

Extracting critical scenarios from a Petri net model using linear logic

Hamid Demmou¹, Sarhane Khalfaoui^{1,2}, Nicolas Rivière¹, Edwige Guilhem²

¹Laboratoire d'Analyse et d'architecture des Systèmes LAAS CNRS
7 avenue du Colonel Roche, F-31077 Toulouse cedex
{hamid, nriviere}@laas.fr

²PSA Peugeot Citroën, Direction des Systèmes d'Information
18 rue des Fauvelles, F-92256 La Garenne Colombes cedex
{sarhane.khalifaoui, edwige.guilhem}@mpsa.com

RÉSUMÉ. La connaissance des scénarios critiques est indispensable, dès la phase de conception des systèmes mécatroniques, afin d'estimer leur sûreté de fonctionnement. Ceci permet de valider les reconfigurations et d'orienter le choix de l'architecture de ces systèmes. Cet article présente une méthode de recherche de scénarios potentiellement dangereux dans un cadre formel (logique linéaire) à partir d'un modèle réseau de Petri. Après un rappel sur le lien entre les réseaux de Petri et la logique linéaire, nous présenterons une méthode de raisonnement arrière et raisonnement avant pour la recherche de scénarios critiques sera ensuite illustrée sur un cas d'étude simple et appliquée sur un exemple de système mécatronique du monde automobile.

ABSTRACT. To evaluate reliability of mechatronic systems, one should know the feared scenarios, in order to choose the safe architecture of the system during the development phase. The aim of this work is to propose a logical based approach (linear logic) for deriving the critical scenarios from a Petri net model. After a brief summary about Petri nets and linear logic, we present an original method for backward reasoning on Petri nets. Then, we illustrate the general approach (combining backward and forward reasoning) for deriving feared scenarios on a case study and apply it on a mechatronic system from the automotive field.

MOTS-CLÉS : réseau de Petri, logique linéaire, mécatronique, sûreté de fonctionnement.

KEY WORDS: Petri net, linear logic, mechatronic, safety.

1. Introduction

The design of new cars includes more and more electronic and computing elements that enhance the capabilities of a vehicle, but makes more complex the safety analysis of such systems composed of mechanic, hydraulic, electronic and computing parts, and called mechatronic systems. Classical methods of safety are not sufficient to deal with this kind of complex and hybrid systems [KHA 02].

The qualitative analysis of Petri net models of mechatronic systems is limited by the combinatorial explosion of states in the reachability graph [MON98]. This qualitative analysis aims at identifying the actions that leads to situations where safety of the persons inside the car is no more guaranteed. The search of the feared scenarios (by exploring the graph) contributes to the evaluation of safety and the choice of the system architecture at the design stage.

One way to avoid the combinatorial explosion is to use directly the Petri net model to extract the feared scenarios. To do so, it is helpful to use linear logic to get a new representation of the Petri net model, and then extract the scenarios from this new representation. The advantage is that with linear logic we can express partial order of firing of transitions and focus the search on the parts of the model that are interesting for safety analysis. This approach is based on the equivalence of reachability in the Petri net and provability of a sequent¹ in linear logic.

We first present a method that links Petri nets and linear logic and then a method of extracting scenarios that will be illustrated on a simple example. This method will be finally applied on a mechatronic subsystem from the car domain.

2. Petri nets and linear logic

2.1 Logical reasoning on Petri nets

One way to deal with reachability in Petri nets is to resolve the characteristic equation: $M' = M + C \cdot \bar{s}$. This equation only gives the necessary condition (not sufficient) of the reachability, but it doesn't give the firing order of transitions of \bar{s} .

Based on the sequent calculus, linear logic [GIR87] helps to get a necessary and sufficient condition of reachability from one marking to another, thanks to the equivalence between the provability of a sequent and reachability in the corresponding Petri net [GIR97].

¹ A sequent is a logic expression of the form : $\Gamma, X \multimap Y, \Delta$ which means: « Γ and X » permit to deduce « Y or Δ ». Γ, X, Y and Δ are logical formulas.

The translation of a Petri net to linear logic has been presented in [PRA99]. A logical formula is associated with each marking and each transition. The left hand of the initial formula (sequent) must hold the list of all the transitions that must be fired to obtain a marking M' from an initial marking M . The building of the proof generates a proof tree beginning by a sequent and finishing by the identity axiom (leaves). Moreover, it is possible to extract information about the firing order of transitions from the proof tree of the sequent [PRA99], and temporal evaluation of scenarios in temporal Petri nets. In this way, linear logic is considered as an analysis tool for Petri nets. Some fundamental rules have to be used such as the left introduction rule of the linear implication.

2.2. Left introduction rule of the linear implication

This rule, noted \multimap_L , acts on the left member of the conclusion of a sequent $(\Gamma, \Gamma', F \multimap G \vdash H)$ and generates two fragments $\Gamma \vdash F$ and $\Gamma', G \vdash H$ as it is shown on the following formula: $\frac{\Gamma \vdash F \quad \Gamma', G \vdash H}{\Gamma, \Gamma', F \multimap G \vdash H} \multimap_L$. When analysing a Petri net with linear logic, the use of this rule corresponds to the firing of a transition.

2.3. Forward reasoning

In this approach the transitions of the Petri net are translated to linear logic propositions. When building the proof tree, the consumption of one proposition will correspond to the effective firing of the transition. For a given Petri net the translation is done as follows:

- An atomic proposition P is associated with each place p of the Petri net
- A monome using the multiplicative conjunction \otimes (TIMES), is associated with each marking, pre-condition $\text{Pre}()$ and post-condition $\text{Post}()$ of transitions.
- To each transition t of the net an implicative formula is defined as follows:

$$t : \quad \bigotimes_{i \in \text{Pre}(p_i, t)} P_i \multimap \bigotimes_{o \in \text{Post}(p_o, t)} P_o$$

Each sequent of the form $M, t_1, \dots, t_p \vdash M'$ expresses the reachability between the marking M and M' , by indicating which are the fired transitions (t_1, \dots, t_p) . The proof is derived in a canonical way [PRA99]. Using the rule for introducing the \otimes connector on the left hand side (\otimes_L) allows changing the initial marking with a set of atomic formulas (tokens, not necessarily used at the same date). By applying the \multimap_L rule, it is now possible to extract the causal relations of the

atomic formulas from marking M to M'. To describe this method we applied it on the following simple Petri net:

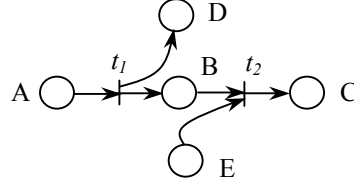


Figure 1. **Example**

The forward reasoning is derived on the canonical form as follows:

$$\begin{array}{c}
 \frac{\frac{B \mid \overline{B} \text{ }^{id} \quad E \mid \overline{E} \text{ }^{id}}{B, E \mid \overline{B \otimes E}} \otimes_R \quad \frac{C \mid \overline{C} \text{ }^{id} \quad D \mid \overline{D} \text{ }^{id}}{C, D \mid \overline{C \otimes D}} \otimes_R}{B, D, E, B \otimes E \multimap C \mid \overline{C \otimes D}} \multimap_L(t_2)}{\frac{A \mid \overline{A} \text{ }^{id}}{B \otimes D, E, B \otimes E \multimap C \mid \overline{C \otimes D}} \otimes_L} \multimap_L(t_1)}{A \otimes E, t_1, t_2 \mid \overline{C \otimes D}} \otimes_L
 \end{array}$$

2.4. Backward reasoning

In this approach, it is possible to do a backward search from the final marking to the initial one. The reasoning is done on resources that can be produced, and we are interested in the date of their production. In the forward reasoning the resources are consumable and we are interested in the date of their consumption. In linear logic it leads to exchange the \otimes connector by the \wp (PAR) connector. So for a given Petri net we have:

- An atomic proposition P associated with each place p of the net
- A monome using the multiplicative disjunction (\wp) is associated with each marking, precondition Pre(), and post-condition Post() of transitions.
- To each transition t of the net an implicative formula is defined as follows:

$$t : \quad \wp_{i \in \text{Pre}(p_i, t)} P_i \multimap \wp_{o \in \text{Post}(p_o, t)} P_o$$

Each sequent of the form $M, !t_1, \dots, !t_n \mid \overline{M'}, \Gamma$ express the reachability between the marking M and M'. This time, it is the final marking M' that we can express with a list of atomic formulas, not necessarily produced at the same date. But the initial marking M must be kept as a monome with \wp . Using the same rule \multimap_L will correspond to the causal relations from the marking M' backward to M.

Applying this method to the Petri net of figure 1 gives the following proof tree:

$$\begin{array}{c}
 \frac{\frac{A \mid A \quad E \mid E}{A \wp E \mid A, E} \wp L \quad \frac{B \mid B \quad D \mid D}{B \wp D \mid B, D} \wp L}{A \wp E, A \multimap B \wp D \mid B, E, D} \multimap L(t_1) \\
 \frac{C \mid C}{A \wp E, A \multimap B \wp D \mid B \wp E, D} \wp R \\
 \frac{\frac{A \wp E, A \multimap B \wp D \mid B \wp E, D}{A \wp E, A \multimap B \wp D, B \wp E \multimap C \mid C, D} \multimap L(t_2)}{A \wp E, t_1, t_2 \mid C \wp D} \wp R
 \end{array}$$

The use of the \otimes (TIMES) connector in the forward reasoning means that resources are simultaneously consumed but not necessarily simultaneously produced. In the backward reasoning the use of the \wp (PAR) connector means that resources are simultaneously produced but not necessarily simultaneously consumed.

2.5. Reasoning in an unknown context

We want to find a sequence of actions (transition firing), and the associated context (necessary tokens) that leads to a token in the place representing the partial feared state. We don't know the initial marking, and about the final marking we only know a part that contains the partial feared state. We don't know which transitions have to be fired. The problem is to write the right sequent that will initiate the desired search. It is necessary to write the list of the transitions that have to be considered, without knowing how many times exactly they will be fired. To express this kind of constraint in linear logic we use the exponential connector "!". When we write $!t$ in a sequent, it means that transition t can be fired zero, one or k times, depending on the needs and the progress of the proof. If M_d represents the partial feared state, the sequent that initiates the backward reasoning will be: $M, !t_1, \dots, !t_n \mid M_d, \Gamma$ where Γ is a context that must be produced simultaneously with M_d , and t_1, \dots, t_n represent all the transitions of the Petri net. The formula $M, \Gamma, !t_1, \dots, !t_n \mid M'$ can be used in the same way for the forward reasoning.

3. Extracting critical scenarios: a general method

The aim of a qualitative analysis is to point out the sequence of actions that leads to the feared states and to analyse more precisely what makes the system leave the normal behaviour and reach the feared state. Our method starts by a backward reasoning from the feared state in order to identify the causal chain of actions leading to that feared state. The backward reasoning is stopped when a nominal state is reached. A forward reasoning follows it in order to obtain all the possible evolutions from this partial nominal state. The bifurcation between the nominal behaviour and the feared one is identified and corresponds to a transition conflict in the Petri net.

3.1. Case study

The proposed approach is now illustrated in the following example:

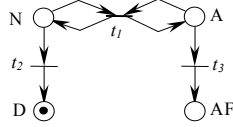


Figure 2. **Case study**

Place D represents the feared state, place N a normal behaviour state, place A a non faulty actuator, and place AF a faulty actuator. Transition t_1 corresponds to a normal behaviour. We are searching for all the scenarios (set of transition firings) that lead to the marking of place D. By applying our method it is possible to find out, in a logical framework, the causal link between the marking of D and that of AF.

3.1.1. Backward reasoning

At this stage, the transition of the Petri net are expressed as follows:

- $t_1: A \wp N \multimap A \wp N$,
- $t_2: N \multimap D$,
- $t_3: A \multimap AF$.

The initial sequent expressing the reachability of the marking of D is:

$M, !t_1, !t_2, !t_3 \vdash D, \Gamma_1$ (1). Only transition t_2 produces a token in place D, so the sequent can be rewritten $M, t_2, !t_1, !t_2, !t_3 \vdash D, \Gamma_1$. Then we apply the $\multimap L$ rule to the sequent (2):

$$\frac{M, !t_1, !t_2, !t_3 \vdash N, \Gamma_1 \quad D \vdash D}{M, t_2, !t_1, !t_2, !t_3 \vdash D, \Gamma_1} \multimap L$$

We obtain a first sequent that expresses the reachability of the marking of place N: $M, !t_1, !t_2, !t_3 \vdash N, \Gamma_1$ (3).

It can be noticed that only transition t_1 produces a token in place N, but in the same time, it produces a token in place A. Let's put $\Gamma_1 \equiv A \wp \Gamma_2$. It corresponds to the enrichment of the context of the marking, assuming that place A has a token that will be used at the same time than the one in place N. We apply again the $\multimap L$ rule to the sequent (3), using the expression of Γ_1 :

$$\frac{M, !t_1, !t_2, !t_3 \vdash N, A, \Gamma_2 \quad N \wp A \vdash N, A}{M, t_1, !t_1, !t_2, !t_3 \vdash N, A, \Gamma_2} \multimap L$$

We can see that the initial sequent, $M, !t_1, !t_2, !t_3 \vdash N, A, \Gamma_2$ of the proof tree, and the sequent obtained after the application of the \multimap_L rule, are the same. We stop the process of building the proof and put $\Gamma_2 \equiv \perp$ (\perp is the neutral element of the disjunctive multiplication (\wp)). Let's put $M \equiv N \wp A$. We obtain the following cycle $N \wp A, !t_1 \vdash N, A$.

During the proof building, we applied twice the \multimap_L rule. This corresponds to a reverse firing of transition t_2 , followed by an undefined number of firing of t_1 . The final sequent resuming all the steps is: $N \wp A, !t_1, t_2 \vdash D \wp A$.

3.1.2. Forward reasoning

Thanks to the backward reasoning we have identified a scenario leading to the marking of place D. It represents the reachability of this marking from the marking $N \otimes A$, after an undefined number of firings of t_1 , followed by one firing of t_2 . We are now going to verify if, starting from the marking $N \otimes A$, we obtain a marking different from $D \otimes A$, with an indeterminate order of the transition firings.

The transitions of the Petri are now expressed as follows:

- $t_1: A \otimes N \multimap A \otimes N$,
- $t_2: N \multimap D$,
- $t_3: A \multimap AF$.

The initial sequent is: $N \otimes A \otimes \Gamma_3, !t_1, !t_2, !t_3 \vdash \Gamma_4$, with Γ_3 and Γ_4 representing a priori unknown marking context. We can see that the transitions t_1 and t_2 are in conflict, and also t_1 and t_3 , but not t_2 and t_3 . As a consequence we determine two different proof trees:

- The one corresponding to the firing of t_1 (tree 1)
- The one representing the firing of the sequence $\{t_2; t_3\}$ (tree 2).

3.1.2.1. Proof Tree 1

Firing the transition t_1 gives: $\frac{N, A \vdash N \otimes A \quad N, A, \Gamma_3, !t_1, !t_2, !t_3 \vdash \Gamma_4}{N, A, \Gamma_3, t_1, !t_1, !t_2, !t_3 \vdash \Gamma_4} \multimap_L$. We

obtain the same sequent. So we put $\Gamma_3 \equiv 1$ and $\Gamma_4 \equiv N \otimes A$ (so that $N, A \vdash \Gamma_4$ is provable). The obtained scenario is: $N \otimes A, !t_1 \vdash N \otimes A$. It corresponds to a linear invariant of transitions.

3.1.2.2. Proof Tree 2

The transitions t_2 and t_3 are parallel, so their firing order is not significant. Let's choose to fire t_2 first and write the corresponding proof:

$$\frac{N \vdash N \quad D, A, \Gamma_5, !t_1, !t_2, !t_3 \vdash \Gamma_6}{N, A, \Gamma_5, t_2, !t_1, !t_2, !t_3 \vdash \Gamma_6} \text{---o}_L . \text{ Now, we can fire } t_3 \text{ and write:}$$

$$\frac{A \vdash A \quad D, AF, \Gamma_5, !t_1, !t_2, !t_3 \vdash \Gamma_6}{D, A, \Gamma_5, t_3, !t_1, !t_2, !t_3 \vdash \Gamma_6} \text{---o}_L . \text{ We stop the proof because there is no}$$

more fireable transitions. We put $\Gamma_5 \equiv 1$ and $\Gamma_6 \equiv D \otimes AF$. Finally we obtain the following sequent: $N \otimes A, t_2, t_3 \vdash D \otimes AF$. From this sequent we can see that the scenario leading to the marking of D, produces simultaneously the marking of the place AF.

3.1.3. Discussion

Our objective is to identify all the scenarios leading to markings containing place D. We started from a sequent expressing the reachability of the marking of D, from an unknown initial marking. By applying a backward reasoning on this sequent and then a forward reasoning, we obtain the final sequent $N \otimes A, !t_1, t_2, t_3 \vdash D \otimes AF$ that contains all the possible scenarios leading to the marking of place D. From the proof tree we deduce two results:

- If the firing of t_1 is the normal behaviour, then the state with the marking of D is irreversible (if t_2 or t_3 is fired, it is no more possible to fire t_1),
- The obtained final sequent associating the marking of D and AF, gives more information about the conditions of the occurrence of the feared state, than the one that leads to the marking of D only.

4. An application example from the car domain

The case we will present is an electromechanical system (figure 3) made of a pump, an accumulator, a tank, an electrovalve and a sensor. A computer is used to control the system. This system regulates the oil pressure in the accumulator. This pressure must be kept inside a given interval. The computer opens or closes the electrovalve depending on the value of the pressure given by the sensor. The normal behaviour of the system can be described by the following phases:

- A conjunction phase with high consumption when the electrovalve is open and the hydraulic circuits are strongly used. The pressure in the accumulator is going up during this phase.
- A conjunction phase with high consumption when the electrovalve is closed and the hydraulic circuits are used. The pressure in the accumulator is going down during this phase.
- A disjunction phase with low consumption when the electrovalve is closed. We consider no consumption so the pressure is constant.

- A disjunction phase with low consumption when the electrovalve is open. As we consider that there is no consumption, the pressure is going up.

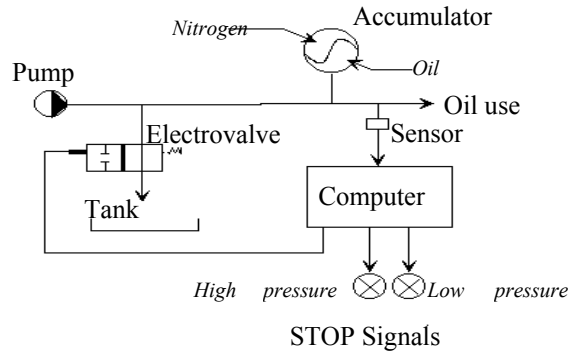


Figure 3. Oil alimention electromechanical system

4.1. Model with failures and reconfigurations

In the model, we suppose that only the electrovalve can have failures which are: blocked in open state or blocked in closed state.

When the electrovalve is blocked (failure), a possible reconfiguration is to shake the electrovalve for 0.1 seconde. If the reconfiguration is successful, it is unblocked (with probability p) and the system is back to a normal state. In case the reconfiguration is not successful (probability $1-p$), the system will reach a faulty state and the alarm will be put on.

4.2. A Petri net model of the system

First we model the normal behaviour of the system, and secondly we introduce the failures and reconfigurations of the electrovalve.

4.2.1. Model of the normal behaviour

Four differential equations express the evolution law of the pressure in the accumulator. Each equation corresponds to a combination of the electrovalve state (open, closed) and to the consumption of the hydraulic circuits (high, low).

In figure 4 we see the Petri net model of the system. It is a Differential-Predicat-Transition net [CHA98], where the differential equations are associated with places P_1, P_2, P_3, P_4 . In fact P_1 and P_4 represent the conjunction phase, and P_3 and P_2 represent the disjunction phase. The states of the electrovalve correspond to P_5 (open) and P_{15} (closed).

The transitions t_{41} and t_{32} represent the changing of the consumption from high to low, and the transitions t_{14} and t_{23} from low to high.

The transitions t_{12} and t_{43} represent the closing command of the electrovalve when $P \geq P_{\max}$, and t_{21} represent the opening command when $P \leq P_{\min}$.

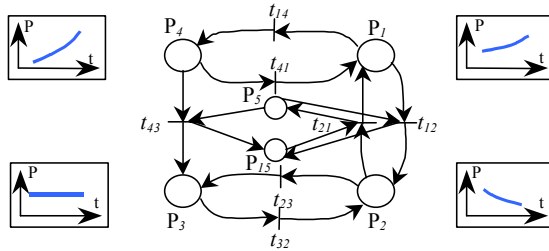


Figure 4. Petri net model of the electromechanical system

4.2.2 Modelling the failures and reconfigurations of the electrovalve

In figure 5, when a token is put in P_0 it means that there is a failure on the electrovalve. If it is in open state, P_5 is also marked and if $P \geq P_{\max}$ then the reconfiguration starts with the firing of t_5 followed by the marking of P_6 (corresponding to the blocked open state). The transition t_6 is immediately fired and P_7 is marked. Depending on the probability p of success of the reconfiguration, transition t_7 is fired in case of success. The electrovalve is unblocked by firing transition t_9 and marking place P_5 . In case the reconfiguration is unsuccessful, transition t_8 is fired and place P_9 is marked.

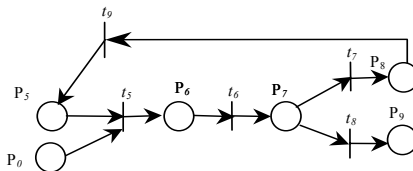


Figure 5. Failure and reconfiguration model

The figure 6 shows a simple model for the alarm signal. When the pressure goes out of the interval $[P_{\text{alarm_min}}, P_{\text{alarm_max}}]$ the alarm is set. Place P_l represents a conjunction phase with low consumption. P_h represents the state of the system when the pressure is over the high security limit, and transition t_1 is fired when this limit is crossed.

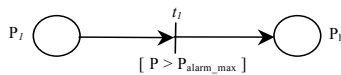


Figure 6. Alarm signal model

4.2.3. A complete model of the system

The complete Petri net model of the system is a composition of the three previous models as it is shown in figure 7.

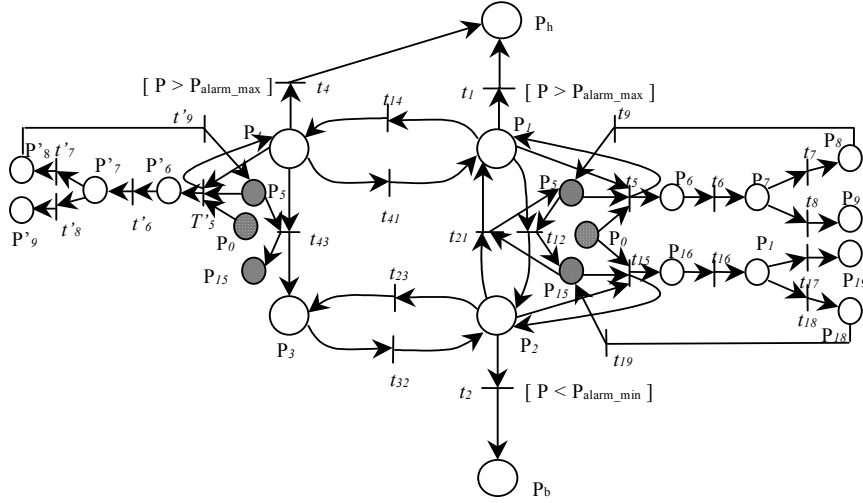


Figure 7. A complete Petri net Model of the system

4.3. Extracting critical scenarios

Our method aims at pointing out the critical scenarios and the bifurcations between the normal states and the feared one. We are now going to describe how the method is applied on the model of figure 7 in order to identify the scenarios leading to the making of P_b , which is considered as a feared state (low pressure alarm activated).

The initial sequent is $M_1, !t_1, \dots, !t_n \mid\!\!-\ P_b \wp M_2$. After the first step of the backward reasoning, we obtain the sequent: $P_2, t_2 \mid\!\!-\ P_b$ that corresponds to the reachability of place P_b from P_2 after one firing of t_2 . The backward reasoning is stopped because P_2 corresponds to a nominal behaviour state.

The forward reasoning starts with a marking including place P_2 . The outgoing transitions of P_2 are t_2 , t_{23} , t_{21} and t_{15} . We can deduce three conflicting scenarios described by the following sequents:

$$P_2, t_2 \mid\!\!-\ P_b \quad (4), \quad P_2, !(t_{23}, t_{32}) \mid\!\!-\ P_2 \quad (5), \quad P_2 \otimes P_{15}, t_{21} \mid\!\!-\ P_1 \otimes P_5 \quad (6)$$

From the conflict between t_{17} and t_{18} that appears when we generate sequent (4), two more scenarios can be deduced:

$$P_2 \otimes P_{15} \otimes P_0, t_{15}, t_{16}, t_{18}, t_{19} \mid\!\!-\ P_2 \otimes P_{15} \quad (7), \\ P_2 \otimes P_{15} \otimes P_0, t_{15}, t_{16}, t_{17}, t_2 \mid\!\!-\ P_b \otimes P_{19} \quad (8).$$

The sequent (4) is obtained after a backward reasoning. Sequent (5) represents the changing cycle of the consumption (high, low). Sequent (6) represents the opening of the electrovalve when the pressure reaches the lower limit. Sequent (7) represents the failure scenario of the electrovalve when it is closed and the reconfiguration is successful. Finally, sequent (8) represents a failure scenario with unsuccessful reconfiguration and the setting of the low pressure alarm.

After analysing this results we can conclude that there is only one feared scenario leading to low pressure alarm activation and it is the one identified by sequent (8).

5. Conclusion

In this paper we presented a method based on Petri nets and linear logic to extract critical scenario. The extraction process is based on backward and forward reasoning on the model using linear logic. The method allows to identify critical scenarios but also to localise the bifurcation between normal and failure states. The originality of this method is that it uses linear logic to derive scenarios from a Petri net model. These scenarios enrich the information given by the fault tree method because our approach respects the hybrid and dynamic nature of mechatronic systems [KHA 02]. The next objective is to complement and improve our method by an algorithm in order to derive automatically feared scenarios.

6. Bibliography

- [CHA 98] Champagnat R., Esteban P., Pingaud H., Valette R., « Modeling and simulation of a hybrid system through Pr/Tr PN DAE model », *ADPM'98 3rd, International Conference on Automation of Mixed Processes*, March 19-20, 1998, Reims, France, p. 131-137.
- [GIR 87] Girard J.Y., « Linear Logic », *Theoretical Computer Science*, 50, 1987, p.1-102.
- [GIR 97] Girault F., « Formalisation en Logique Linéaire du fonctionnement des réseaux de Petri », Thèse de Doctorat, N°2870, Université Paul Sabatier, Toulouse.
- [KHA 02] Khalfaoui S., Guilhem E., Demmou H., Valette R., «A method for deriving critical scenarios in mechatronic systems», *λμ13, European Conference on System Dependability and Safety*, March 18-20, 2002, Lyon, France.
- [MON 98] Moncelet G., « Application des réseaux de Petri à l'évaluation de la sûreté de fonctionnement des systèmes mécatroniques du monde automobile », Thèse de Doctorat, N°3076, Université Paul Sabatier, Toulouse.
- [PRA 99] Pradin-Chézalviel B., Valette R., Künzle L.A., « Scenario duration characterization of t-timed Petri nets using linear logic », *IEEE PNPM'99, 8th International Workshop on Petri Nets and Performance Models*, Zaragoza, Spain, September 6-10, 1999, p.208-217.