



HAL
open science

Assessment of the critical events sequences of systems by means of probabilistic languages

Dorina-Romina Ionescu, Nicolae Brinzei, Jean-François Pétin

► **To cite this version:**

Dorina-Romina Ionescu, Nicolae Brinzei, Jean-François Pétin. Assessment of the critical events sequences of systems by means of probabilistic languages. European Safety and Reliability Conference ESREL 2014, Sep 2014, Wroclaw, Poland. pp.217-224. hal-01083191

HAL Id: hal-01083191

<https://hal.science/hal-01083191>

Submitted on 10 Dec 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Assessment of the critical events sequences of systems by means of probabilistic languages

D. Ionescu^{1,2}, N. Brînzei^{1,2} & J-F. Pétin^{1,2}

¹Université de Lorraine, CRAN, UMR 7039

2, avenue de la forêt de Haye, Vandoeuvre-lès-Nancy, Cedex, 54518, France

²CNRS, CRAN, UMR 7039

2, avenue de la forêt de Haye, Vandoeuvre-lès-Nancy, Cedex, 54518, France

ABSTRACT: In dependability studies of dynamic systems it is important to assess the probability of occurrence for the events sequences which describe the system evolution or which are critical for the mission of the system or for the humans and environment safety. In this paper we use the probabilistic languages framework in order to realize the quantitative assessment and we start by modeling the system as a finite state automaton. This is ulterior transformed in a probabilistic automaton using the embedded discrete time Markov chain. The determination of the languages afferents at each state of the automaton enable to calculate the probability of occurrence for every events sequence that can be subtract from these languages.

1 INTRODUCTION

Assessing the dependability of a critical system generally relies on Boolean models (Fault Trees models, Reliability block diagrams, etc.) for:

- qualitative analysis, such as the determination of the sets of component's failures that cause a critical or undesired system situation (it is defined as the *cut* set),
- quantitative analysis to evaluate the probabilities of undesired events or RAMS indicators, that can be based on the Sylvester-Poincaré Theorem or Binary Decision Diagrams technique (Dutuit & Rauzy 2005, Ibanez-Llano et al. 2010).

This paper explores the use of probabilistic languages introduced by (Garg et al. 1999) as a formal framework for qualitative/quantitative analyze of the event sequences. Before to describe the *probabilistic languages* concept, we will present some reasons that motivate the choice of *events sequences* in dependability studies.

Evaluation using *cut* analysis requires a full independence of the events involved in the *cut* set. However, this assumption cannot be stated in the case of dynamic reparable systems due to the order relationship between events occurrence:

- occurrence of event may depend on previous occurrence of other events; *e.g.* in the dynamic controlled systems event occurrence may be prohibited by its control according to a previous occurrence, or not, of some other events,
- impact of event sequence on the system failure may be different according to the scheduling of the event occurrence within the sequence; *e.g.* event e_1

followed by event e_2 leads to an undesired event while e_2 followed by e_1 has no impact.

In these cases, the value of the obtained probability, based on *cut-set* calculus, represents an overstatement of the real probability that is searched, corresponding no longer with the actual needs of dependability studies.

Moreover, the *cut*-based analysis approaches fit well to the non-reparable systems. However, in dynamic reliability context, it is necessary to take in consideration the changes between the different modes of functioning/failure or when the systems turns back into a functioning state after having passed in a failure state. In this case the systems must be considered like reparable and only dynamic models are able to capture system reconfigurations.

For all these reasons, the *cut-set* based analysis needs to be enriched by the determination and assessment of event sequences within the *cut-set*. This evolution justifies the use of state space models to capture the system dynamic and the impact of component's failures and reparation on the system state. Some approaches have been recently developed to determine the critical sequences of events and some basic properties such as minimality and consistency has been proposed for dynamic reparable system (Bouissou & Bon 2003, Chaux et al. 2013). However, these approaches are based on deterministic language theory and focus on the identification of a set of events sequences but present some limitations and divergences for qualitative and quantitative assessment due to modeling and sequence calculus assumptions.

The paper is organized as follows. Section 2 presents the state of the art and highlights the current

limitations of existing approaches for sequences assessment. Section 3 introduces the theoretical framework of the probabilistic languages (Garg et al. 1999) and the used notations. In section 4 is introduced the case study which is used like a support for the rest of the paper. Section 5 develops a theoretical framework based on probabilistic languages for quantitative assessment of events sequences. This proposed approach is illustrated using the case study in the section 6. Finally, section 7 presents the conclusions of this work and identifies some future research directions.

2 STATE OF THE ART

The existing approaches in quantitative assessment area can be classified in two categories:

- *Boolean models* where the failure of the system is expressed using the Boolean structure function;
- *state space models* where the system behavior is represented by states and transitions and his evolution is described by the associated language.

2.1 Boolean models

The most important Boolean models are: Event Trees and Fault Trees (classics/dynamics). *Event Trees* (Papazoglou 1998) are graphical models that can be discretized according to their possible effects or distinction in a series of simple events. In the same time *Event Trees* are adapted for modeling and assessment of the events sequences for non-reparable systems such as safety or protection systems. Instead, the reparable systems or instrumentation and control systems can not be modeled by *Event Trees*. *Fault Trees* are built according to an undesired event that is decomposed into basic events till this decomposition becomes impossible or judged useless (Dutuit & Rauzy 2005). These models are efficient for *cut*-based analysis but have limitations for *sequence*-based analysis because of the static point of view they consider.

2.2 State space models

The most currently used state space models in the context dependability studies are (Cassandras & Lafortune 2008): Finite State Automata and Language theory, Petri Nets and Markov chains. Other models can also be mentioned in the context of this paper. *Boolean logic Driven Markov Process (BDMP)* developed by Bouissou & Bon (2003) replaces the basic events of a fault tree with Markov chains; this combination introduces dynamic features to cover the order and impact of events occurrence. Events sequences are determined by exploration of the model starting with the initial state of the system (Bouissou 2006). *Stochastic Hybrid Automa-*

ton is a stochastic temporized finite state automaton with embedded differential equations for each mode (Perez Castaneda et al. 2011); this model enables to determine sequences of events and their probabilities thanks to Monte-Carlo simulation, but it is not possible to obtain analytical solution.

All these models can be used, according to their deterministic or stochastic features for event sequences analysis (Bouissou & Bon 2003, Chaux et al. 2013) with two kinds of problem solving: *determination* and *probability assessment* of the event sequences.

For the determination problem, approaches based on the language theory are efficient to compute some event sequences leading to a given state. Calculus is possible only for coherent systems and helps to determine the minimal set of sequences aiming to reconstruct the whole failure language (Chaux et al. 2013). This approach does not allow quantitative assessment for the determined events sequences.

For the assessment of sequence probability, some limitations are encountered by the existing approaches:

- state space models such as Markov chains are efficient for determining a state probability. Consequently, a sequence needs to be assimilated to its final state that must be considered as an absorbing state. In this case, probability of this state represents the probability of the language reaching this state.
- the sequence probability calculus is based on the events occurrence probability; in existing approaches, the value of this probability remains the same whatever is the state from which the event occurs. However, in dynamic system, a given event may have different occurrence probability in different system states even if its meaning – failure or reparation of the component c_i – remains the same in all sequences.

Probabilistic Languages (Garg et al. 1999), that takes advantages from both languages theory and stochastic process theory, appears to be an interesting way towards a formal framework for the analysis of event critical sequences. The next section presents the theory foundation of the *probabilistic languages* concept.

3 PROBABILISTIC LANGUAGES FRAMEWORK

The theory of probabilistic languages was developed by Garg, Kumar & Marcus (1999) in order to model the stochastic discrete event systems (DES) behavior.

To simplify the formal definition of the probabilistic language, a special event called “termination event”, noted e_{Δ} , is used to represent the fact that the state of the system obtained after the occurrence of a

sequence is a terminal state. A system state is terminal if it represents a special interest for the study (e.g. system mission achievement, dangerous failure state). Thus the system behavior is given by the set Ω of finite length sequences followed or not by the termination event:

$$\Omega = \Sigma^*(e_\Delta + \varepsilon) = \Sigma^*e_\Delta \cup \Sigma^* \quad (1)$$

In this equation Σ is the set of all events called alphabet, Σ^* is the set of all finite events sequences of the alphabet Σ (* is the iteration operation called also Kleene-closure) and ε represents an empty sequence of events.

To determine the occurrence probability for an events sequence s , should be considered not only the independent occurrence of this sequence but also the occurrence of all sequences that starts with s (all the sequences which have s like prefix). The set of all the sequences having a given prefix sequence s is given by:

$$\langle s \rangle = \{st \mid st \in \Omega\} \quad (2)$$

Definition 1 (Garg et al. 1999): Consider the measurable space (Ω, F) , where $\Omega = \Sigma^*(e_\Delta + \varepsilon)$ and F is the σ -algebra generated by $\{\langle s \rangle \mid s \in \Omega\}$. Then a *probabilistic language* (p-language) \mathbb{P} is a probability measure on the measurable space (Ω, F) .

Thereby the *termination probability for a sequence* s (the probability that the evolution of a system ends after the occurrence of s) is given by the following expression:

$$\mathbb{P}(se_\Delta) = \mathbb{P}(s) - \sum_{e \in \Sigma} \mathbb{P}(se), \forall s \in \Sigma^* \quad (3)$$

where $\mathbb{P}(s)$ represents the occurrence probability of s and $\sum_{e \in \Sigma} \mathbb{P}(se)$ represents the probability of continuous operation beyond s .

The probabilities of termination corresponding to different sequences are mutually exclusive, so the cumulative probability that the system arrives in a terminal state can be obtained by adding the individual probabilities for all possible sequences:

$$\mathbb{P}(\text{Syst } e_\Delta) = \sum_{s \in \Sigma^*} \mathbb{P}(se_\Delta) \quad (4)$$

Sometimes it is easier to describe a *p-language* using an associated automaton that can recognize this language.

Definition 2: A probabilistic automaton (or p-automaton) over the alphabet (set of events) Σ is defined by the following quintuple:

$$A_p = (X, \Sigma, f, \mathbb{P}, x_0) \quad (5)$$

where:

- X is a finite set of states;
- Σ is a finite set of events called alphabet;
- $f: X \times \Sigma \rightarrow X$ is the transition function that associates to each initial state and each event an arrival state;

- $\mathbb{P}: X \times \Sigma \times X \rightarrow [0, 1]$ is the probability transition function affecting to each transition a discrete occurrence probability which satisfies the following relation:

$$\sum_{x_j \in X} \sum_{e \in \Sigma} \mathbb{P}(x_i, e, x_j) \leq 1, \forall x_i \in X \quad (6)$$

- x_0 represents the initial state.

The evolution of a p-automaton is the following: if the system is in the state x_i , the transition e to the state x_j is done with the probability $\mathbb{P}(x_i, e, x_j)$. The probability transition function can be extended to the paths $\pi \subset X(\Sigma X)^*$ in the p-automaton A_p (a path is obtaining by concatenating the transitions, where the end state and the initial state of two consecutive transitions coincide). The probability of the paths is defined by the next equation:

$$x_j \in X: \mathbb{P}(\pi e x_j) = \mathbb{P}(\pi) \mathbb{P}(x_{|\pi|} e x_j) \quad (7)$$

Each p-automaton defines a p-language and inversely each p-language can be represented by a p-automaton.

Definition 3 (Garg et al. 1999): The p-automaton $A_p = (X, \Sigma, f, \mathbb{P}, x_0)$ with:

$$\forall s, t \in \Sigma^*, e \in \Sigma: \mathbb{P}(s, e, t) = \begin{cases} \frac{L(t)}{L(s)}, & \text{if } t = se \\ 0, & \text{otherwise.} \end{cases} \quad (8)$$

generates a L p-language.

4 PRESENTATION OF THE CASE STUDY

In order to present the application of probabilistic languages we start by considering an oven temperature control system (Figure 1). It operates in the following manner: the oven temperature can be controlled by a proportional integral controller (PI) or by an On/Off controller.

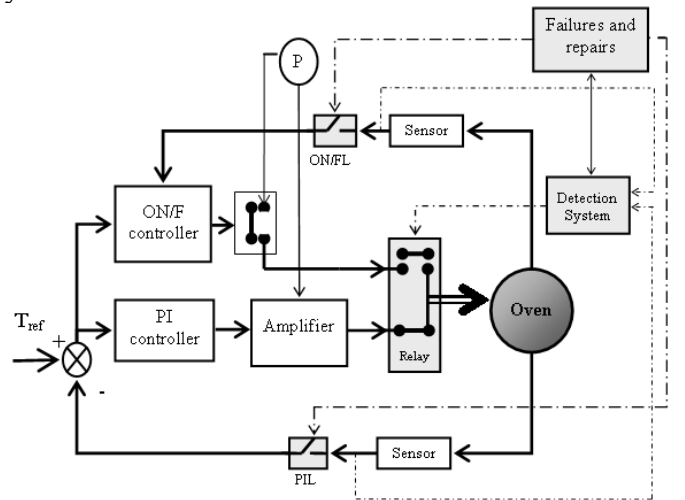


Figure 1. Structural diagram of the temperature control system of an oven.

The proportional integral controller (PI) has the role of keeping the temperature at a given target value and is characterized by a failure rate λ_{PI} and by a probability of failure on demand p_{refPI} . The role of On/Off controller is to maintain the temperature in a given range and is also characterized by a failure rate $\lambda_{On/Off}$ and by a probability of failure on demand $p_{refOn/Off}$. When the controller PI fails, the On/Off controller provides the temperature control if this one is not failing on demand. In the same manner if the On/Off controller fails the PI controller will provide the temperature control (if it not fails on demand).

When either of the controllers is failed it will be repaired, the repair process being characterized by a repair rate μ_{PI} respectively $\mu_{On/Off}$. This operating mode represents the simplified operating mode (obtained by removing the continuous part of the system describing the temperature evolution) of the case study defined in (Perez et al. 2011).

The values for the different rates that characterize the system transitions are:

$$\begin{aligned}\lambda_{PI} &= 3.5 \cdot 10^{-5} h^{-1}, \lambda_{On/Off} = 2 \cdot 10^{-5} h^{-1}, \\ \mu_{PI} &= 8 \cdot 10^{-2} h^{-1}, \mu_{On/Off} = 10 \cdot 10^{-2} h^{-1}, \\ p_{refPI} &= 0.03, p_{refOn/Off} = 0.05.\end{aligned}$$

The automaton that describes this system is presented in Figure 2.

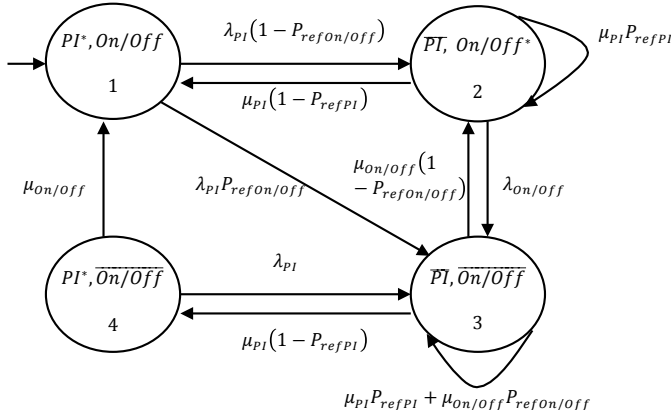


Figure 2. Continuous Time Markov Chain of the case study.

5 PROPOSED APPROACH

The proposed approach is decomposed into four steps.

5.1 Step 0: system modelling

The approach based on probabilistic languages theory proposed by Garg et al. needs to give a priori the p-language (the set of sequences and for each sequence s the occurrence probability of sequences sub set that have s like prefix). In other words, the p-language is built from equation (3). This knowledge (the whole set of sequences and their probabilities)

is not aware in the dependability studies. In dependability studies, the problem is rather reverse: giving a system, we need to determine the events sequences and, after that, to calculate their occurrence probabilities. Thus, we propose to work with, as initial data, the p-automaton.

5.2 Step 1: events probability determination

The p-automaton needs to be provided with event occurrence discrete probability for each transition (as in definition 2). As in dependability studies the probability laws describing the stochastic phenomena are rather continuous, we propose to determine discrete probabilities required by p-automata theory using embedded Discrete Time Markov Chain (DTMC) (Figure 3) in a continuous time stochastic process. This embedded DTMC is obtained by considering only the instants of transition firing between states. Using embedded DTMC ensures that the system reaches a stationary probabilities distribution of states. Consequently, the probability p_{ij} of a transition e_{ij} that starts from a given state s_i to another state s_j is determined as the ratio between the rate of this transition and the sum of the rates for all transitions starting from state s_i . This probability is given by the following relation:

$$\mathbb{P}(e_{ij}) = p_{ij} = \frac{\lambda_{ij}}{\sum_{j \neq i} (\lambda_{ij})} \quad (9)$$

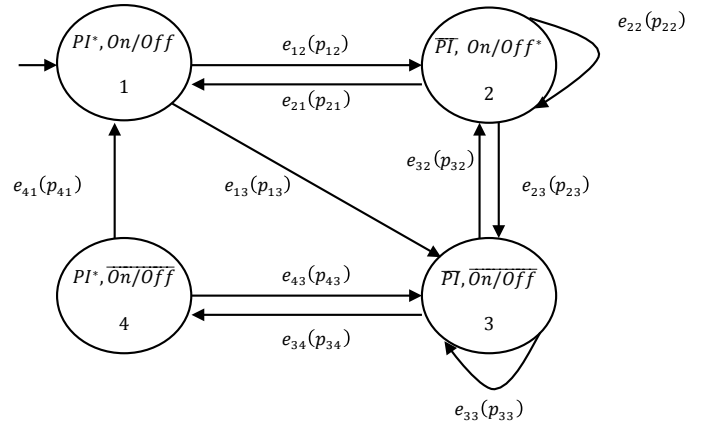


Figure 3. Discrete Time Markov Chain representing the p-automaton for the case study.

For example on the case study, the occurrence probabilities for the transitions e_{32} , e_{33} and e_{34} are:

$$\mathbb{P}(e_{32}) = p_{32} = \frac{\mu_{On/Off}(1 - p_{refOn/Off})}{\mu_{On/Off} + \mu_{PI}}$$

$$\mathbb{P}(e_{33}) = p_{33} = \frac{(\mu_{On/Off} p_{refOn/Off} + \mu_{PI} p_{refPI})}{\mu_{On/Off} + \mu_{PI}}$$

$$\mathbb{P}(e_{34}) = p_{34} = \frac{\mu_{PI}(1 - p_{refPI})}{\mu_{On/Off} + \mu_{PI}}$$

Note that the sum off all the transitions, which can start from a given state, is equal to one.

5.3 Step 2: *sublanguages determination*

After event probability determination, next step is about probability sequence calculus.

Considering the case study, we focus on the analysis of the sequences that reach the state 3 (dangerous state). Indeed, in this state, the system is completely failed and the temperature oven is uncontrolled.

To use the equation (3), which gives the expression of the probability of these sequences, state 3 must be considered as a terminal state. Thus for a sequence, which terminates in state 3, considering first that the occurrence probability of a sequence is equal with the product between transitions probabilities (equation 7) and also considering the events e_{32} , e_{33} and e_{34} , which allowed the system to continue his evolution beyond the state 3, the termination probability for the sequence s it is:

$$\begin{aligned} \mathbb{P}(se_{\Delta}) &= \mathbb{P}(s) - \sum \mathbb{P}(se) \\ &= \mathbb{P}(s) - (\mathbb{P}(se_{32}) + \mathbb{P}(se_{33}) + \mathbb{P}(se_{34})) \\ &= \mathbb{P}(s) - (\mathbb{P}(s)\mathbb{P}(e_{32}) + \mathbb{P}(s)\mathbb{P}(e_{33}) \\ &\quad + \mathbb{P}(s)\mathbb{P}(e_{34})) \\ &= \mathbb{P}(s) - \mathbb{P}(s)(\mathbb{P}(e_{32}) + \mathbb{P}(e_{33}) + \mathbb{P}(e_{34})) \\ &= \mathbb{P}(s) - \mathbb{P}(s)(p_{32} + p_{33} + p_{34}) \\ &= \mathbb{P}(s) - \mathbb{P}(s) \cdot 1 = 0 \end{aligned}$$

This approach is proposed in the p-languages theory and it was developed in the context of synthesis of control strategy by supervision (Wang & Ray 2004). The obtained result shows that, in dependability studies, the calculus of events sequences probability is not possible if this p-language is coupled with embedded DTMC without any cautions (if the terminal state of a sequence is not an absorbent state, its probability is always equal to zero).

To avoid this, we propose to determine firstly the whole set of events sequences of the system, that is represented by the language L_{sys} . This language can be represented as the union of sublanguages associated with the states of the system. A sublanguage associated to a state s_i is defined as the set of all events sequences, which lead the system from its initial state to the considered state.

$$L_{sys} = \bigcup_{s_i} L_i \quad (10)$$

To determine the sublanguages L_i we propose to use the theory of rational languages and more precisely the Arden lemma (Carton 2008).

Arden Lemma: Being two languages A and B and being the equation:

$$L_i = L_i A + B \quad (11)$$

where L_i represents the unknown language:

1. if $\varepsilon \notin A$, the only solution of the equation is $L_i = BA^*$.
2. if $\varepsilon \in A$, the solutions have the form $L_i = (B + C)A^*$ where $C \subseteq \Sigma^*$.

The lemma is mainly used in the case where $\varepsilon \notin A$ and the language $L_i = BA^*$ is the unique solution.

For each state s_i of the system the equation (11) can be written, considering the sequences starting from all the other states $s_j \neq s_i$ and arriving in the state s_i by only one transition. The set of n equations (11) (where n is the number of system states) allow to obtain all the sublanguages L_i .

Having a system represented by a p-automaton and using this approach we can determine all the events sequences without model exploring.

5.4 Step 3: *sequences probability calculation*

Each events sequence of the system $s = e_{12}e_{23} \dots e_{(n-1)n}$ that starts from the initial state s_1 and arrives in the state s_n can be extracted from the language L_{sys} obtained in the previous step. Using the probability for each transition of the p-automaton obtained in step 1, we propose to use the following equation in order to obtain the termination probability for the sequence $s = e_{12}e_{23} \dots e_{(n-1)n}$:

$$\mathbb{P}(se_{\Delta}) = \prod_{e_{ij} \in s} \mathbb{P}(e_{ij}), \forall s \in \Sigma^* \quad (12)$$

The meaning of this equation is that the probability of a events sequence is equal with the product of the probabilities of all of its events e_{ij} .

Figure 4 summerizes the different steps of our proposed approach to formally determine the whole set of events sequences and to assess their probability using the probabilistic languages.

6 APPLICATION ON THE CASE STUDY

6.1 *Application of the proposed approach*

Step 0: Figure 3 shows the embedded Discrete Time Markov Chain representing the p-automaton of the case study.

Step 1: The transition probabilities p_{ij} are calculated using the equation (9).

Step 2: The sublanguages associated to each state represent the sub set of the sequences that lead the system from the initial state 1 to one of his states. Considering each state of the automaton as terminal state (not absorbing) we obtain the next equations:

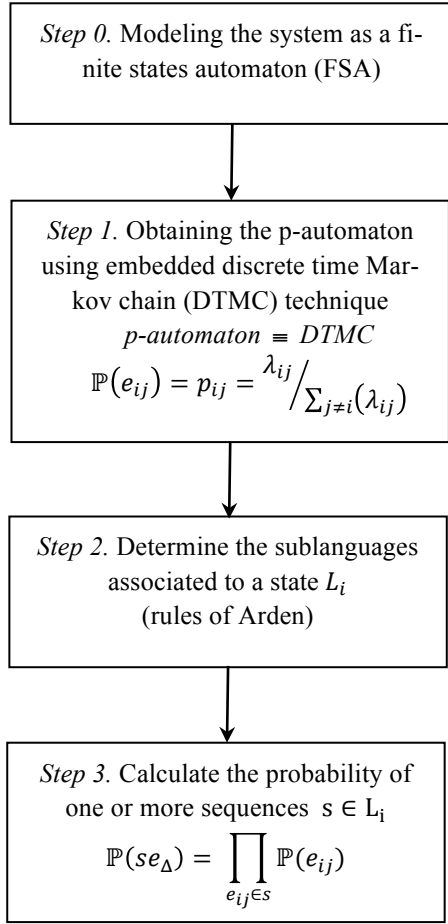


Figure 4. Approach proposed for the probabilistic assessment of events sequences.

$$L_1 = L_2 e_{21} + L_4 e_{41} \quad (13)$$

$$L_2 = L_1 e_{12} + L_2 e_{22} + L_3 e_{32} \quad (14)$$

$$L_3 = L_1 e_{13} + L_2 e_{23} + L_3 e_{33} + L_4 e_{43} \quad (15)$$

$$L_4 = L_3 e_{34} \quad (16)$$

For example the first equation (13) denote the sublanguage associate to state s_1 as being the all events sequences arriving in this state by only one transition: from state s_2 (all the events sequences previously terminated in s_2 : L_2) followed by the transition e_{21} or from state s_4 (all the events sequences previously terminated in s_4 : L_4) followed by the transition e_{41} (according to Figure 3).

Using the Arden lemma, the equations (13-16) give the next expressions for the sublanguages:

- to state 1:

$$L_1 = (e_{12} + e_{13}(e_{33} + e_{34}e_{43})^*e_{32})(e_{22} + e_{23}(e_{33} + e_{34}e_{43})^*e_{32})^*e_{23}(e_{33} + e_{34}e_{43})^*e_{34}e_{41}[(e_{12} + e_{13}(e_{33} + e_{34}e_{43})^*e_{32})(e_{22} + e_{23}(e_{33} + e_{34}e_{43})^*e_{32})^*e_{21} + e_{13}(e_{33} + e_{34}e_{43})^*e_{34}e_{41}]^* \quad (17)$$

- to state 2:

$$L_2 = L_1(e_{12} + e_{13}(e_{33} + e_{34}e_{43})^*e_{32})(e_{22} + e_{23}(e_{33} + e_{34}e_{43})^*e_{32})^* \quad (18)$$

- to state 3:

$$L_3 = (L_1 e_{13} + L_2 e_{23})(e_{33} + e_{34}e_{43})^* \quad (19)$$

- to state 4:

$$L_4 = L_3 e_{34} \quad (20)$$

Step 3: Having the sublanguage defined for each state of the considered system we can obtain firstly the probability for a sublanguage and secondly the probabilities for events sequences belonging to this sublanguage. To illustrate the sequence probabilities assessment, the results obtained for two sublanguages (L_1 and L_3) are presented in this paper.

For the simplification we will use the following notations:

$$p_{12} = p; p_{13} = 1 - p; p_{21} = p_1; p_{23} = p_2; p_{22} = 1 - p_1 - p_2; p_{32} = p_3; p_{34} = p_4; p_{33} = 1 - p_3 - p_4; p_{41} = p_5; p_{43} = 1 - p_5.$$

Probabilities assessment for the sublanguage L_1

Using the expression (17) we can calculate the probabilities of different iterations that compose the regular expression of the sublanguage L_1 and, after that, we obtain the final analytic expression:

$$\mathbb{P}(L_1) = \sum_{s \in \mathcal{E}^*} \mathbb{P}(se_{\Delta}) = \frac{p_2 p_4 p_5}{p_2(p_3 + p_4 p_5) - p_2 p_3} = 1 \quad (21)$$

This result, the probability of the sublanguage $\mathbb{P}(L_1)$, is compliant with the expected one: indeed, for the p-automaton of the Figure 3, the probability of all sequences starting from state 1 and reaching back to state 1 must be obviously equal to one. Nevertheless, the approach is able to provide the probability of all individual sequences reaching to this state. To illustrate this fact, the occurrence probability for some particular sequences, extracted from L_1 and that lead the system in the state 1, are obtained doing the numerical calculus. Their values are presented in Table 1. Firstly we can observe that the sum of probabilities of considered events sequences is approximately equal with 1 and this corresponds to analytical solution of the sublanguage probability (eq. 21). Secondly we can identify the most relevant events sequences (by the greatest values of their probability).

Probabilities assessment for the sublanguage L_3

The sublanguage L_3 describes infinity sequences which bring the system in the failure state 3 from the initial state 1:

$$\mathbb{P}(L_3) = \frac{p_1 + p_2 - p p_1}{p_1(p_3 + p_4 p_5) + p_2 p_4 p_5} \quad (22)$$

Calculating the numerical values for some sequences we obtain the values from Table 2. The state 3 represents the dangerous state because the oven temperature is uncontrolled and thus we can

identify the most critical events sequences (with the greatest values of their probability) of the system.

Table 1. Occurrence probabilities for few sequences that lead the system in the state 1.

Sequence (s_i)	Sequence probability ($\mathbb{P}(s_i)$)
$s_1 = e_{12}e_{21}$	0.9213
$s_2 = e_{12}e_{22}e_{21}$	0.0276
$s_3 = e_{12}e_{23}e_{33}e_{32}e_{21}$	4.9961e-06
$s_4 = e_{12}e_{22}e_{23}e_{32}e_{23}e_{32}e_{21}$	4.8080e-10
$s_5 = e_{12}e_{23}e_{34}e_{43}e_{32}e_{21}$	1.8331e-08
$s_6 = e_{12}e_{22}e_{23}e_{33}e_{34}e_{43}e_{32}e_{21}$	2.2602e-11
$s_7 = e_{12}e_{23}e_{34}e_{41}$	1.0233e-04
$s_8 = e_{13}e_{34}e_{41}$	0.0215
$s_9 = e_{13}e_{33}e_{34}e_{41}$	8.8586e-04
$s_{10} = e_{13}e_{34}e_{43}e_{34}e_{41}$	3.2502e-06
$s_{11} = e_{13}e_{32}e_{21}$	0.0256
$s_{12} = e_{13}e_{33}e_{32}e_{22}e_{23}e_{32}e_{21}$	4.1624e-09
$\mathbb{P}(L_1) = \sum_{s_i} \mathbb{P}(s_i) \cong 1$	

Table 2. Occurrence probabilities for few sequences that lead the system in the state 3.

Sequence (s_i)	Sequence probability ($\mathbb{P}(s_i)$)
$s_1 = e_{12}e_{23}$	2.3744e-04
$s_2 = e_{12}e_{22}e_{21}e_{13}$	0.0014
$s_3 = e_{13}e_{33}$	0.0021
$s_4 = e_{12}e_{22}e_{23}e_{32}e_{23}e_{33}$	3.8620e-11
$s_5 = e_{12}e_{23}e_{33}e_{34}e_{43}$	1.4724e-09
$s_6 = e_{13}e_{33}e_{34}e_{43}e_{32}e_{23}$	4.0899e-11
$\mathbb{P}(L_3) = \sum_{s_i} \mathbb{P}(s_i) \cong 0.0037$	

6.2 Analytic validation of the results

In order to check the validity of the results (sequences probabilities), we propose the next approach:

1. To determine the stationary distribution of states probabilities for embedded DTMC [$\pi_1 \pi_2 \pi_3 \pi_4$] by using the equations:

$$\pi = \pi \cdot M \quad \text{and} \quad \pi \cdot \bar{1} = 1 \quad (23)$$

where the vector π represents the state probability distribution, M is the matrix of transitions probabilities and $\bar{1}$ is a vector with all elements 1.

Using the equations (23) we determined the analytic form for the state probabilities.

2. To prove that the probabilities for different sublanguages, obtained by the equations (17-20), the following relation is to be validated:

$$\mathbb{P}(s_i) \cdot \sum_{\substack{s \in \Sigma^* \\ i \rightarrow j}} \mathbb{P}(se_{\Delta}) = \mathbb{P}(s_i) \cdot \mathbb{P}(L_{i \rightarrow j}) = \mathbb{P}(s_j) \quad (24)$$

It means that the probability to reach state s_j is equal to the product of the probability of the initial state s_i and the probability of the sublanguage from s_i to s_j .

- for L_1 : $\pi_1 \cdot \mathbb{P}(L_1) = \pi_1$
- for L_2 : $\pi_1 \cdot \mathbb{P}(L_2) = \pi_2$
- for L_3 : $\pi_1 \cdot \mathbb{P}(L_3) = \pi_3$
- for L_4 : $\pi_1 \cdot \mathbb{P}(L_4) = \pi_4$

7 CONCLUSIONS

In this paper, we presented the use of the theory of probabilistic language for the assessment of the occurrence probability of events sequences that describe the system evolutions. In order to be able to apply this theory, for a system modelled by a finite state automaton, we proposed a 3 steps approach.

The use of the embedded Discrete Time Markov Chain in a continuous stochastic process for determining the events probability makes assumption that the system is in a stationary state characterizing by stationary distribution probabilities over its states. But the embedded DTMC is not limited to Continuous Time Markov Chain; a DTMC can also be defined from semi-Markov or under some hypothesis from more generally stochastic processes. Another advantage to use the DTMC to obtain the events probability is that the probability of an event is not the same during the system evolution, but can depends on the state where it occurs (in other words the same event can be characterized by different occurrence probabilities). The use of the Arden lemma permits to formally determine the whole set of events sequences, without model exploring. Finally, the probability occurrence for relevant or critical events sequences and for a sublanguage is determined.

The approach is applied on a control system temperature. This case study shows the ability of the proposed approach to take into account together stochastic phenomena characterized by continuous laws of probability (such as failures and reparations processes) and phenomena characterized by discrete laws of probability (such as failures on demand).

This work opens several ways of research in our future activity. First of all, it will be interesting to propose an approach that allows calculating the probabilities of events sequences in a transient system state before reaching the stationary distribution. Secondly, we are interested to apply this approach in more complex systems including components ageing

with the more generally underlying stochastic continuous process. Also, an extension of the operations provided by the p-languages theory should be useful for taking in account different types of systems (non-deterministic systems, reconfigurable systems, systems composed of several sub-systems) in dependability studies.

8 REFERENCES

- Bouissou, M. & Bon, J.L. 2003. A new formalism that combines advantages of fault-trees and Markov models: Boolean logic Driven Markov processes. *Reliability Engineering and System Safety* 82 (2): 149-163.
- Bouissou, M. 2006. Efficient determination of minimal failure scenario for sequential systems. *15^{ème} Congrès de Maîtrise des Risques et de Sécurité de Fonctionnement, Lambda-Mu'2006*, Lille (France).
- Carton, O. 2008. *Langages formels, calculabilité et complexité*. Paris: Ed. Vuibert.
- Cassandras, C.G. & Lafortune, S. 2008. *Introduction to Discrete Event System*, Second Edition. New-York: Springer.
- Chaux, P.Y, Roussel, J.M., Lesage, J.J., Deleuze, G., Bouissou, M. 2013. Towards an unified definition of Minimal Cut Sequences. *4th IFAC Workshop on Dependable Control of Discrete Systems (DCDS 2013)*, York (United Kingdom).
- Dutuit, Y. & Rauzy, A. 2005. Approximate estimation of system reliability via fault trees. *Reliability Engineering and System Safety* 87 (2): 163-172.
- Garg, V.K., Kumar, R. & Marcus, S.I. 1999. A Probabilistic Language Formalism for Stochastic Discrete-Event Systems. *IEEE Trans. on Automatic Control* 44(2): 280-293.
- Ibanez-Llano, C., Rauzy A., Melendez E. & Nieto F. 2010. A reduction approach to improve the quantification of linked fault trees through binary decision diagrams. *Reliability Engineering and System Safety* 95 (12): 1314-1323.
- Papazoglou, I.A. 1998. Mathematical foundations of event trees. *Reliability Engineering and System Safety* 61(3): 169-183.
- Perez Castaneda, G. A., Aubry, J.F. & Brinzei, N. 2011. Stochastic hybrid automata model for dynamic reliability assessment. *Proceedings of the Institution of Mechanical Engineers Part O Journal of Risk and Reliability* 225(1): 28-41.
- Wang, X. & Ray, A. 2004. A language measure for performance evaluation of discrete event supervisory control systems. *Applied Mathematical Modelling* no. 28: 817-833.