evaluation of systems is DFT approach. The Markov model is usually used for solving DFT modules. The introduced approaches in the literature have been faced with Markov models with a large number of states specially in the case of modeling large systems.

This paper proposes a hierarchical method to solve this problem. Our methodology converts the Markov model of each dynamic gate in to a simple two state Markov model. In this way the size of original Markov model is dramatically decreased. This, in turn, simplifies the computation and decreases the evaluation phase.

This paper is organized as follows: after this introduction, Section II proposes the method of obtaining equal failure and repair rates of dynamic fault tree by using Markov model, and then the method for calculation of steady state availability is presented. In Section III, we introduce some dynamic gates with equal Markov model. In section IV, the effectiveness of the proposed method is illustrated through examples. Finally, a short conclusion is given in last section.

II. EVALUATION OF STEADY STATE AVAILABILITY OF DYNAMIC FAULT TREE BY

A. Equal Markov Model

Generally, a given fault tree can be easily concerted to its equivalent Markov model by replacing each gate with its Markov model based on the tree topology. In the next section we will show that the Markov model of each gate (of tree) can be reduced to a two states model shown in Figure 1.

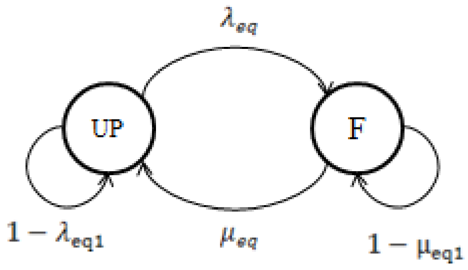

Figure 1. Equal markov model

To obtain the equal rates, the following method is used [2]:

$$\lambda_{eq} = \sum_{t_{i,j} \in R} \Pr\{S_i \mid U\} q_{i,j} = \frac{\sum_{t_{i,j} \in R} P_i . q_{i,j} (r_i - r_j)}{P_U} \tag{1}$$

$$P_U \equiv \sum_{S_k \in U} P_k \tag{2}$$

In the above equations, we have:

P_U : is steady state probability for k mode.

S: set of all states.

S_i : state i.

U: set of active mode.

$t_{i,j}$: transition from state i to j.

$q_{i,j}$: element of the infinitesimal generator matrix of Q which it is a diagonal matrix.

$$\mu_{eq} = \sum_{t_{i,j} \in G} \Pr\{S_i \mid D\} q_{i,j} = \frac{\sum_{t_{i,j} \in G} P_i . q_{i,j} (r_i - r_j)}{P_D} \tag{3}$$

$$P_D \equiv \sum_{S_k \in D} P_k \tag{4}$$

where:
G: set of transition from failure modes to active modes.
D: set of failure mode.
r_i : reward rate of state i.

If the state is active, so reward rate is zero, else it is unit. Since the difference between reward rates in equations 1 and 4 is unit, they do not influence, so are neglected.

Thus, according to the said method, firstly Markov model is simplified then steady state availability is achieved. So, in the next section, we briefly review the methods to calculate the steady state availability.

B. Evaluate of steady state availability of systems

In this section, the method for evaluation of steady state availability is accessed. After drawing a Markov model, the state transitions matrix to obtain, so we write the following equation:

$$\dot{P}_i(t) = QP_i(t) \tag{5}$$

where $i = 1,\dots,n$ and n is the states numbers, $P_i(t)$ is the failure probability for state i .
Also, we know the probability in the steady state is zero e.:

$$\dot{P}_i(\infty) = 0, i = 1,\dots,n \tag{6}$$

We know the sum of the probabilities is unit, so:

$$P_1(\infty) + \dots + P_n(\infty) = 1 \tag{7}$$

By using the equations (5)-(7), availability can be written as the following equation:

$$A(\infty) = P_1(\infty) + \dots + P_{n-1}(\infty) = 1 - P_F(\infty) \tag{8}$$

In the next section we introduce the dynamic gates.

III. THE INTRODUCTION OF TWO DYNAMIC GATES AND THEIR MARKOV MODEL

In this section, we introduce of PAND and Spare gate and then access Markov model.

A. PAND Gate

The structure of PAND is shown in Figure 2.

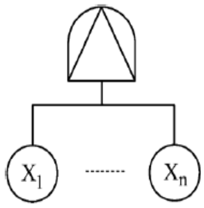

Figure 2. PAND dynamic gates

PAND is a Priority AND gate. In this gate, the first input has priority to the others. In fact, if the primary module is failed, the gate is failed; else the system can be operated.
To learn how to calculate availability of this gate, a gate with two inputs is considered. The first input has propriety to the second one. Markov model of the gate is given in Figure 3:

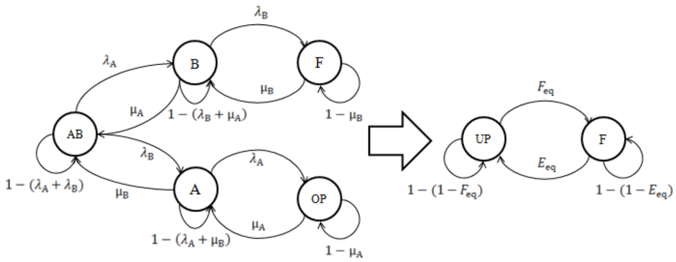

Figure 3. Markov model of PAND dynamic gates and its equal Markov Model

Therefore, the availability of the gate PAND is as (9):

$$A = \frac{(\lambda_A + \mu_A)(\lambda_B + \mu_B)}{2\lambda_A\lambda_B + \lambda_A\mu_B + \lambda_B\mu_A + \mu_A\mu_B} \tag{9}$$

B. Spare gate

The structure of this gate is shown in Figure 4:

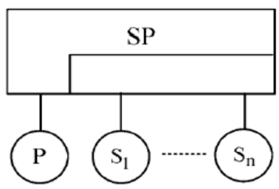

Figure 4. SP dynamic gates

The Gate has a primary input, and n spare inputs. When the primary input is failed, spares inputs are replaced. Spares operates in cold, warm, or hot inputs

To learn how to calculate availability of spare gate, we consider a gate with two inputs. First input is active, and if it fails, the second input is replaced. The Markov model of this gate is shown in Figure 5.

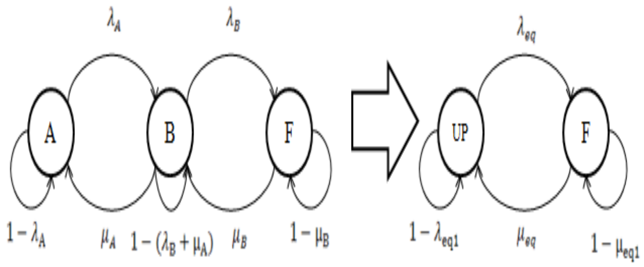

Figure 5. Markov model of SP dynamic gates and its Equal

According to the method mentioned in the previous section for availability evaluation, system availability is calculated as follows:

$$A = \frac{\lambda_A \mu_B + \lambda_B \mu_A + \mu_A \mu_B}{\lambda_A \lambda_B + \lambda_A \mu_B + \lambda_B \mu_A + \mu_A \mu_B} \tag{10}$$

IV. USING THE TEMPLATE EXAMPLES OF DYNAMIC FAULT TREE

A couple of examples are demonstrated in the next section to show the effectiveness of our methods.

A. *PAND gate supported by a cold spare*

The first example is tree with two dynamic gates and tree basic events (Figure 6).

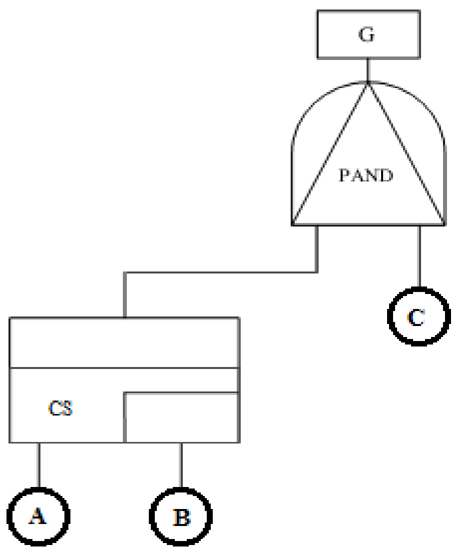

Figure 6. PAND gate supported by cold spare

To solve this tree, the following assumptions are made:

1. All components perform correctly.
2. The failure rate of each component obeys exponential distribution function.
3. At any time, only one component fails.
4. The switching mechanism of spare gate is ideal.
5. Repair is possible.

The original Markov model of this tree is shown in Figure 7. When it is solved the availability is obtained as follows (11).

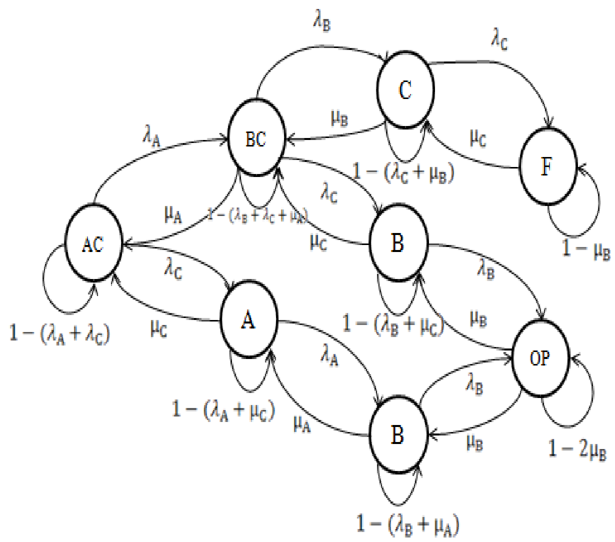

Figure 7. Markov model of PAND gate supported by cold spare

$$A = \frac{\lambda_A\lambda_B\lambda_C + \lambda_A\lambda_B\mu_C + 2\lambda_A\lambda_C\mu_B + \lambda_A\mu_B\mu_C + \lambda_C\mu_A\mu_B + \mu_A\mu_B\mu_C}{2\lambda_A\lambda_B\lambda_C + \lambda_A\lambda_B\mu_C + 2\lambda_A\lambda_C\mu_B + \lambda_A\mu_B\mu_C + \lambda_C\mu_A\mu_B + \mu_A\mu_B\mu_C} \tag{11}$$

The equal Markov model of cold spare gate has already been shown in Figure 8. For which λ_{eq}, μ_{eq} can be obtained from equations 1 to 4, as follows

$$\lambda_{eq} = \frac{\lambda_A\lambda_B\mu_B}{\left(\dfrac{\lambda_A\lambda_B}{\lambda_A\lambda_B + \lambda_A\mu_B + \mu_A\mu_B} - 1\right)(\lambda_A\lambda_B + \lambda_A\mu_B + \mu_A\mu_B)} \tag{12}$$

$$\mu_{eq} = \mu_B \tag{13}$$

According to our methodology simplified Markov model is shown in Figure 8.

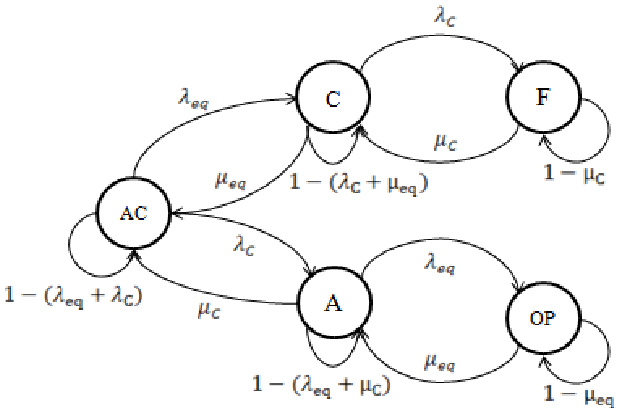

Figure 8. Fig. 8. Simplified Markov model

We now solved numerically the original tree and simplified tree to show correctness our methodology. Assumed that:

$\lambda_A = 0.5, \mu_A = 0.006$

$\lambda_B = 0.7, \mu_B = 0.007$

$\lambda_C = 0.8, \mu_C = 0.008$

$\lambda_C = 0.9, \mu_C = 0.009$

According to Figure 8, equal failure rate and equal repair rate are obtained as follows:

$\lambda_{eq} = 0.691, \mu_{eq} = 0.007$

After calculating the equal model of this tree, simplified model is obtained directly.
The diagram of comparing steady state availability of system by using main Markov model and proposed our method is shown in Figure 9.

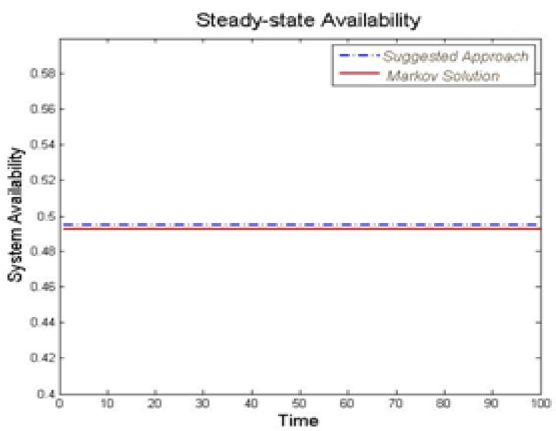

Figure 9. Compare PAND gate availability support with cold spare using the original model and our proposed method

According to the results of availability in Figure 9, we observe our method has a good approximation.

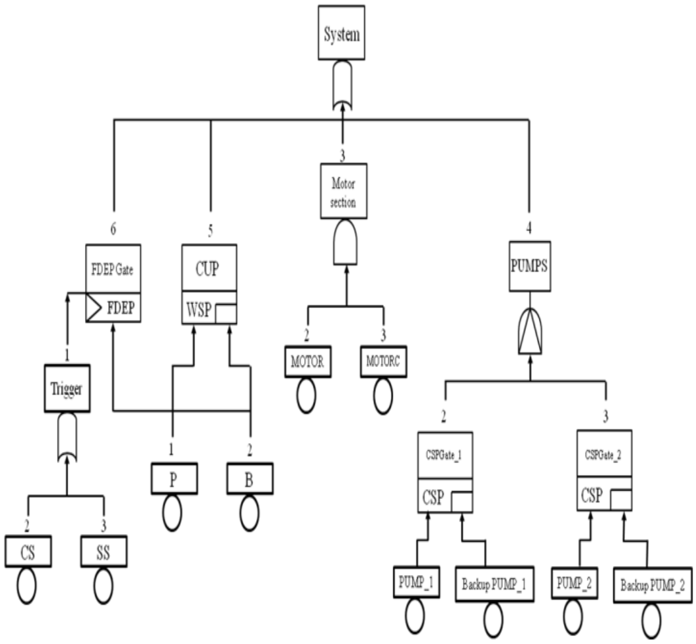

Figure 10. Fault tree of cardiac assistant device [27].

B. Cardiac assistant device

This device is design for the treatment of mechanical and electrical heart faults. Figure 10 shows the fault trees diagram of this system. The system consists of four modules: Trigger, CPU unit, motor section, and pumps. Trigger section conclude of a switch (CS) and a monitoring system (SS). CS failure or SS failure cause the failure of two processors. CPU unit is warm spare that includes a basic unit (P) and a spare that have a rate is equal 0.5. For motor section, one of the MOTOR or MOTORC must perform. The pumps are composed of two cold spare units; each consists of a main pump and a spare pump. For failure of pumps, all of the pumps must be failed and also, before fail of CSP2, CSP1 must be failed. The following table shows the values of failure and repair rates [27].

TABLE I. PARAMETERS OF CARDIAC ASSISTANT DEVICE

Basic Component	Failure Rate (10e-6)
CS	1
SS	2
P, B	4
P1,P2, BP	5
MOTOR	5
MORORC	1

The equal Markov model of PAND gate has already been shown in Figure 11.

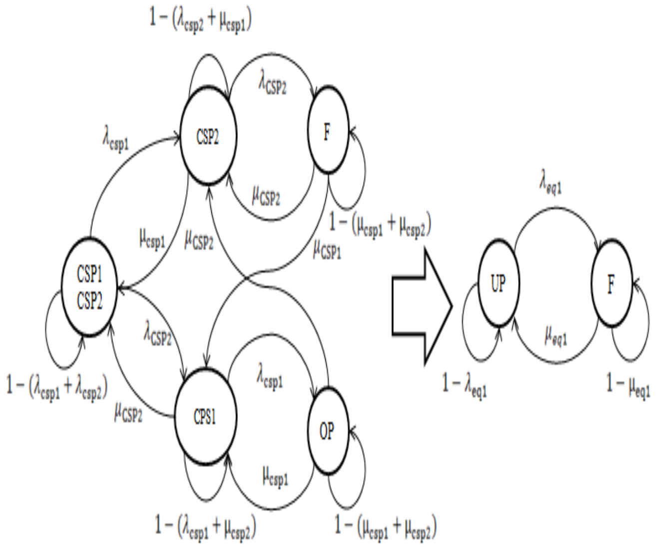

Figure 11. Markov model of the PAND gates in pumps section

λ_{eq1}, μ_{eq1} can be obtained from equations 1 to 4:

$$\lambda_{eq1} = \frac{\lambda_{CSP1}\lambda_{CSP2}\mu_{CSP2}}{\left(\frac{\lambda_{CSP1}\lambda_{CSP2}\mu_{CSP2}}{(\lambda_{CSP1}+\mu_{CSP1})(\lambda_{CSP2}+\mu_{CSP2})(\mu_{CSP1}+\mu_{CSP2})}-1\right)(\lambda_{AU}+\mu_{AU})(\lambda_{B/A}+\mu_{B/A})} \quad (14)$$

$$\mu_{eq1} = \mu_{CSP1} + \mu_{CSP2} \quad (15)$$

Now the motor section consist of a gate AND, and a motor. The failure rate and repair rate are calculated. Because the function of CPU depends to trigger module, so we use same Markov model for analyzing (Figure 12).

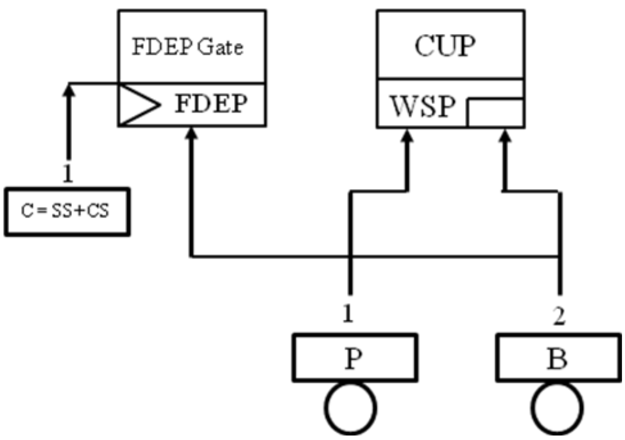

Figure 12. Fault tree of trigger and CPU section

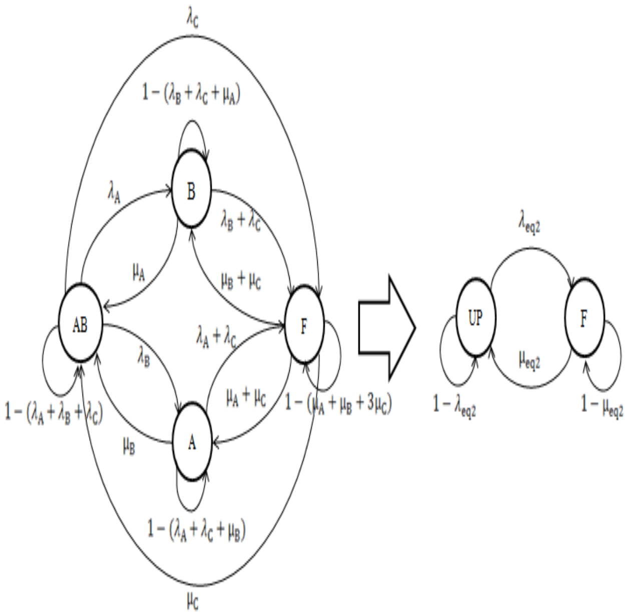

Figure 13. Equal markov model of the PAND gates in pumps section

Equal repair rates are calculated:

$$\lambda_{eq2} = 8.4e-006 \quad (16)$$

$$\mu_{eq2} = \mu_A + \mu_B + 3\mu_C = 1.7e-007 \quad (17)$$

After calculating all the sub sections of the model, the availability of the results of the previous system is obtained with OR of all previous sections (Figure 14).

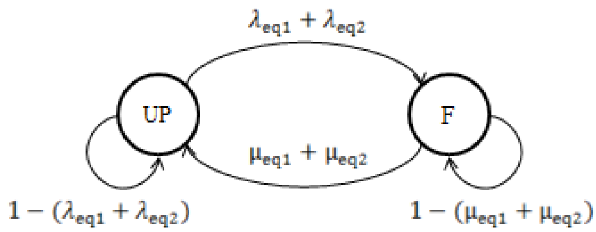

Figure 14. Final Markov model

So, availability is:

$$A = \frac{\mu_{eq1} + \mu_{eq2}}{\lambda_{eq1} + \lambda_{eq2} + \mu_{eq1} + \mu_{eq2}} \quad (20)$$

By comparing the original model and the availability of simplified our method in Figure 15, we can see the effectiveness of the method.

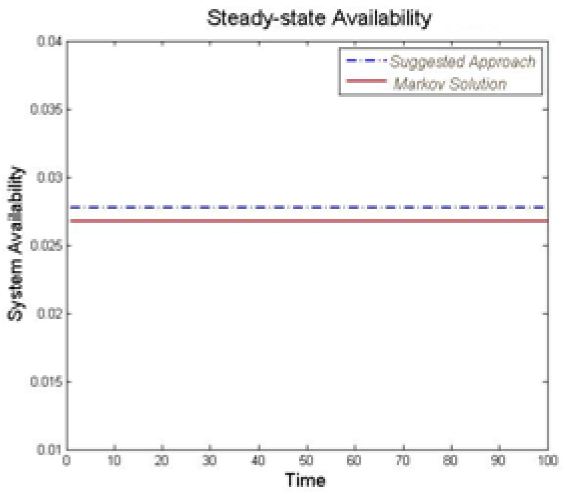

Figure 15. Compare cardiac assistant device the original model and our proposed method

V. CONCLUSION

This paper introduced a methodology for solving easily DFTs by the use of idea taken from reference [2]. Our purpose in this paper is the evaluation of availability. The Markov model of a given DFT is converted to a simplified model by replacing its constituent gate (Markov) models with their equal Markov models. In this way, the size of original model was decreased. We showed the correctness of our methodology with two examples.

REFERENCES

[1] R. A. Sahner, K. S. Trivedi, and A. Puliafito, “Performance and reliability analysis of computer systems: An example-based approach using the Sharpe software package,” Kluwer Academic Publishers, 1996.

[2] M. Lanu, L. Yin, and K. S. Trivedi, “Hierarchical composition and aggregation of state-based availability and performability models,” IEEE Transactions on Reliability, vol. 52, no. 1, 2003.

[3] M. Malhotra and K. S. Trivedi, “A methodology for formal specification of hierarchy in model solution,” in Proc. Fifth Int. Workshop Petri Nets and Performance Models, pp. 258–267, 1993.

[4] J. Hao, L. Zhang, and L. Wei," “Reliability analysis based on improved dynamic fault tree,” Petroleum, pp. 283-300, 2014.

[5] J. B. Dugan, S. J. Bavuso, and M. Boyd, “Dynamic fault-tree models for fault-tolerant computer systems,” IEEE Transactions on Reliability, vol. 41, no. 3, pp. 363–377, 1992.

[6] W. E. Veseley, F. F. Goldberg, N. H. Roberts, and D. F. Haasl, “Fault tree handbook, NUREG-0492,” Technical Report, NASA, 1981.

[7] Y. Mo, “A multiple-valued decision-diagram-based approach to solve dynamic fault trees,” IEEE Transactions on Reliability, vol. 63, no. 1, pp. 81-93, 2014.

[8] Manno G, Chiacchio F, Compagno L, D'Urso D, Trapani N., Mat Carlo Re, "an integrated FT and Monte Carlo Simulink tool for the reliability assessment of dynamic fault tree." Expert Systems with Applications, vol. 39, no. 12, p.p. 10334–42, 2012.

[9] Chiacchio F, Compagno L, D'UrsoD, Manno G, Trapani N., "An open-source application to model and solve dynamic fault tree of real industrial systems," Proceedings of the SKIMA, Benevento, 8–11, 2011.

[10] Manno G, Chiacchio F, D'UrsoD, Trapani N, Compagno L. RAATSS, "An extensible Matlabs toolbox for the evaluation of repairable dynamic fault trees. In," 11th international probabilistic safety assessment and management conference and the annual European safety and reliability conference, p.p. 1055–1064, 2012.

[11] Montani S, Portinale, L, Bobbio, A, Codetta-Raiteri, D., " Automatically translating dynamic fault trees into dynamic Bayesian networks by means of a software tool," Proceedings of the 1st international conference on availability, reliability and security, 2006.

[12] Hillston JA. Compositional approach to performance modelling. Cambridge University Press; 1996.

[13] Bouissou M, Lefebvr e Y. A, "path-based algorithm to evaluate asymptotic unavailability for large Markov models," Proceedings of the 48th reliability and maintainability annual symposium. p.p. 32–39,2002.

[14] Boudali H, Dugan J B., "A new Bayesian network approach to solve dynamic fault trees," Proceedings of the Annual Reliability and Maintainability Symposium, p.p. 451-456, 2005.

[15] Codetta Raiteri D. T, "The conversion of dynamic fault trees to stochastic Petri nets, a case of graph trans formation," Electronic Notes in Theoretical Computer Science 2004;vol. 127, no. 2, p.p. 45–60, 2004.

[16] J. B. Dugan, S. J. Bavuso, and M. A. Boyd., "Dynamic Fault-Tree Models for Fault-Tolerant Computer Systems," IEEE Transactions on Reliability, vol. 41, p.p. 363.377, 1992.

[17] K. Trivedi., "Probability & Statistics with Reliability," Queueing & Computer Science applications. Wiley, II Edition, 2001.

[18] R. Manian, D.W. Coppit, K. J. Sullivan, and J. B. Dugan., "Bridging the Gap between Systems and Dynamic Fault Tree Models," Proceedings Annual Reliability and Maintainability Symposium, p.p. 105.111, 1999.

[19] R. Gulati and J. B. Dugan. "A modular approach for analyzing static and dynamic fault-trees" Proceedings IEEE Annual Reliability and Maintainability Symposium, p.p. 57.63, 2003.

[20] D. Codetta-Raiteri, "The conversion of dynamic fault trees to stochastic petri nets, as a case of graph transformation," Proc. Workshop on Petri Nets and Graph Transformations, vol. 127, p.p. 45–60, 2005.

[21] G. Chiola, C. Dutheillet, G. Franceschinis, S. Haddad," Stochastic well-formed colored nets and symmetric modeling applications," IEEE Trans. Comput. vol. 42 p.p. 1343–1360, 1993.

[22] F. Arnold, A. Belinfante, D.G. Freark van der Berg, M. Stoelinga, DFTCalc, "A tool for efficient fault tree analysis," Proc. 32nd Int. Conf. Computer Safety, Reliability and Security, p.p. 293–301, 2013.

[23] Yevkin, O, “An improved monte carlo method in fault tree analysis,” Reliability and Maintainability Symposium (RAMS), 2010 Proceedings - Annual, pp. 1 – 5, 2010.

[24] R.N.Allan, R. Billinton, Reliability Evaluation of Power Systems, Plenum press. New York and london, 1996.

[25] K. Aslansefat, G. Latif-shabgahi "A Systematic Approach to Sensitivity Analysis of Fault Tolerant Systems in NMR Architecture", Journal of Intelligent Procedures in Electrical Technology (JIPET), vol. 20, no. 5, p.p. 3-14, 2014.

[26] "K. Aslansefat1, G. Latif-Shabgahi1, M. Kamarlouei, " A Strategy for Reliability Evaluation and Fault Diagnosis of Autonomous Underwater Gliding Robot based on its Fault Tree," nternational Journal of Advances in Science, Engineering and Technology (IJASEAT), vol. 2, no. 4, p.p. 83-89, 2014.

[27] K. Durga Raoa, V. Gopikaa, V. V. S. Sanyasi Raoa, H.S. Kushwahaa, A.K. Vermab, A. Srividyab, “Dynamic fault tree analysis using Monte Carlo simulation in probabilistic safety assessment,” Reliability Engineering & System Safety, vol. 94, no. 4, pp. 872–883, 2009.