RELIABILITY ASSESSMENT OF EMBEDDED SYSTEMS USING STOPWATCH PETRI NETS

Afifa Ghenai[1], Mohamed Youcef Badaoui[1] and Mohamed Benmohammed[1]

[1]LIRE Laboratory, Computer Science Department, Mentouri University,
Constantine, 25000, Algeria
afifa.ghenai@gmail.com, joseph-moh@hotmail.com, ben_moh123@yahoo.com

ABSTRACT

In this paper, we propose a reliability approach in which feared events define reliability requirements and taking them into account allows to design systems which will be able to avoid the drift towards a feared state. The description of feared scenarios since the system design phase enables us to understand the reasons of the feared behavior in order to envisage the necessary reconfigurations and choose safe architectures. In order to face the increasing complexity of embedded systems and to represent the suspension and resumption of task execution we propose to extract directly feared scenarios from Stopwatch Petri net model avoiding the generation of the associated reachability graph and the eternal combinative explosion problem.

KEYWORDS

Embedded Systems, Reliability, Time Constraints, Feared Scenarios, Stopwatch Petri Nets

1. INTRODUCTION

Embedded systems must answer several requirements of which the criticality, which requires a guarantee of the major challenge: a suitable level of reliability. One of the principal problems encountered when we study the reliability of these systems is the taking into account efficiently and in realistic way of time constraints to which they are subjected.

Verifying time constraints is in particular difficult to carry out by traditional tests, since it would be necessary in theory to test infinity of different time sequences. An alternative is then to build a formal model of the system. The sure design and the development of complex systems require in particular their modeling according to a rigorous formalism [4]. To this end, many formal languages were developed among those, Stopwatch Petri Nets model (SWPN) is a powerful tool of design and analysis, particularly adapted to the description of embedded systems. Stopwatch Petri nets were proposed because timed models are not sufficient to model and verify real time applications. Indeed, in these models, time passes in an identical way for all the components of the system, what does not make it possible to represent the preemptive policies of scheduling where the execution of task is stopped and restarted at the same place later [5]. Consequently, it is necessary to represent the suspension and resumption of task execution by considering models with stop watches in which the concept of clock used in timed models is replaced by a stop watch. Contrary to a clock, a stop watch can be stopped (it preserves its value during the passing of time) then started again. The use of SWPN model enables us to express temporal behaviors better than TPN model by taking into account the interruption and resumption of tasks, and thus, to propose a

DOI : 10.5121/ijcsea.2012.2504

more detailed configuration of the system which gives more feared scenarios (more dangerous behaviors).

The rest of the paper is organized as follows: after presenting the feared scenarios approach in section 2, we present our approach and discuss its advantages in section 3. In section 4, we present a case study to illustrate our approach. In section 5, we present the application of the proposed method and discuss the obtained results. Finally, we conclude the paper in section 6.

2. FEARED SCENARIOS APPROACH

The feared scenarios approach proposed by Khalfaoui [1], allows to use directly Petri net model without generating the associated reachability graph, to extract scenarios carrying out towards a critical condition (called : *feared scenarios*) which are indeed, unknown during the design phase of embedded systems. Medjoudj in [2] and Sadou in [3] improved and implemented the approach of Khalfaoui to determine more precisely, by a time Petri net model, the exact conditions of the feared event occurrence.

2.1. Feared scenarios

A scenario implies a beginning, an end and a history which describes the evolution of a system. In the context of reliability, a feared scenario leads to a catastrophic or dangerous state: the final state (called: feared state). The initial state is a state of good functioning of the system. The feared scenario describes how the system leaves the good functioning to evolve to a functioning considered to be dangerous. The definition of a scenario is based on the concept of event and the relations between the events.

Definition 1. (Event): Let be a Petri net (P, T, Pre, Post), M_0 its initial marking. An event is a particular firing of a transition $t \in T$. the set of events is noted E. For example, if during an evolution of the Petri net from M_0 the transition ti is fired for the j^{th} time, we will say that it is the occurrence of the event e_i^j

Definition 2. (Scenario): A scenario sc, noted sc = (l, $\square_{sc}$) associated with the Petri net P and the couple M_0 and M_F markings, is a set of events l provided with a strict partial order $\square_{sc}$ defined on the events of l. If for e1, e2 $\in$ l we have e1$\square$ e2, that wants to say that the event e1 precedes the event e2 in the scenario sc [3].

3. OUR APPROACH

To circumvent the combinative explosion problem of the reachability graph of an embedded system, The proposed approach in [1], improved and implemented in [2] and [3] and which is based on the extraction of feared scenarios from a Petri net model, seems to us well adapted to face the increasing complexity of embedded systems. For a good taking into account of time constraints and to represent the suspension and resumption of task execution, we propose to extract directly feared scenarios from Stopwatch Petri net model which is a powerful tool of design and reliability analysis of real time systems.

3.1. Stopwatch Petri nets (Post and Pre initialization)

Stopwatch Petri nets (SWPN) extend Time Petri nets (TPN) by including in its semantics the behavior of interruptible systems. That implies the suspension and resumption of tasks execution at the time of interruptions.

In a SWPN [6], there are two types of transitions: interruptible and non-interruptible transitions. A mechanism of initialization of the stop watches called post-initialization is used. This mechanism is based on the firing of the corresponding interruptible transition. The firing of an interruptible transition puts at zero the stop watch associated with this transition, while the firing of another transition which desensitizes the interruptible transition, suspends this stop watch. It begins again when the interruptible transition is sensitized again.

SWPN offer a simple graphic formalism where the modification concerns only the initialization of the clocks, stopwatches can be reset, stopped and started [7]. That enables to represent interruptible systems which are a class of real time systems. SWPN are based on more simple principles than IHTPN (Time Petri Nets with Inhibitor Hyper arcs) which connect a place to an interruptible transition by an inhibitor arc [8]. SWPN combine the concision of Petri net model and the analysis power of stopwatches automata.

In the following section, we present the basic steps of the feared scenarios generation method using a stopwatch Petri net model.

3.2. Feared scenarios generation method using stopwatch Petri nets

3.2.1. Principle

For a good taking into account of time constraints and to represent the interruption and resumption of task execution, we improve the algorithm presented in [2] and [3], to which we refer the interested reader for complete details, since the algorithm is so long. In our approach, we propose a more detailed configuration of the system using Stopwatch Petri nets which gives us more feared scenarios (more dangerous behaviors). These scenarios are not taken into account by the preceding feared scenarios approaches.

The method is based on an analysis of the stopwatch Petri net model of the system. It enables to extract feared scenarios and consists on going up the chain of causalities by a back reasoning starting from the feared state until we arrive in a normal functioning state. Then a front reasoning is carried out starting from the normal state in order to understand the conditions of occurrence of the feared behavior.

3.2.2. Method steps

The proposed method contains four steps. Its goal is to determine the conditions of occurrence of the feared event in the form of a scenario. Figure1 presented in [2] shows the method principle.

- The first step is the determination of nominal states of the system: this step determines the places whose marking represents a normal functioning state. These states are either known or obtained by a Monte Carlo simulation.

- The second step is the determination of target states: a target state is either the feared state (E.P.R) or the states which have direct or indirect causal relations with the feared state (E.P.D).
- The third step is the back reasoning using the inverted stopwatch Petri net model of the system and starting from the target state: this step determines the different normal functioning states (E.P.N) from which the system can leave its normal functioning towards a dangerous behavior. It consists of going back up through all the possible preceding states, until we reach the normal functioning states called: the conditioners states. These states are the starting points of the next step.

- The fourth step is the front reasoning starting from the conditioners states determined at the preceding step. The goal is to determine all the possible sequences which lead to the feared state from the initial conditioner state. The bifurcations between the normal functioning and the feared behavior (a bifurcation is a conflict between transitions) determine the precise information of the feared event context. The analysis of these conflicts begins by the marking enrichment in order to determine the events which lead the system to the feared state.

The maximal marking enrichment consists of introducing the maximum of tokens in the unmarked input places of the potentially fired transitions involved in a conflict. This enrichment enables us to find the normal conditions which allow the priority transition to be fired as many times as possible. Since the priority transition is fired, the system remains in its normal functioning. The minimal marking enrichment consists of introducing the minimum of tokens in the unmarked input places of the potentially fired transitions which have a relation with the feared state but are not involved in a conflict.

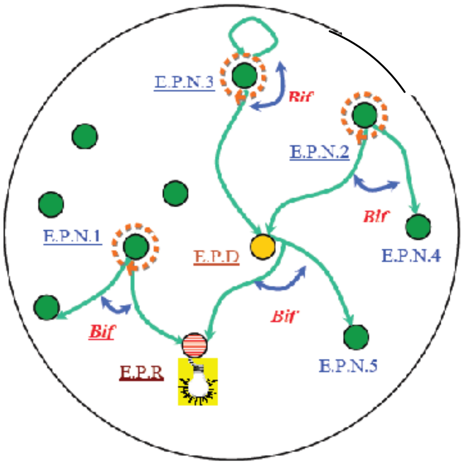

Figure1. The principle of the feared scenarios method

The use of SWPN model enables us to express temporal behaviors better than TPN model [9] by taking into account the suspension and resumption of tasks. The advantage of our method is that the more detailed configuration of the system enables us to find new feared scenarios.

Indeed, modeling interruptible systems using stopwatch Petri nets generates two kinds of bifurcations. The new bifurcations are the conflicts between transitions which represent the non-resumption of interruptible transitions and transitions which represent the resumption (the normal functioning) of interruptible transitions. The condition which leads the system to the feared state is the firing of the interruptible transition which cannot be resumpted because of non-respect of time constraints.

This condition allows the generation of new feared scenarios which are not found by the preceding feared scenarios approaches. Consequently, in the front reasoning step of the method, we must memorize the two kinds of bifurcations encountered. The stopwatch value enables to determine the presence of the new kind of bifurcations. In this case, the system reconfiguration is based on modification of time constraints.

3.2.3. Data structures

In the new version of the feared scenarios generation algorithm, input data are changed. We add the input α_{max} which represents the maximum stop time of a task. If the stop time of a task exceeds α_{max}, the system cannot make the resumption of task. This is expressed by the addition of the procedure « check transition (t_k) ». In the procedure « fire transition (t_k) » we add the condition: if the transition to be fired is that of the stop, then the stopwatch α starts.

3.2.3.1 Input Data

They are made up of the list of the initial tokens (L_i) and the list of normal tokens (L_n), the maximum stop time α_{max} which enables to define a restart condition of task, and the list of prohibited transitions (Lint).

3.2.3.2. Output Data

It is the result of the algorithm. It contains all the partial orders corresponding to the various scenarios.

3.2.3.3. Internal data

- The current list (**L_c**) contains the list of the current tokens.
- The list of prohibited transitions (**L_{int}**).
- The list of transitions of non-initial normal tokens (**Ln**).
- The list of particular transitions (**L_p**). It contains resumption transitions (**t_r**), and stop transitions (**t_s**).
- Stop time of each task (α), it enables to calculate the duration of a task suspension.
- The context (**C_i**), **L_c** is the current list.

Other lists of internal data are generated from the current list L_c:

- **TfscEc** : the set of fired transitions without conflict with fired transitions.
- **Tpfsc** : the set of potentially fired transitions without conflict.
- **Tfcpf** : the set of fired transitions in conflict with at least a potentially fired transition.
- **Tpfc** : the set of potentially fired transitions in conflict either with fired transitions or with potentially fired transitions.

3.2.3.4. Procedures

Since the feared scenarios generation algorithm is so long we present in this paper only some procedures changes. The interested reader can find complete details of the old version of the algorithm in [2] and [3].

- Fire a transition (t_k): In this procedure, the current list is updated when the transition is fired by removing consumed tokens and adding produced tokens. Events are memorized in E and arcs corresponding to a precedence relation between two events are memorized in A.

If $t_k = t_s$ then

- Add t_s in E
- For each token (t_i,p) necessary to fire t_s , remove (t_i,p) from L_c list and add (t_i,t_s) in A ;
- For each output place p_s of t_s, add a token (t_s, ps) in Lc.

α++ ;

```
    -   If $P_k$ is a normal place, add $P_k$ to $L_{nni}$
Else
    -   Add $t_k$ in E
    -   For each token $(t_i,p)$ necessary to fire $t_k$ remove $(t_i,p)$ from $L_c$ list and add $(t_i,t_k)$ in A ;
    -   For each output place $p_s$ of $t_k$, add a token ($t_k$, ps) in Lc.
    -   If $P_k$ is a normal place, add $P_k$ to $L_{nni}$
```

-Specify a transition (t_k): This procedure enables to compare the value of α with α_{max} , if $\alpha \leq \alpha_{max}$ then make the resumption of task. However, if $\alpha > \alpha_{max}$ then fire another transition that does not allow the resumption.

```
 If $\alpha \leq \alpha_{max}$ then
    -  Add $t_r$ in E
    -  For each token $(t_i,p)$ necessary to fire $t_r$ , remove $(t_i,p)$ from $L_c$ list and add $(t_i,t_r)$ in A ;
    -  For each output place $p_s$ of $t_r$, add a token ($t_r$, ps) in Lc.
    -  If $P_k$ is a normal place, add $P_k$ to $L_{nni}$

 Else ($\alpha > \alpha_{max}$ )
    If $\exists$ $t_k$ then
    -  Remove $t_r$ from the list of sorted transitions.
    -  Add $t_k$ in E
    -  For each token $(t_i,p)$ necessary to fire $t_k$ remove $(t_i,p)$ from $L_c$ list and add $(t_i,t_k)$ in A ;
    -  For each output place $p_s$ of $t_k$, add a token ($t_k$, ps) in Lc.
    -  If $P_k$ is a normal place, add $P_k$ to $L_{nni}$
```

-Sort transition (t_k): We associate respectively the time intervals: I_k , .., I_{k+1} to the transitions t_k , .., t_{k+1}.

I_{k+1} =[t_{kmin} , t_{kmax}]. The transition t_k can be fired in T_k units of time, with: $t_{kmin} \leq T_k \leq t_{kmax}$.
I_{k+1} =[t_{k+1min} , t_{k+1max}]. The transition t_{k+1} can be fired in T_{k+1} units of time,
with: $t_{k+1min} \leq T_{k+1} \leq t_{k+1max}$

We choose strong semantics of time Petri nets which imposes that a transition t_k must be fired at the latest at its date of firing at the latest: t_{kmax} .

Sort transition (t_k)

```
For each transition $t_k$ , $k \in \{1,2……………K….n\}$ /$n \in N$ do
If $t_{kmin} < t_{k+1min}$ then $t_K$ is the first transition to be fired.
Else
    If $t_{kmin} > t_{k+1min}$ then $t_{K+1}$ is the first transition to be fired.
    Else
        $t_{kmin} = t_{k+1min}$ then
        If $t_{kmax} < t_{k+1max}$ then $t_K$ is the first transition to be fired.
        Else $t_{kmax} > t_{k+1max}$ then $t_{K+1}$ is the first transition to be fired.
```

4. Case Study

4.1. Description

To illustrate the new proposed version of feared scenarios generation algorithm, the example chosen is an embedded system presented in [3]. It is a tank system regulation which contains a calculator, two pumps and three electrovalves, two level sensors, two regulated tanks and an emptying tank. The level of the two tanks (tank1 and tank2) must remain in the interval:
[v_{imin},v_{imax}], (i=1 or 2).

The calculator (computer) controls the level of a tank (the level is given by the sensor) and supplies the tank by feeding the electrovalve. For each tank, there are two functioning phases according to the opening or the closing of the electrovalve which feeds the tank:

- A phase when the electrovalve is open, the tank level is increasing.
- A phase when the electrovalve is closed, the tank level is decreasing.

When the tank level exceeds the upper limit V_{imax}, the calculator orders the closing of the electrovalve. However, when the tank level is lower than the lower limit V_{imin} , the calculator orders the opening of the electrovalve and the functioning phase is changed. This is the calculator control law for each tank. This system must ensure the supply of users and avoid the overflow of tanks. Indeed, a third electrovalve between the two tanks is intended to be used to ensure the emptying of a tank when its level exceeds the safety limit (V1L). The calculator orders the opening of this electrovalve until the level becomes lower than the lower limit V_{imin}. The third electrovalve can be used by only one tank at the same time.

We suppose that only actuators (the electrovalves: EV1 and EV2) can undergo failures: they can be blocked from closing or blocked from opening with a possibility of repair. Sensor failures are integrated in electrovalves failures. If the third electrovalve EV3 undergoes a failure, it is put out of service.

We suppose that the electrovalve EV1 is blocked from closing (it cannot be closed), the level of tank1 increases until V1L. In this case, the electrovalve EV3 must be opened to empty the tank1.

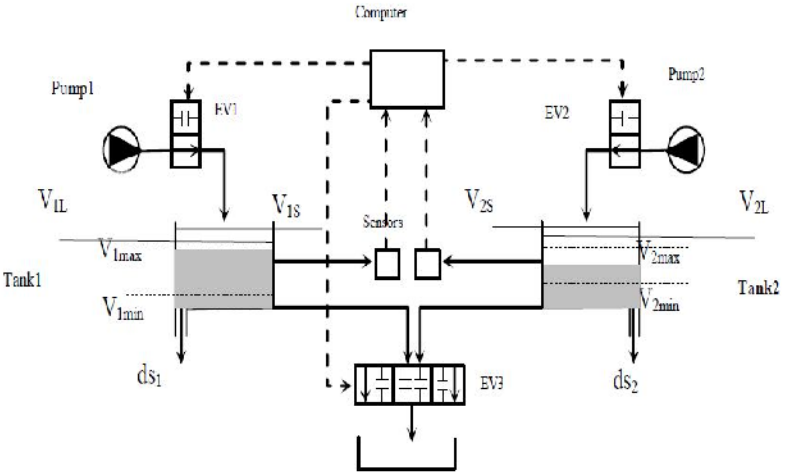

Figure 2. Tank system regulation

If the third electrovalve EV3 is out of service or is ensuring the emptying of the second tank, the level of tank1 exceeds V1L and increases until V1S. The result is the overflow of tank1 which is: a feared event. The second feared event is the overflow of tank2 when the level of tank2 increases until V2L and the electrovalve EV2 is blocked from closing [10].

4.2. System modeling using Time Petri nets and Stopwatch Petri nets

To model the tank system regulation, we propose two classes: a tank class and two electrovalve classes. Tank1 and tank2 are two instances of the tank class, EV1 and EV2 are two instances of the first electravalve class and EV3 is an instance of the second electrvalve class. We represent classes with two formalisms: Time Petri Nets (TPN) and Stopwatch Petri Nets (SWPN) in order to represent the suspension and resumption of tasks in each component and to show the difference between the results obtained by the application of the old version of the feared scenarios generation algorithm and those obtained by the application of the new proposed version of the feared scenarios generation algorithm. The representation with time Petri nets is presented in [3].

4.2.1. Time Petri net representation of the tank model

Figure 3 and Figure 4 presented below show a time Petri net representation of tank1 model and a time Petri net representation of tank2 model respectively. In Figure 3, the place V1_dec represents the disjunction phase (when the tank level decreases) and the place V1_cr represents the conjunction phase (when the tank level increases). When the level reaches V_{1max} , the tank1 calls the `close electrovalve_1' method which corresponds to the transition t11. When the level becomes lower than $V_{1min}$ , the tank1 calls the `open electrovalve_1' method which corresponds to the transition t12. When the tank1 level exceeds V_{1L} , the tank1 calls the `open electrovalve_3' method (to empty the tank1 until its level reaches V_{1min}) which corresponds to the transition t14. We suppose that the minimum duration for closing the electrovalve EV1 is 2 units of time and its maximum duration is 7 units of time. Time corresponding to the time constraint is represented by the interval [2, 7] associated to the transition t11. Since there is no other input places of the transition t11, the token must remain in the place V1_dec at least 2 units of time and 7 units of time at the most before the firing of this transition (the place V1_cr marking corresponds to the conjunction phase).

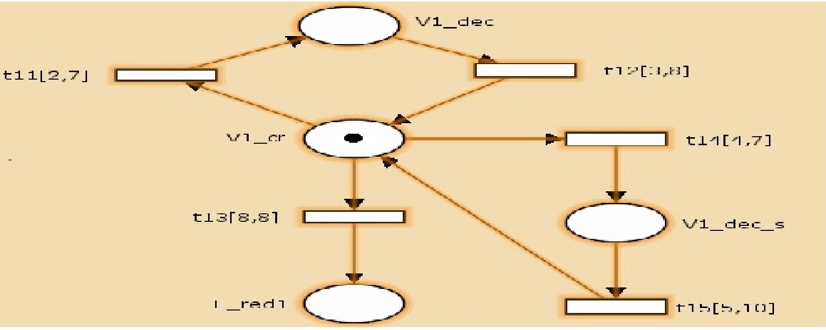

Figure 3. TPN representation of the tank1 model

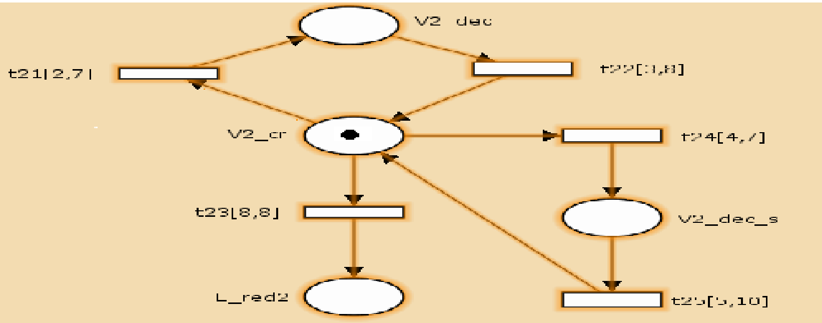

Figure 4. TPN representation of the tank2 model

4.2.2. Stopwatch Petri net representation of the tank model

Figure 5 and Figure 6 presented below show a stopwatch Petri net representation of the tank1 model and a stopwatch Petri net representation of the tank2 model respectively. Tank1 and tank2 have the same representation with Petri nets.

If the level of tank1 reaches V1L, the calculator orders the use of the electrovalve EV3 represented in Figure 5 by the firing of the transition t14 in the time interval [4,7]. Consequently, the place V1_dec_S is marked. If the electrovalve EV3 is functioning correctly, the transition t15 is fired and the place V1_cr which represents the system reconfiguration is marked.

However, if the electrovalve EV3 undergoes a failure, the suspension of the task EV1_dec_s begins. This is represented by the firing of the task ts in the time interval [0,4]. If the failure duration exceeds this time interval, the system leaves its normal functioning towards a feared (dangerous) state. Consequently, the place E_red1 is marked. The place E_red1 represents the overflow of the tank1: the feared state. But if the failed electrovalve EV3 is repaired in the considered time interval, the system will return to its normal functioning by the resumption of EV1_dec_s which is represented by the transition tr in the time interval [5,7].

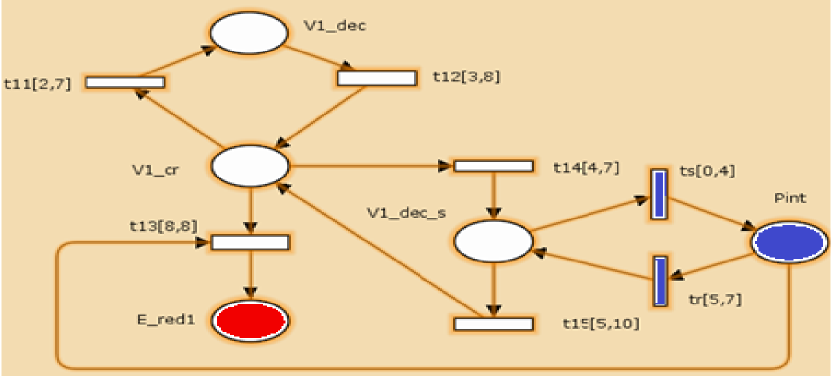

Figure 5. SWPN representation of the tank1 model

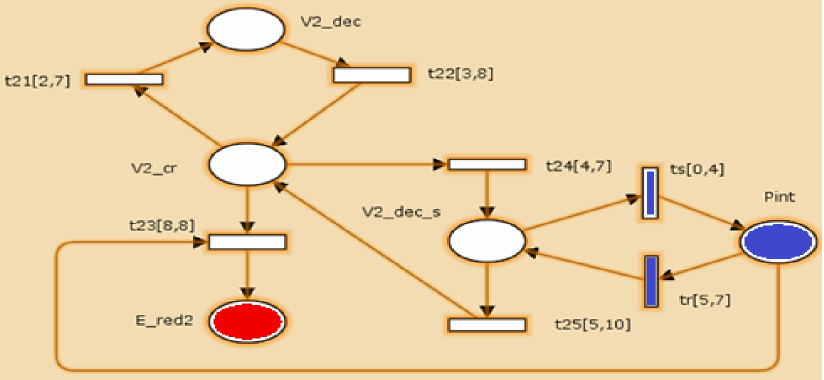

Figure 6. SWPN representation of the tank2 model

4.2.3. Time Petri net representation of the electrovalve model

4.2.3.1. Time Petri net representation of the electrovalves (EV1 and EV2) model

Figure 7 and Figure 8 presented below show a time Petri net representation of the electrovalve EV1 model and a time Petri net representation of the electrovalve EV2 model respectively. The electrovalves EV1 and EV2 have the same representation with Petri nets. In Figure 7, the opening and the closing of the electrovalve EV1 are represented by the transitions t2 and t1 respectively. The failures of the electrovalve EV1 are represented by the firing of the transitions def1_O (EV1 is blocked from closing) and def1_F (EV1 is blocked from opening).

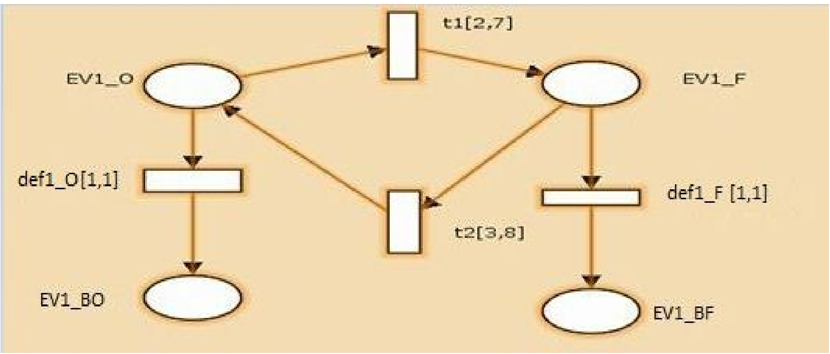

Figure 7. TPN representation of the electrovalve1 model

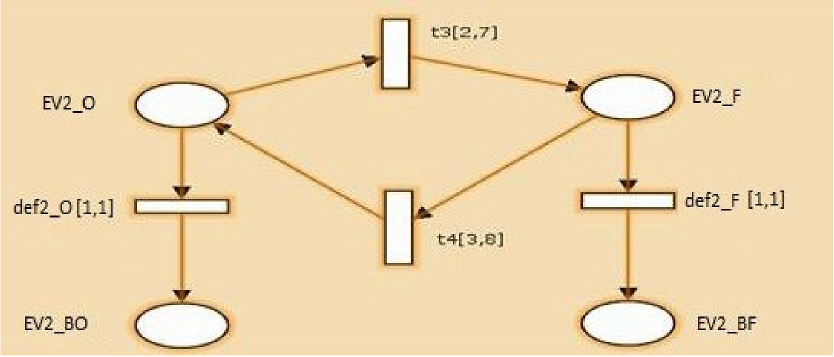

Figure 8. TPN representation of the electrovalve2 model

4.2.3.2. Time Petri net representation of the electrovalve EV3 model

Figure 9 presented below show a time Petri net representation of the electrovalve EV3 model. The place EV3_OK represents the availability of the electrovalve EV3. The transition t14 represents the opening of the electrovalve EV3, when the tank1 level exceeds V1L or the tank2 level exceeds V2L. When a conjunction phase begins and the place V1_cr (in the tank1 model) is marked, the calculator orders the use of the electrovalve EV3 if it is functioning correctly which is represented by the marking of the place EV3_OK. The transition t14 is then fired and the place V1_dec_S is marked (in the tank1 model). Since the electrovalve EV3 is used for the emptying of the tank1 the place EV3_OC1 is marked.

The electrovalve EV3 may undergo a failure which is represented by the firing of the transition def3. In this case, the place EV3_OH is marked and the electrovalve EV3 is out of service.

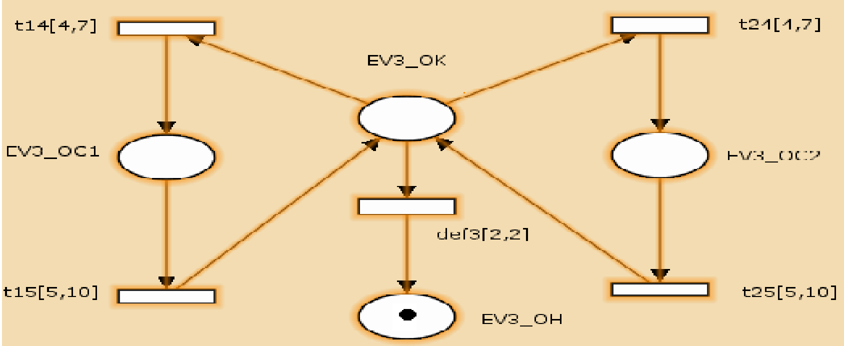

Figure 9. TPN representation of the electrovalve3 model

4.2.4. Stopwatch Petri net representation of the elecrovalve model

4.2.4.1. Stopwatch Petri net representation of the electrovalves (EV1 and EV2) model

Figure 10 and Figure 11 presented below show a stopwatch Petri net representation of the electrovalve EV1 model and a stopwatch Petri net representation of the electrovalve EV2 model respectively. The suspension of task of the electrvalves (EV1 and EV2) is due to the electrovalves failures which are represented by the firing of the transitions def1_O (EV1 blocked from closing) or def1_F (EV1 blocked from opening). The resumption of task is represented by the firing of the transition rep1 (the reparation of EV1).

The suspension of the task EV1_O of the electrovalve EV1 is represented by the firing of the transition def1_F in the time interval [1,1]. The place EV1_BO is then marked and the tank1 level increases and exceeds the limit V1L. If the transition rep1 which represents the resumption of task will be fired in the time interval [4,6], the place EV1_O is then marked. The tank1 begins the disjunction phase by the firing of the transition t11 (in the tank1 model). Consequently, the tank1 level decreases. However, if the failure duration exceeds the considered time interval, the calculator orders the emptying of the tank1 by the use of the electrovalve EV3 if it is available.

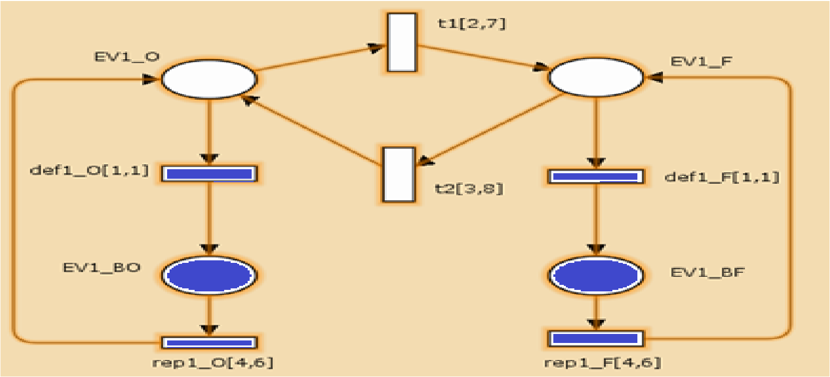

Figure 10. SWPN representation of the electrovalve1 model

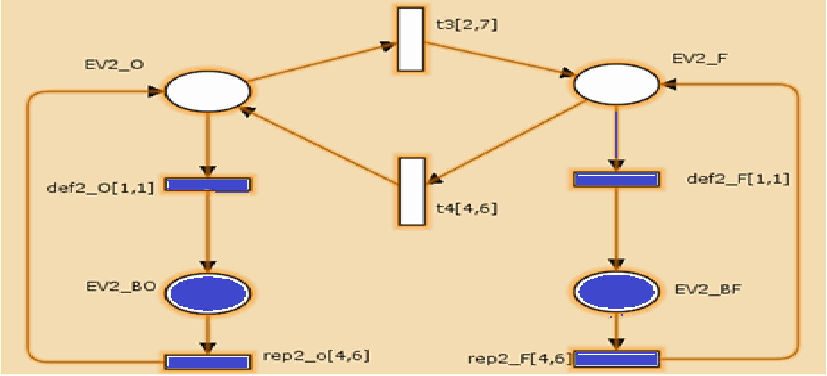

Figure 11. SWPN representation of the electrovalve2 model

4.2.4.2. Stopwatch Petri net representation of the electrovalve EV3 model

Figure 12 presented below show a stopwatch Petri net representation of the electrovalve EV3 model. If the electrovalve EV3 undergoes a failure, the transition def3 is fired in the time interval [0,3] which represents the suspension of the task EV3_OK. In this case EV3 is not available and the place EV3_OH is marked. The firing of the transition rep3 represents the resumption of this task if it does not exceed the time interval [2,4]. The place EV3_OK is then marked and the electrovalve EV3 becomes available, for example, to empty the tank1. However, if the failure duration exceeds the considered time interval, the electrovalve EV3 remain unavailable and the system leaves its normal functioning towards a feared (dangerous) state: the overflow of the tank1. Consequently, the place E_red1 is marked (in the tank1 model).

If the electrovalve EV3 is occupied by the emptying of the second tank**,** the place EV3_OC2 is marked. If the emptying duration exceeds the considered time interval, the transition t25 cannot be fired and the electrovalve EV3 remain unavailable. It is like a failure**,** the transition def3 is then fired and the place EV3_OH is marked. The electrovalve EV3 becomes available at the end of the emptying of the second tank. The system leaves then its normal functioning towards a feared (dangerous) state: the overflow of the tank1. Consequently, the place E_red1 is marked (in the tank1 model).

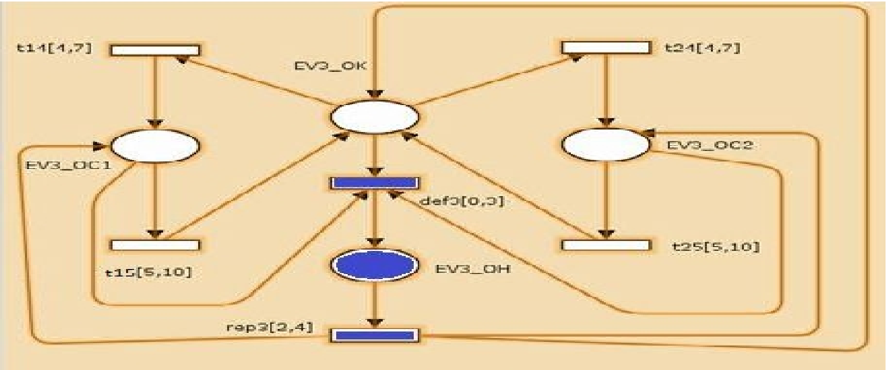

Figure 12. SWPN representation of the electrovalve3 model

The advantage of our approach is that the more detailed system configuration using Stopwatch Petri nets enables us to express temporal behaviors better than time Petri nets model by taking into account the suspension and resumption of tasks. This configuration allows more interactions between the different components of the system and gives more feared scenarios (more dangerous behaviors). These scenarios are not taken into account by the preceding feared scenarios approaches.

5. APPLICATION OF THE METHOD

In this section, we present the application of the new version of the feared scenarios generation algorithm to the tank system regulation. We show the interactions between the different components of this system which lead to the feared state: the overflow of the tank1. We present the four steps of the proposed method using stopwatch Petri nets in order to generate a new feared scenario (we can generate other scenarios). This scenario cannot be generated when the system is modeled by time Petri nets.

Step 1: there are many nominal states of the system such as V1_dec, V1_cr, V1_dec_s and Pint.
Step 2: the target state considered is the feared state: the overflow of the tank1.

Step 3: using the inverted stopwatch Petri net model of the system, the back reasoning starts from the target state: the overflow of the tank1. We go back up through all the possible preceding states, until we reach two normal functioning states: V1_cr and Pint which are conditioner states.
Step 4: the front reasoning begins from the conditioner states: V1_cr and Pint. The marking of the place V1_cr is the cause of the first kind of bifurcation (conflicts between the normal functioning and the feared behavior). The marking of the place Pint which is an input place of the transition t13 is the cause of the second kind of bifurcation (conflicts between a transition which represents the non-resumption of an interruptible transition and a transition which represents the resumption of an interruptible transition).

If the place V1_cr is marked (this marking causes a conflict between the transitions t14 and t13) and the electrovalve EV1 is blocked from closing, the tank1 level increases and exceeds the limit V1L. If the electrovalve EV3 undergoes a failure or is unavailable because it is occupied by the emptying of the tank2, the transition t14 cannot be fired. Consequently, the system leaves its normal functioning towards a feared (dangerous) state: the overflow of the tank1. The place E_red1 is then marked. In this case, we must make a maximal marking enrichment of the place V1_cr in order to avoid the drift towards the feared state which is represented by the firing of the transition t13. The two scenarios which lead the system towards this feared state are represented below by Figure 14 and Figure 15.

If the place Pint is marked (this marking causes a conflict between the transitions tr and t13) and the electrovalve EV1 is blocked from closing, the tank1 level increases and exceeds the limit V1L. If the electrovalve EV3 undergoes a suspension of task, this suspension will be memorized during the front reasoning, if its duration exceeds the specified time interval, the stopwatch value is also memorized. Consequently, the transition tr which represents the resumption of the task cannot be fired and the system leaves its normal functioning towards a feared (dangerous) state: the overflow of the tank1. The place E_red1 is then marked. In this case, we must make a modification of time constraints in order to avoid the drift towards the feared state which is represented by the firing of the transition t13.

Scenarios are represented by a partial order defined by a directed graph (E, A) where the nodes E are a set of transition firings and the arcs A are pairs (t_i, t_j) such that t_i precedes t_j (t_i and t_j are transition firings).

Figure 13 shows this feared scenario which is composed of the following events: the failure of the electrovalve EV1, the failure of the electrovalve EV3 after a task suspension (represented by the firing of the transition ts) and a non-resumption of this task because of non-respect of time constraints, and then, the overflow of the tank1 because of the firing of the transition t13.

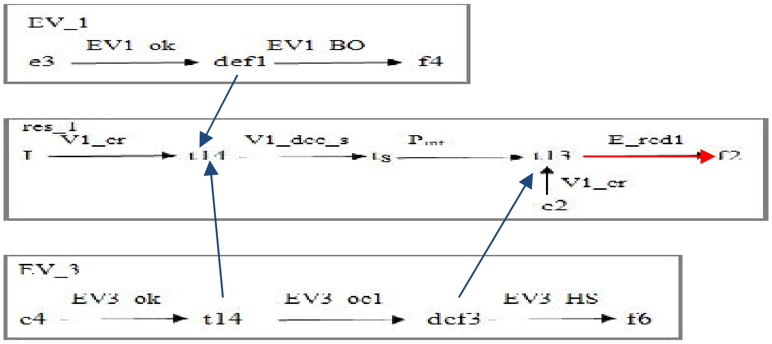

Figure 13. The first feared scenario

Figure 14 shows the second feared scenario which is composed of the following events: the failure of the electrovalve EV1, the failure of the electrovalve EV3, and then the overflow of the tank1. This scenario is also generated when the system is modeled by time Petri nets [3].

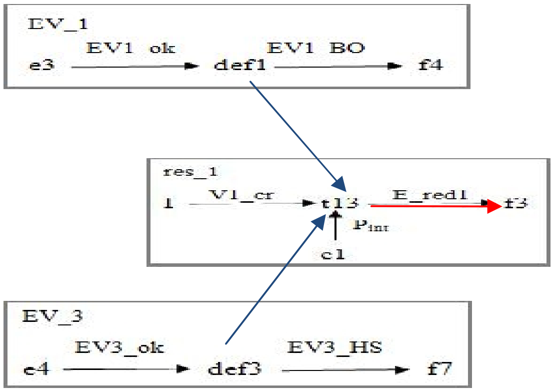

Figure 14. The second feared scenario

Figure 15 shows the third feared scenario which is composed of the following events: the failure of the electrovalve EV2, the use of the electrovalve EV3 by the tank2, and then the overflow of the tank1. This scenario is also generated when the system is modeled by time Petri nets [3].

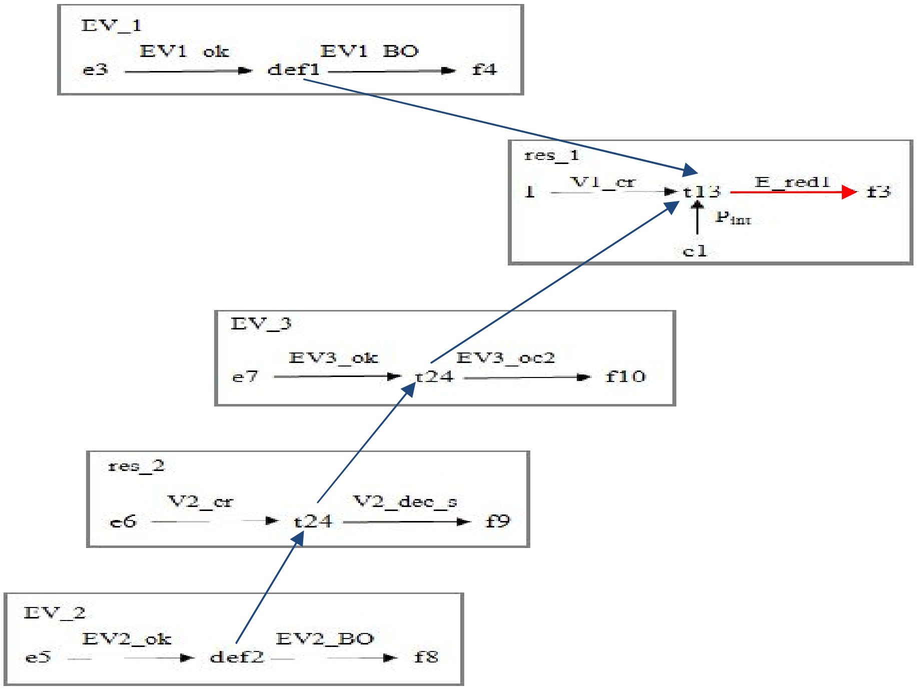

Figure 15. The third feared scenario

6. CONCLUSIONS

In this paper, we have proposed a reliability approach based on the extraction of feared scenarios from a stopwatch Petri net model. The advantage of this method is that the more detailed configuration of the system enabled us to represent the suspension and resumption of task execution and to find new feared scenarios which are generated because of non-respect of time constraints. These scenarios are not taken into account by the preceding feared scenarios approaches when the system is modeled by time Petri nets. The description of feared scenarios enables us to understand the reasons of the drift in order to envisage the necessary reconfigurations to avoid them.

To illustrate the application of the proposed method, we presented a detailed case study: a tank system regulation. This system is modeled using two formalisms: Time Petri Nets (TPN) and Stopwatch Petri Nets (SWPN) in order to show the difference between the results obtained by the application of the old version of the feared scenarios generation algorithm and those obtained by the application of the new proposed version of the feared scenarios generation algorithm.

REFERENCES

[1] Khalfaoui, S, (2003) Méthode de recherche des scénarios redoutés pour l'évaluation de la sûreté de fonctionnement des systèmes mécatroniques du monde automobile. Thesis, Institut National Polytechnique, Toulouse, France.

[2] Medjoudj, M, (2006) Contribution à l'analyse des systèmes pilotés par ordinateurs : extraction de scénarios redoutés et vérification de contraintes temporelles. Thesis, Paul Sabatier University, Toulouse, France.

[3] Sadou, N, (2007) Aide à la conception des systèmes embarqués sûrs de fonctionnement. Thesis, Toulouse III University- Paul Sabatier, France.

[4] Ghenai, A. & Benmohammed, M, (2011) Une Approche Basée sur la Logique TPN-TCTL pour la Conception Sûre des Systèmes Embarqués. In: Congrès International Pluridisciplinaire en Qualité et Sûreté de Fonctionnement, Angers, France.

[5] Magnin, M, (2007) Réseaux de Petri à chronomètres Temps dense et temps discret. Thesis, Nantes University, France.

[6] Allahham, A. & Alla, H, (2007) Réseaux de Petri à chronomètres Post et Pré initialisés. In: 6ème colloque francophone sur la modélisation des systèmes réactifs, Lyon, France.

[7] Magnin, M. & Pierre Molinaro, P & H. Roux, O, (2009) Expressiveness of Petri Nets with Stopwatches. Dense-time part. In: Fundamenta Informaticae.

[8] H. Roux,O. & Lime,D, (2004) Time Petri Nets with Inhibitor Hyperarcs. Formal Semantics and State Space Computation. In: International conference on Applications and Theory of Petri Nets, Bologna, Italy.

[9] Villani, E. & Miyagi, P.E. & Valette, R, (2007) Modelling and analysis of hybrid supervisory systems. A Petri net approach. Springer.

[10] Sadou, N. & Demmou, H. & Pascal, J.C. & Valette, R, (2006) Continuous dynamic abstraction for reliability and safety analysis of hybrid systems. 6th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes, Beijing, China.

AUTHORS

Afifa Ghenai is with the department of computer science, Mentouri University Constantine, Algeria. She obtained her BEng degree from University of Mentouri Constantine, Algeria. She prepares a PhD thesis at LIRE laboratory. Her research domains are embedded systems and formal methods.

Mohamed Youcef Badaoui received his master's degree in computing sciences from Mentouri University Constantine, Algeria in 2011. His research domain is formal methods for verifying real time systems.

Mohamed Benmohammed is a professor at the department of computer science, Mentouri University Constantine, Algeria. Also, he is the head of the AS research group of LIRE laboratory. His main research domain is embedded systems.