Available online at www.sciencedirect.com

ScienceDirect

Procedia Technology 22 (2016) 623 – 628

Procedia Technology

9th International Conference Interdisciplinarity in Engineering, INTER-ENG 2015, 8-9 October 2015, Tirgu-Mures, Romania

Fault Tolerant Control Multiprocessor Systems Modelling Using Advanced Stochastic Petri Nets

Mariana Dumitrescu[a,]*

University "Dunarea de Jos" of Galati, Automation and Electrical Engineering Department, 111 Strada Domneasca, RO 800201 Galati, Romania

Abstract

Power systems need a safe management and power control. This means they are fault tolerant systems and specially need a strong safety of the processors, which have one of the most important tasks of the control. The paper presents how a processor can be modeled, what types of failures can appear in the functional state and the availability advanced modeling using discreet stochastic events models. The advanced Stochastic Petri Nets model proposed and used in this paper has a strong modeling power and this will be proved in comparison with classical availability models.

© 2016 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).
Peer-review under responsibility of the "Petru Maior" University of Tirgu Mures, Faculty of Engineering

Keywords: Distributed power systems; Control; Stochastic Petri Nets; Processor Availability; Dependability.

1. Introduction

Renewable systems power management and control is an application requiring high availability. These types of systems need modelling and computing of the various dependability measures, so many model types have been developed.

Is well known dependability is a measure of quality, correctness and continuity of service delivered by a system. Dependability encompasses measures such as reliability, availability and risk. The dependability models are different from one another taking in account their modelling power. Modelling power is very much determined by the types of dependencies within subsystems that can be computed. For instance, if various components of a system

* Mariana Dumitrescu. Tel.: +4-074-202-0132; fax: +4-074-202-0132.
E-mail address: mariana.dumitrescu@ugal.ro

2212-0173 © 2016 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).
Peer-review under responsibility of the "Petru Maior" University of Tirgu Mures, Faculty of Engineering
doi:10.1016/j.protcy.2016.01.129

share a repair dependency among components, than Stochastic Petri Nets (SPN) can easily model such a repair dependency [1,2,3].

The usual dependability model types, according with Malhotra and Trivedi [4], Puliafito et al. [5], are the following: reliability block diagrams (RBD); fault trees without repeated events (FT); fault trees with repeated events (FTRE); reliability graphs (RG); continuous-time Markov chains (CTMC); generalized stochastic Petri nets (GSPN) and the proposed simplified SPN model, Logical Explicit Stochastic Petri Nets (LESPN).

Malhotra and Trivedi [4] has been shown that a RBD is equivalent to a FT. In Trivedi [3] has been presented the algorithm to convert the RBD model type into FT model type and the counter-assertion. The FTRE possesses higher modelling power than FT or RBD because any RBD or FT model can also be modelled by a FTRE. Any FT can be converted to a RG. In the paper Malhotra and Trivedi [4] is presented this conversion algorithm, but also that not every RG can be converted to an equivalent FT. Since FT is equivalent to RBD, this also proves that not every RG can be converted to an equivalent RBD. Any RG can be converted to an equivalent FTRE, so in Malhotra and Trivedi [4] is presented the converting algorithm, but also that the FTRE to RG converse is not true.

Markov models can handle some dependencies in a system, which combinatorial models cannot, according with Fricks and Trivedi [6], Murata [7]. Repair-dependency cannot be modeled by any combinatorial model type, but a CTMC and a GSPN model can easily model such dependency. Trivedi [3] has been shown that CTMC model is equivalent to GSPN model. For every GSPN model, an equivalent CTMC exists and vice-versa. Malhotra and Trivedi [4] present the GSPN to CTMC conversion algorithm and also the CTMC-GSPN conversion algorithm. The overall hierarchy of dependability model types is shown in Fig. 1.

Section 2 presents the fault-tolerant multiprocessor system (FTMS) and its GSPN dependability-model type. In section 3 we present the algorithm to develop the model -Logical Explicit Stochastic Petri Nets (LESPN)- for the FTMS. Section 4 presents the comparative study of the LESPN model and GSPN model for FTMS.

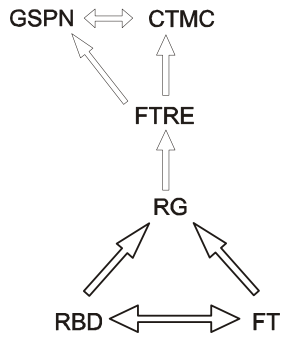

Fig. 1. Power hierarchy among the dependability model types.

2. GSPN modeling FTMS dependability

Fig. 2 shows the fault-tolerant analyzed multiprocessor system (FTMS) having a shared memory M3. The multiprocessor construction has two processors P1 and P2, each with a private memory M1 and M2 respectively. A processing unit consists of a processor and its memory and both processing units are connected to a mirrored-disk system. They are connected via an interconnection network N.

The proposed system is functional while N is functional and at least one of the processing subsystems is functional too. For a processing subsystem to be functional, the processor, the memory module and at least one of the two disks must be functional.

In the fault tolerant systems dependability modelling, have to be represented random variables as functional time, repair time. A SPN is able to associate a time random variable to the timed transition and also an exponential low for the random variable „time of the transition execution". In Malhotra and Trivedi [4] for availability computing of the multiprocessor has been proposed the generalized stochastic Petri net model, which allows using timed stochastic transitions and also immediate transitions (no-timed). According with Puliafito et al [5], GSPN has two types of markings: vanishing -having at least one validated immediate transition- and stable having no validated immediate transition.

Fig. 3a shows the GSPN model of the non- repairable FTMS with shared memory. The fault events for system elements are modelled using RP1 subnets in Fig. 3b. The logical conditions of multiprocessor failure are modelled using RP2, RP3 subnets presented in Fig. 3c,d. The RP1 subnets use timed stochastic transitions and build the „events modelling subnet" (EMS). The RP2, RP3 subnets, which are modelling the (C_1, C_2 ...C_n) logical conditions leading to the system failure D(x), use immediate transitions and build the „logical subnet" (LS). The RP2 subnet (AND type condition) is modelling the system failure, if all the (C_1,C_2,....C_n) conditions are true. The RP3 subnet (OR type condition) is modelling the system failure, if one of the (C_1,C_2,...,C_n) conditions is true.

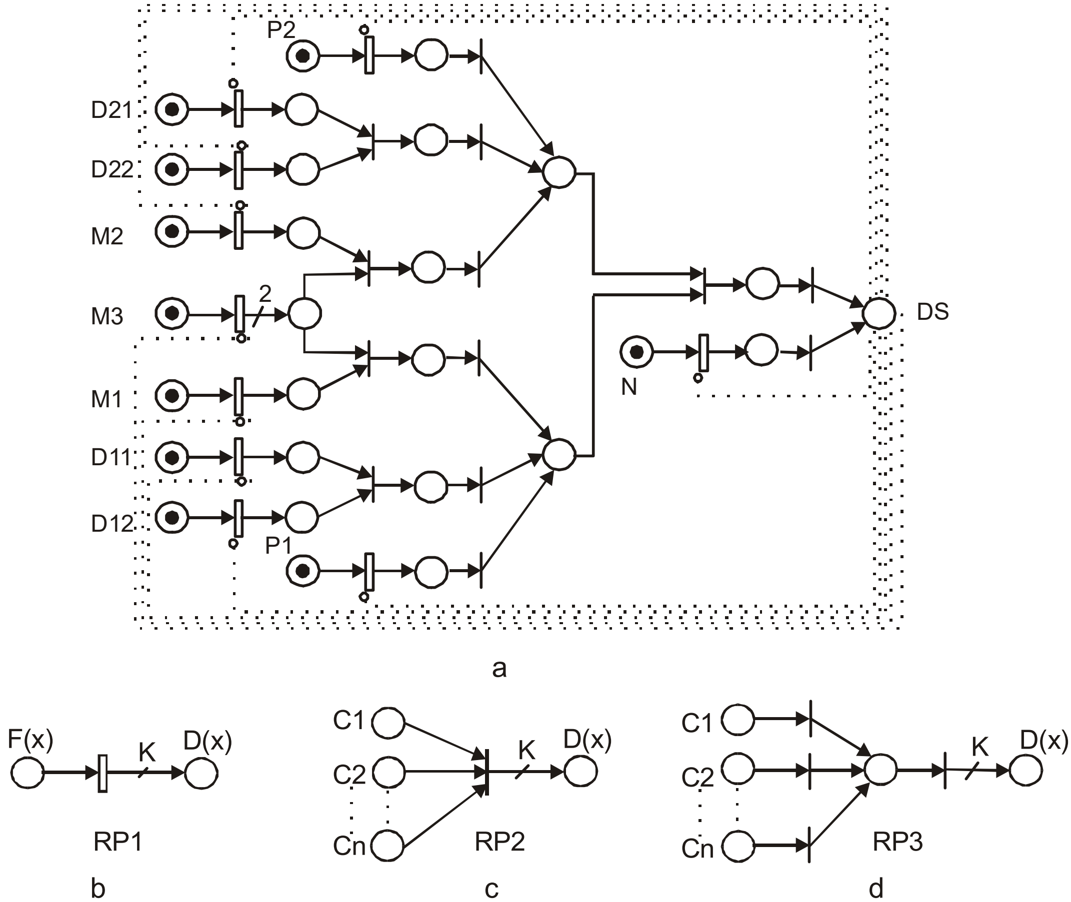

Fig. 2. The GSPN model for the FTMS (a), events modeling subnet (b), AND logical subnet (c), OR logical subnet (d).

In Malhotra and Trivedi [4] the GSPN is presented and it conducts to the multiprocessor failure, happening with the apparition of a token in DS place of the GSPN model, due to some of the system elements faults. At this time the inhibitor arcs of the GSPN are blocking the execution of the validated stochastic transitions, as Zijal explains in [10].

3. LESPN model and FTRE-LESPN conversion algorithm

The GSPN model type, although is situated on the top of power- modelling hierarchy, presents some important limits because of the logical subnet (LS), used in operational dependency modelling. Also the very large size of the

GSPN model appears because of modelling the logical system performance conditions, as Fricks and Trivedi in [6] concluded. This conducts to a very large GSPN model, even for simple systems, making very difficult to represent the GSPN model for a repairable complex system, according with Navet in [8, 9] and Zijal in [10].

The proposed simplified model LESPN, which is explained in Dumitrescu [2], used in this paper for the multiprocessor analyzed system, does not use the LS for the dependability modelling. The logical performance conditions of the system are modelled outside the SPN, in a SPN associate table. The SPN structure uses only the events modelling subnet (EMS).

Comparing to the GSPN model, the LESPN model, has the following advantages:

- it uses a higher level PN, with colours, with predicates/transition, giving a very easy to use and intuitive structure;
- its dimensions are reduced, giving a simplified dependability modellling;
- it uses only the EMS modular architecture;
- the system logical conditions are modelled outside the SPN, in the logical table called "PERFORMANCE";
- the operational dependencies between the system components, associate to EMS different modules, are modelled by the arcs set and the predicates/transition set;
- the system behaviour does not possess the vanishing markings created by the LS of the GSPN model;
- it is easy to extend the model from non repairable system to the repairable system, only adding the transitions for the repair events of the components.

If a converting algorithm from FTRE (Fig. 3) to LESPN model will be presented, this conducts to the conclusion that LESPN model is more powerful than FTRE model. Also it is obvious clear the equivalence between the LESPN model and GSPN, CTMC models.

FTRE dependability model for FTMS, presented in Fig. 3, is composed of: fourth level gates AND(1), AND(2), AND(3), AND(4), third level gates OR(5), OR(6), second level gate AND (7) and fist level gate OR(8). The proposed algorithm built LESPN subnets beginning with the biggest level gates. We build the LESPN subnets for the AND(1), AND(2), AND(3), AND(4) gates, presented in Fig. 4.

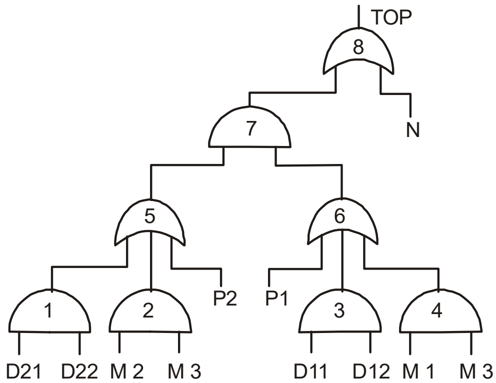

Fig. 3. FTRE model for FTMS.

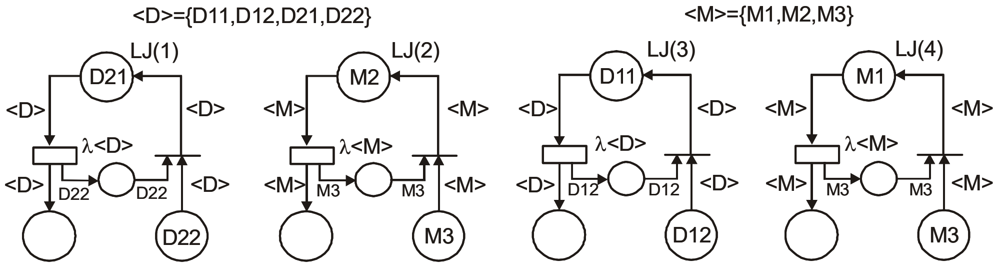

Fig. 4. LESPN subnets for AND(1), AND(2), AND(3), AND(4) gates.

The next step of the algorithm elaborate LESPN subnets for third level OR(5), OR(6) gates (Fig. 5) which are connected with interrupted arcs to the subnets of fourth level gates. Than in the following step we elaborate the LESPN subnet for the second level AND(7) gate by reconstructing both the LJ(5) and LJ(6) places in the new LJ(7) place, which add all the tokens of LJ(5) and LJ(6) places and all their arcs (Fig. 6).

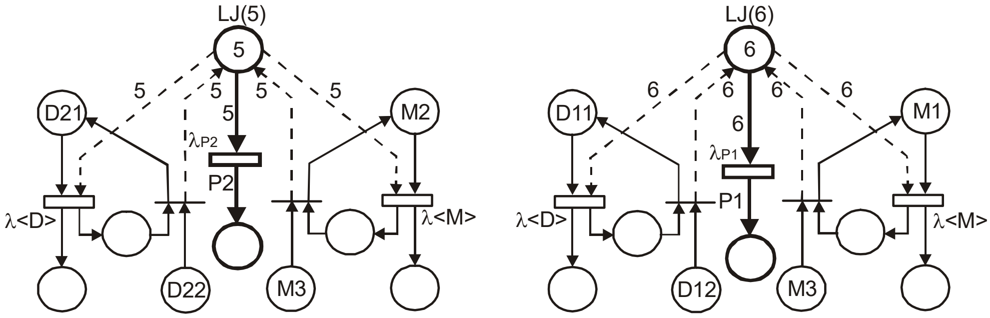

Fig. 5 LESPN subnets for OR(5), OR(6) gates.

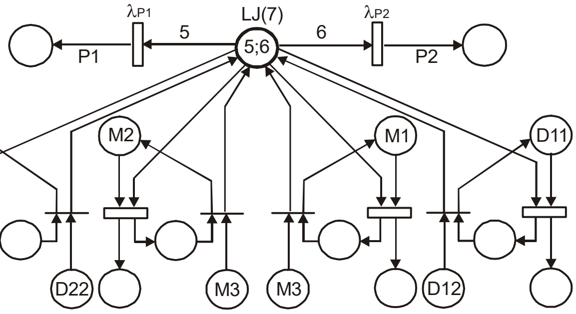

Fig. 6. The non-coloured model of LESPN subnet for AND(7) gate.

The next step of the algorithm is to colour the LESPN presented in Fig. 6 and to elaborate the predicates/transition sets added in Fig. 7. Then we built the LESPN subnet for the first level OR(8) gate, beginning with the N primary event of the gate. The LJ(8) place is connected with out-arcs to the stochastic transitions and with in-arcs to the immediate transitions of the LESPN four AND gates.

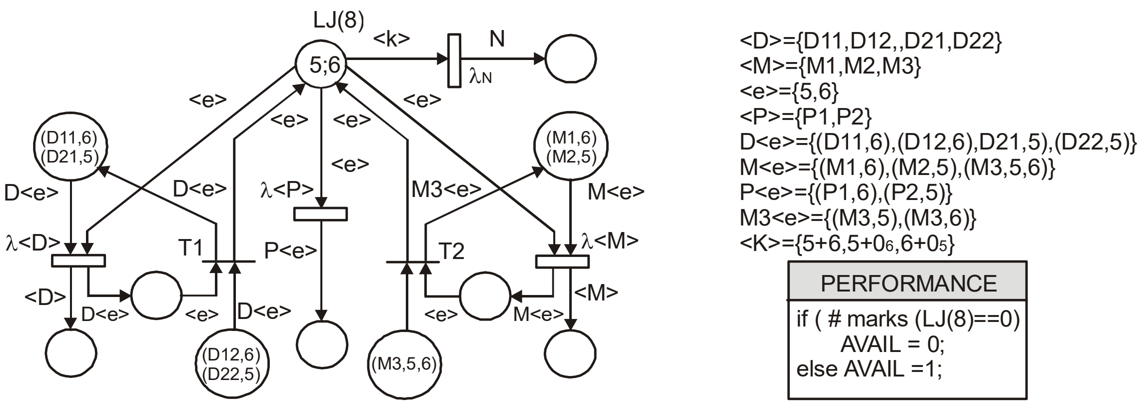

Fig. 7. Colored having predicates/transition LESPN model for FTMS.

The LESPN→FTRE transformation isn't always possible. In the complex case of the renewal system it's impossible to find the equivalent FTRE model. That means the LESPN model has a higher dependability modelling power than FTRE model.

4. Conclusions

The LESPN model for the FTMS has a modular architecture and uses high level SPN. It has a very intuitive, an easy to understand structure , which make it and easy to use. The immediate transitions T1 and T2 are modelling the redundancy of the D and M system components. The LJ tokens (5, 6) show that both processing units are operable. It is obviously clear that the fail state of the system appears when both LJ tokens are moving, then the system availability is affected (AVAIL=0).

The LESPN model for the FTMS with shared memory has a reduced number of places and transitions comparing with the GSPN model (Fig. 3). Also the system behavior is very correct modeled because the LESPN model eliminates the vanishing markings, created by the GSPN model. Because of the strong modeling power the processor tests, before implementing in the control systems, are more secure and the measurable practical results, meaning the availability of FTMS present a big level of trust.

References

[1] Ajmone-Marsan A et. al. A class of generalized stochastic Petri nets for the performance evaluation of multiprocessor systems. ACM Trans. Computer Systems, 1984, 2:93-122.

[2] Dumitrescu M. Contribution to the structure configuration and control of naval power system, on reliability criteria. Physics theses "Dunarea de Jos" University of Galati; 1997.

[3] Trivedi K. Probability and Statistics with Reliability, Queuing and Computer Science Applications - 2nd Edition. Wiley; 2002.

[4] Malhotra M, Trivedi KS. Dependability Modeling Using Petri-Net Based Models. IEEE Transactions on Reliability 1995, 44(3):428-440.

[5] Puliafito A, Telek M, Trivedi KS. The Evolution of Stochastic Petri Nets. Proc. World Congress on Systems Simulation (WCSS '97), (2nd Joint Conference of International Simulation Societies), Singapore, Sept. 1-3; 1997, p. 3-15.

[6] Fricks R, Trivedi KS. Modeling Failure Dependencies in Reliability Analysis Using Stochastic Petri Nets. Proc. European Simulation Multi-conference (ESM '97), Istanbul, Jun; 1997, p.85-99.

[7] Murata T. Petri Nets: Properties, Analysis and Applications. Proceedings of the IEEE, 1989, 77(4):541-580.

[8] Navet, N, Song Y, Simonot-Lion F. Worst-Case Deadline Failure Probability in Real-Time Applications Distributed over Controller Area Network, Journal of Systems Architecture 2000, 46(1):607-617.

[9] Navet N, Song Y, Simonot-Lion F, Wilwert C. Trends in Automotive Communication Systems, Proceedings of IEEE 2005, 93(6):1024-1223.

[10] Zijal R, Ciardo G, Hommel G. Discrete Deterministic and Stochastic Petri Nets, ICASE Technical Report 96-72, NASA Langley Research Center; 1996.