International Journal of Computers, Communications & Control
Vol. II (2007), No. 3, pp. 252-268

Extraction of Critical Scenarios in a Railway Level Crossing Control System

Malika Medjoudj, Pascal Yim

Abstract: This paper deals with the safety of the level crossing control system. We propose one way of the safety evaluation witch consist on the extraction of feared scenarios in the Petri net model of the system. We use ESA_PetriNet tool (Extraction Scenarios & Analyzer by Petri Net model) that was developed in the aim of extraction of feared scenarios in computer-controlled systems. These scenarios characterize the sequences of actions leading to dangerous situations. The taking into account of the failures, the temporal constraints and partially the continuous dynamic (by temporal abstraction) of the system makes it possible to respect the order of appearance of the events in the generated scenarios.
Keywords: Critical scenarios, hybrid dynamic, level crossing control system, safety, temporal Petri net

1 Introduction

One way to evaluate the safety [2] of complex system such as a level crossing control system (*lccs*) is the extraction of critical scenarios leading to the feared states. A qualitative analysis method of safety, aiming the extraction of all the critical scenarios from a Petri Net model [5] of computer-controlled systems was developed by [3]. This approach witch is an extension of a method developed by [6] but which operated only on the discrete aspect of the system, takes into account the continuous aspect of the system and the temporal specifications. This approach based on linear logic [7] determines more precisely the exact conditions of the occurrence of the feared event, i.e what has led the system to leave its normal operation and to evolve into the feared state. The originality of this approach is that the order of occurrence of the events is taken into account, and impossible scenarios with respect to continuous dynamics and temporal specifications of the system are eliminated. The automation of all stages of the process has led to the development of ESA_PetriNet tool (Extraction & Scenarios Analyser by PetriNet model) [4] that has been interfaced with TINA tool (Time Petri Net Analyzer) [8].We will use in this paper ESA_PetriNet tool to extract dangerous scenarios from the level crossing benchmark published by [10].

We will present the method of extraction of feared scenarios and the basic of the algorithm in section 2, the level crossing control system in section 3, its Petri Net modelling in section 4, the use of ESA_PetriNet tool to generate the critical scenarios in the section 5 and we will end by a conclusion.

2 Method of extraction of feared scenarios

The application of this method requires the modelling of the system by a time Petri Net model and identifying the places of nominal behaviour. The appropriate Petri net modelling of computer-controlled systems is a Predicate Transitions Differential Stochastic Petri net (PTDS Petri net) as they are generally hybrid (discrete and continuous dynamics) and there safety analysis require taking into account failures. A temporal abstraction is necessary to translate this model to a time Petri net by associating to the transitions a temporal interval of firing corresponding to the time which the system can spend to reach the state in question. A preliminary analysis will refine fields of variables according to various accessible markings by reasoning on the invariants of places. Indeed, the invariants of places determine the possible dynamics, and which other places can be simultaneously marked when a token is present in a given place.

2.1 Principal

The method of extraction of feared scenarios is made up of two steps [3]: a backward reasoning and a forward reasoning. The backward reasoning takes as an initial marking in the reversed Petri net model (the initial Petri net in which all the arcs are reversed), the only target state (feared) and seeks exhaustively all the scenarios making it possible to consume the initial marking (feared state since forward reasoning) and reach a final marking composed only of places associated to the normal operation. The forward reasoning takes as an initial state these places of normal operation in the initial Petri net model. The objective is to locate the junctions between the feared behaviour and the normal operation of the system as well as the conditions implied in these junctions. Thus we have not only the explanation of the dangerous behaviour but also of strategies allowing its avoidance. A significant point of the method is that the context in which occurred the feared event is enriched gradually. The enrichment (of marking) consists on putting tokens in empty places in the Petri net model when it is necessary to make evolve the system and generate scenarios. The invariant of places are used as a mechanism of checking the coherence of the enrichment of marking. Indeed the new tokens added are removed if they do not respect the dynamics of the system.

Each scenario is given in form of a partial order between the events necessary to the appearance of the feared event what differs from a failure tree, which gives a whole of static combinations of the partial states necessary for obtaining the feared state.

2.2 Dealing with continuous dynamics by temporal abstraction

This method takes into account the conditions associated to the firing of certain transitions. These conditions are thresholds involving continuous variables. By temporal approximation of the hybrid dynamics, these thresholds are transformed to durations, which correspond to time that the system puts to reach when the transitions are enabled. From a qualitative point of view, the objective is to determine the firing order of the transition. Thus, when we enrich the marking, we can find situation where two transitions *t1* and *t2* are enabled if only the ordinary Petri net is considered, but whose are such as *t1* will be always fired before *t2* if the temporal abstraction is also considered. In the generation of the scenarios only the firing of *t1* will be considered since that of *t2* before *t1* would be in fact incoherent with the continuous dynamics. This appears in the form of a priority: if *t1* and *t2* are enabled, only the case of *t1*, priority, is examined. The taking into account of these precedence relations coming from the continuous dynamics and not specified by the ordinary Petri net allows to reduce the number of scenarios generated by eliminating a certain number of incoherent scenarios with respect to continuous dynamics.

Let us consider an example. In Figure 1 we suppose that the differential-algebra system associated to the place *P1* guarantees that the variable x is increasing. We associate to the transition *t1* the threshold $x = v1$ and to the transition *t2* the threshold $x = v2$ with $v1 < v2$. Finally, we suppose that when the token arrives in the place *P1* we have always $x < v1$. So, if the place *P3* is marked, the transition *t1* will be fired before *t2* since the threshold associated to *t1* is lower than that of *t2*. In this case we don't consider the scenario associated to the firing of *t2*. On the other hand, if *t3* is already fired for example if we consider that *t1* is a stochastic transition corresponding to a failure (place *P3* empty) and if the place *P2* is marked, *t1* cannot be fired and then *t2* will be fired.

In the example above, finally only one type of scenarios is examined, those for which the transition *t2* is fired after *t3*. So, there is a precedence relation between the firing of *t3*, which empties the place *P3* and that of *t2*, however there is no place connecting *t3* to *t2*. This precedence relation is so, a consequence of continuous dynamics and thresholds associated to transitions *t1* and *t2*. We are talking in this case about indirect precedence relation and about indirect causality. The direct precedence relations and causality are those that are highlighted by the only Petri net, i.e. by the only discrete aspect.

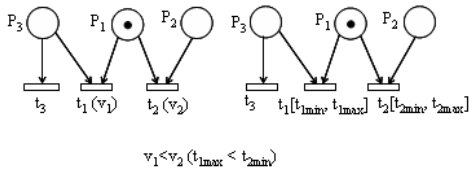

Figure 1: Temporal abstraction and priority due to thresholds of transitions

2.3 ESA_PetriNet tool

ESA_PetriNet tool uses two output files of TINA tool as input files. The first file is a textual description of the Petri net model of the system and the second contains the invariant of places. The indirect precedence between certain transitions firing resulted by the temporal abstraction of continuous dynamics, is expressed in the algorithm in the form of rules of priority (a certain transition is not fired if another is enabled). The transition time interval of TINA tool permits to express this rule of priority. If we take the example of Figure 1, if there is an intersection between the time interval associated to transitions *t1* and *t2*, they will have the same priority of firing and the two scenarios will be generated. We note that only one execution of the algorithm generates automatically several scenarios. All the possible and coherent scenarios with respect to the continuous dynamics and the temporal constraints of the system are generated.

3 Level crossing control system case study

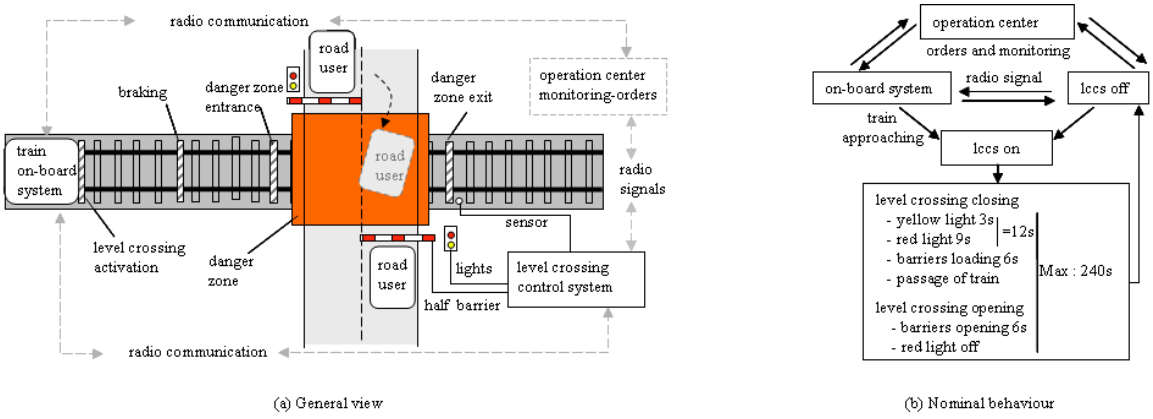

Figure 2: Railway level crossing

3.1 General description

This case study concerns a decentralized radio-based railway level crossing control system taken from a realistic specification of a new radio-based train control system, which has been developed for the German Railways. It is presented by [10] and studied by [9] using a transformation of a p-time Petri net model of the system to automata in the purpose of avoiding forbidden states. This modelling is of a high level of abstraction and does not take into account the failures of the system.

Although, simplification has been made in the presentation of this example, it remain especially interesting as it is well known by the railway specialists, takes into account software and hardware specification, hybrid dynamic and temporal constraints. Our aim is a whole modelling of the system

using a Petri nets model by taking into accounts the hybrid dynamic, temporal constraints and failures. Then, applying the method described above to extract critical scenarios.

3.2 Composition of the system and specification

The radio-based level crossing control system is used in an intersection area between a single track railway line and a road as illustrated in Figure 2a. To avoid collision, trains and road traffic must not enter at the same time this crossing zone called danger zone. The level crossing is controlled by means of signals radio communication between a train-borne control system (on-board system), a level crossing control system and an operation centre which supervises interactions between the two preceding components. It is important to note that transmission times on the network may vary and radio telegrams may be lost.

The railway crossing is equipped with half barriers, a red and a yellow road traffic light. Road users shall stop at the level crossing if possible when the yellow light is shown and must stop when the red light is shown as the level crossing is closed for road users in this case. The yellow light and the red light never must be shown together and when both are off the danger zone can be crossed by road users. The traffic lights and barriers at the level crossing are controlled by the level crossing control system which will be activated (turn on) with the approach of a train to the level crossing. When the level crossing control system is activated, it carries out a sequence of actions at a specific timing to ensure a safely closing of the crossing and the danger zone to be free of road traffic. First, the yellow light is switched on, then after 3 seconds it is switched off and the red light is switched on. After 9 seconds the barriers are started to be lowered within a maximum time of 6 seconds. If the barriers have been completely lowered within this time, the level crossing control system signals the safe state of the level crossing and the train can cross it. When the train has completely passed the danger zone, the level crossing may be opened for road traffic. In the level crossing opening phase, the barriers are first opened then the red traffic light and the level crossing control system are switched off.

The half barriers are used to block the entry lane on either side of the level crossing. As there are no barriers for the exit lanes, imprudent road users may enter the crossing area on the opposite lane if the closure time of the level crossing exceeds 240 seconds. A general view of the normal operating of the level crossing control system is given in Figure 2b. The train is equipped on board by a route map which contains the positions of danger points at level crossings and provides information for the train (lineside equipment or signal staff) when or where to send an activation order to the corresponding level crossing control system. The train on-board system sends so a radio message to the level crossing control system in order to close the level crossing in time and let the train pass through without any delay or braking action. It will also set a breaking curve for speed supervision making the train stop at the danger point in a failure situation. The level crossing control system acknowledges receipt of the activation order to the train. After receipt of the acknowledgement the on-board system waits the necessary time for the closing of the level crossing, then sends a status request to the level crossing control system. If the level crossing is in a safe state it will be reported to the train which allows cancelling the breaking curve and safely pass over the level crossing. The vehicle sensor at the rear of the level crossing will be triggered allowing the opening of the level crossing.

3.3 Possible Failures

A main cause of failures is the malfunctioning of sensors or actuators. The main physical structures, communication systems and the control systems themselves may be failed. Failure may occur at any time. Defective devices will be repaired after some time but will not take place when a train approaching or passing the level crossing in case of non recoverable failure. In this case study, only a limited number of failures are taken into account:

- Failure of the yellow or the red traffic light
- Failure of barriers (actuators)
- Failure of vehicle sensor
- The delay or loss of telegrams on the radio network

The traffic lights and the vehicle sensor are constantly supervised and defect is immediately reported to the level crossing control system. Failure of the barriers can only be detected by time-out when barriers fail to reach upper or lower end position in time or at all.

3.4 Behaviour of the control system under failure

The level crossing control system detects the occurrence and repair of failures of traffic lights and vehicle sensor and immediately reports them as an event to the operations centre. Train operation is not suspended on the affected track section until repair.

When the train sends a status request, if in the sequel it does not receive the status report with the safe state of the level crossing before entering its breaking curve the on-board system will apply the breaks until the status report will be received or the train has come to a stand still. If the status report is received before stand still, breaks are released and train can continue its run. If not a request is prompted on the driver's display to make sure that the level crossing can be passed safely and to confirm the safe state on the display. If meanwhile the status has been received the message is cancelled from the display, the break are released and the driver does not need to confirm anymore. Otherwise the driver has to confirm the safe state of the level crossing in order to release the breaks and continue its run.

The train supervises a maximum arrival time of 240 seconds to avoid long waiting times of road users. If the train detects that it cannot arrive at the level crossing within a specified time and still is able to stop before the danger point it cancels the activation order by sending a deactivated order to the level crossing. In this situation the train discards any information received from the level crossing and supervises a breaking curve ending at the danger point. The level crossing will be opened upon receipt of the deactivation order. The driver has to confirm as described above the safe state before passing unclosed level crossing.

The level crossing control system will not be activated if the red traffic lights or the vehicle sensor are defective and it will not send an acknowledgment to the train. If the level crossing control system has been activated, a minimum green time is considered since the last deactivation of the level crossing before switching on the yellow light for 3 seconds. If the yellow traffic light becomes defective either before or during the yellow light period, the traffic lights are switched to red and the red light period of 9 seconds is extended correspondingly by the messing time of the yellow light period. If the red traffic light fails after activation of the level crossing control the closing procedure has to be cancelled unless the barriers have yet begun to be lowered. The failure state of the level crossing must be reported if the barriers fail to be completely lowered within a maximum duration of 6 seconds or if in the meantime the red traffic light has become defective. The current status of the level crossing will be reported to the train upon request.

If the vehicle sensor becomes defective the level crossing control system can not be deactivated anymore by passing train. Consequently the barriers remain lowered and the red traffic light remains switched on. However, the level crossing control system supervises a maximum closure time starting from the red light be switched on. The exceeding of the maximum closure time will be reported to the operation centre by the level crossing control system. The operation centre finds out, whether the train has yet passed the level crossing or not. In the first case, the operations centre sends a deactivation order to the level crossing. Otherwise the train is still approaching or just running on the level crossing and the rules for late arrival at the level crossing apply as described above.

3.5 Feared events

There are many feared events in the system, but we will interest only to the catastrophic one: the collision, it means the presence of a train and a road user in the danger zone at the same time.

4 Modelling

Petri nets have been used with success as a formal model for traffic signal control [11], urban traffic control [12], and level crossing control system [13] aiming security.

This section deals with the modeling of the level crossing control system by a t-time Petri net model (temporal intervals associated to transitions). Although the appropriate abstraction of certain dynamics of the system is a pt-arc-time Petri net (temporal intervals associated to arcs related places to transitions) or a P-time Petri net (temporal intervals associated to places) we have chosen the t-time Petri net model as the principal of the ESA-PetriNet tool is based on the priority of firing of conflictive transitions.

4.1 General view

In the general view given in Figure 3a, *msgi* represent radio messages. The message *msg1* is sent by the train to the *lccs* to switch on when the on-board system detects the approaching of a level crossing. The message *msg2* represents the receipt acknowledgement of the activation order. The message *msg3* corresponds to the status request of the level crossing and the message *msg4* represents the safe state of the level crossing reported to the train.

Note that, transmission times on the radio network may vary and messages may be lost as represented in Figure 3b. The radio message is represented by the place *msgi*. The time interval *[dmi, dMi]* associated to the out put transition of place *msgi* means that the transmission time may vary between the minimal value *dmi* and the maximum one *dMi*. According to the radio message and the crossing state, the train will pass with out braking, with braking, come to a stand still or stop.

The *dely1* corresponds to the maximum closure time of 240 seconds supervised by the level crossing control system starting from the red lights be switched on. Crossing the danger zone by the train and the road user are respectively represented by *dz1* and *dz2*. The message *msg7* corresponds to the deactivation order of the opening of the level crossing sent by the train when it detects that it can not arrive at the level crossing within the maximum supervised arrival time of 240 seconds.

In this paper, we will interest to the feared scenario corresponding to the presence of a train and a road user in the danger zone at the same time (collision). This is represented by the Petri net model of the Figure 3c, where the transition *E_fail* representing the feared event can be fired only when both places *dz1* (presence of a train in the danger zone) and *dz2* (presence of a road user in the danger zone) are marked. Place *S_fail* represents the feared state (collision).

Figure 4 represents a general view of different radio messages exchanged between the on-board system and the level crossing control system. To simplify the case study, we have not presented the radio messages exchanged with the operations centre like the failure and repair of different devices.

4.2 Petri net model of the level crossing control system

A detailed view of the *lccs* is given in Figure 5. Note that messages *msg1* and *msg2* are the same as in Figure 3. Places *lccs_off* and *lccs_on1* correspond respectively to the deactivation and activation mode of the level crossing control system. Transition *on1* will be fired after reception of the activation order *msg1*, to switch to the activated mode if the vehicle sensor (place *s3_ok*) and the red traffic light (place *red_off*) are not defective. The level crossing control system will be deactivated if the place *lccs_on2* or *lccs_on3* is marked. Place *lccs_on2* will be marked when the barriers are opened after closure. Place

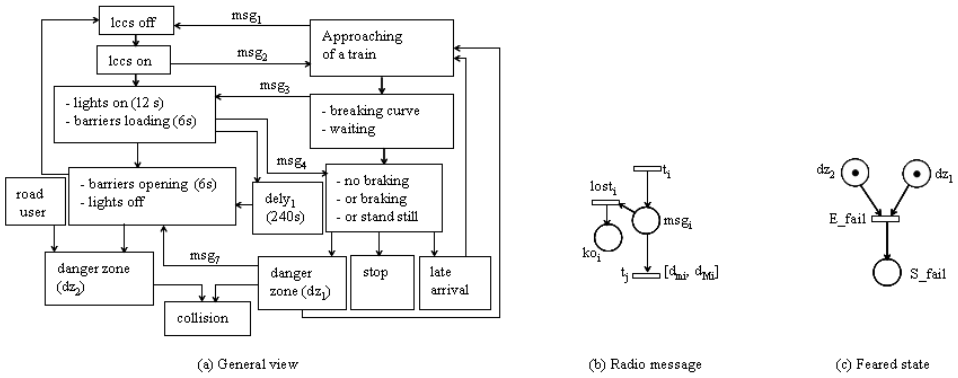

Figure 3: General view of the model

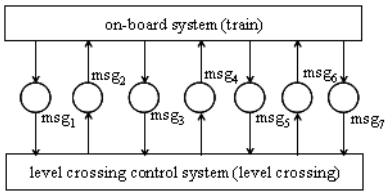

Figure 4: General view of radio messages exchanges

lccs_on3 corresponds to the cancelling of the closing procedure if the red traffic light fails after the activation of the level crossing control system. The green time passed since the last deactivation of the level crossing will be described in section 4.7

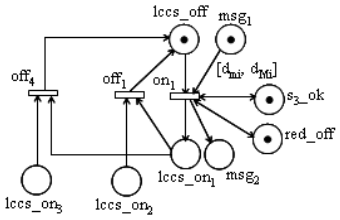

Figure 5: Petri net model of the level crossing control system

4.3 Petri net model of the yellow light

Place *yell_off1* in Figure 6a represents the *mode off* of the yellow light. It will be switched to the activated mode (place *yell_on*) when the level crossing is activated (place *lccs_on1* marked). After 3 seconds of the activation of the yellow light, it will be deactivated (marking of places *yell_off1* and *yell_off*2). The yellow light can fail in the deactivated mode (*yell_ko1*) or in the activated mode (*yell_ko2*). Figure 6b represents the Petri net model of failure and repair of the devices that may be faired in the system (traffic lights, vehicle sensor and barriers). Failure and repair are represented respectively by the stochastic transition *faili* and *repi*. While failure may occur at any time, repair will not take place when there is a train approaching or passing the level crossing. This is represented by the minimal value of reparation *dri*.

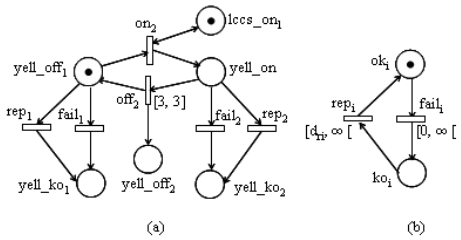

Figure 6: Petri net model of the yellow light

4.4 Petri net model of the red light

The model is similar to the yellow light. The red traffic light can be in *mode off* (place *red_off*), *mode on* (place *red_on*), fail before activation (place *red_ko1*) or after activation (place *red_ko2*). We note three cases of the activation mode of the red light represented in Figure 7 according to the time activation of the yellow light. In case (a), the yellow light was activated for 3 seconds before the lights traffic switch to the red. In this case the place *yell_off2* is marked and transition on3 can be fired to switch to the activated mode of the red traffic for 9 seconds. This delay is represented by the time interval *[9, 9]* related to the transition *cls1* that corresponds to the order of lowering barriers. The place *dely1* is the same as described in Figure 4. The red traffic light can be deactivated when the barrier will be completely opened represented by the place *br4*. Place *lccs_on2* is the same as described in Figure 5. In case of Figure 7b, the yellow traffic light becomes defective before the yellow light period (place *yell_ko1*). In this case the red traffic light period of 9 seconds is extended to 12 seconds to take into account the yellow light period. This is represented by the time interval *[12, 12]* attached to the transition *cls2* that corresponds to the order of lowering barriers. In case (c), the yellow traffic light becomes defective during the yellow light period (place *yell_ko2*). In this case, the red light period of 9 seconds is extended correspondingly to the missing time of the yellow light period. This is represented by the time interval *]9, 12[* associated to the transition *cls3*. Transitions and places concerning the barriers will be described in section 4.6.

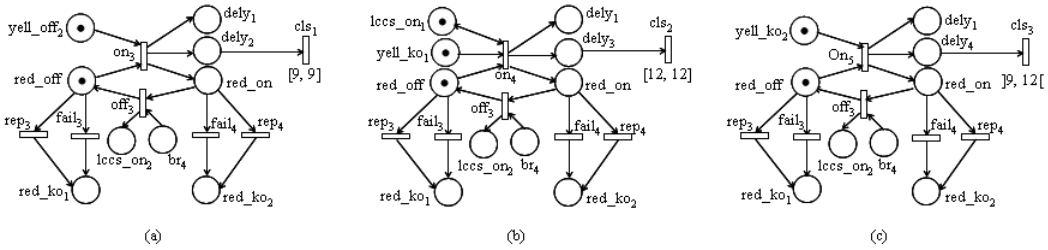

Figure 7: Petri net model of the red light

4.5 Petri net model of sensors

The system contains three sensors (*si*): a sensor for the barriers loading, a sensor for the barriers closing and a vehicle sensor. The Petri net model of a sensor *si* is similar to the model given in Figure 6b. As described in this figure, a sensor *si* can be defective and repaired by firing transition *repi*.

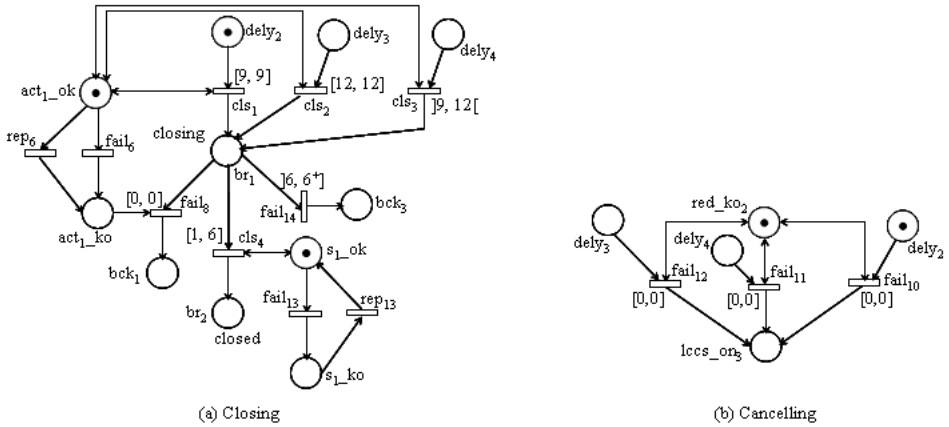

Figure 8: Petri net model of barriers closing and cancelling of the closing

4.6 Petri net model of barriers (actuators)

Closing

To simplify the Petri net model, we assume that Figure 8a represents the closing of the two half barriers which are actuated by an actuator for opening (place *act1_ok*). Note that places *dely2*, *dely3* and *dely4* are the same as in Figure 7. The place *br1* represents the continuous dynamic of the closing barriers. Its temporal abstraction is represented by the temporal interval *[1, 6]* attached to the transition *cls4* as the maximum closure time is 6 seconds and we suppose that the minimum closure duration is 1 second. If the opening actuator fails before the barriers have completely closed (*act1_ko* marked), the immediate transition *fail8* will be fired and the dynamic of place *br1* will be interrupted. This corresponds to the blocking of the barriers in opening represented by the marking of the place *bck1*. If the sensor that detects that the door is closed is defective (marking of place *s1_ko*), the transition *cls4* can not be fired and the level crossing is considered in a failure state. This is represented by the firing of transition *fail14* in the temporal interval *]6, 6+]*.

Figure 8b represents the cancelling of the closing procedure if the red traffic light fails after the activation of the level crossing control system before the barriers begun to be lowered. This is represented by firing the immediate transitions *fail10*, *fail11* or *fail12* according to the activation mode of the red light represented in Figure 7. Place *lccs_on3* corresponding to the order of deactivation of the level crossing will be marked.

Opening

The Petri net model is similar to the Petri net model of closing. The dynamic of the barriers opening is determined by the position of the actuator for opening (place *act2_ok*). The dynamic of the opening is represented by the temporal interval *[1, 6]*. This dynamic can be interrupted if the actuator fails before the end of the opening procedure. In this case the immediate transition *fail9* will be fired and the place *bck2* corresponding to blocking on opening will be marked. If the sensor of opening is defected (place *s2_ko* marked) the transition *fail15* will be fired after 6 seconds. We note four cases for barriers opening represented in Figure 9. Case (a) corresponds to the nominal behaviour. In this case, the vehicle sensor is not defective (place *s3_ok*) and the train has completely passed the danger zone in time (marking of the place *trn12* as it will be detailed in section 4.7). The maximum closure time is represented by the temporal interval *[0, 240]* associated to the transition *opn1*. In case (c), the train detects that it can not arrive to the level crossing in time and it sends a deactivation order to open the level crossing. This is represented by the message *msg7*. In case (b) and (d) the vehicle sensor is defective and the level crossing

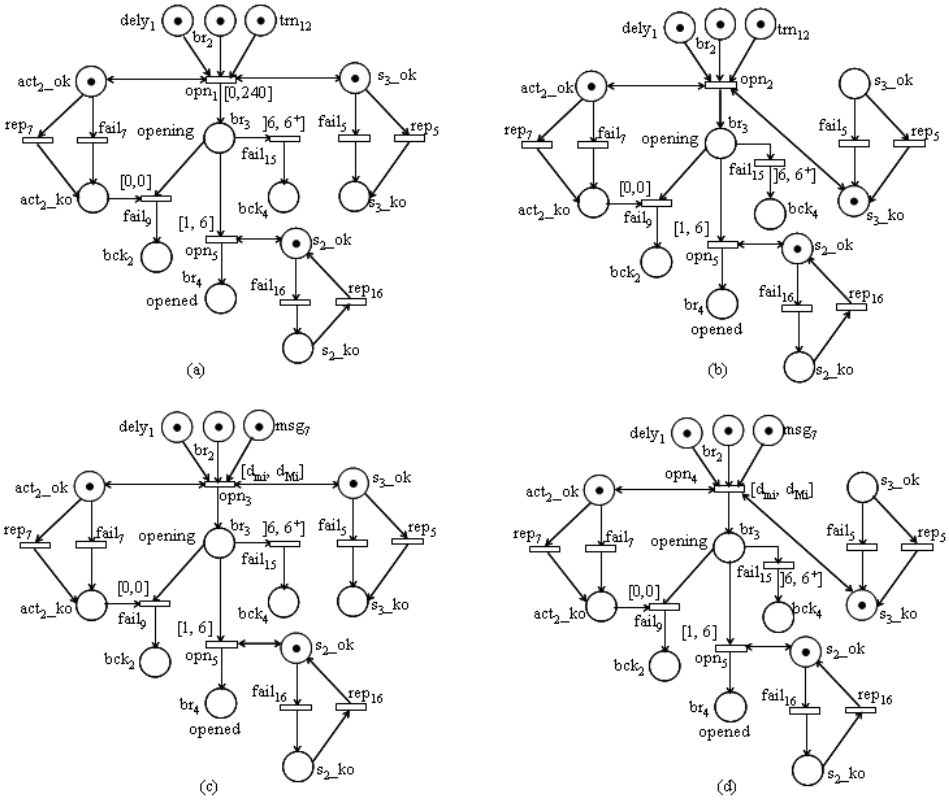

Figure 9: Petri net model of the barriers opening

can not be opened by passing of the train. In this case, the exceeding of the maximum closure time is reported to the operations centre that finds out wither the train has passed the level crossing or is still approaching. Accordingly the operations centre sends a deactivation order (b) in the first case and (d) in the second case.

4.7 Petri net model of the train

Nominal operating and late arrival

The detailed Petri net model of the train is given in Figure 10. Place *trn1* (Figure 10a) corresponds to the approaching of the train a level crossing. When the train on-board system detects this approaching, it sends a radio message *msg1* to the level crossing control system to switch on by firing the transition *tr1*. Place *bc* represents the setting of a breaking curve for speed supervision to make the train stop at the potential danger point in a failure situation. After receipt of the acknowledgement (place *msg2*), the on-board system waits an appropriate time (18 seconds) for the level crossing to be closed and sends the statute request (place *msg3*) to the level crossing control system. The level crossing is reported to the train to be in a safe state (place *msg4*) if the barriers are completely lowered (place *br2*) and the red traffic light is in the activated mode (place *red_on*). After reception of the safe state of the level crossing, the train cancels the breaking curve (place *bc*) and passes the level crossing with out braking. This is represented by firing transitions *tdz1* and *tr6*. The continuous dynamic of the train is represented by the temporal interval *[dti, dTi]* attached to transitions *tri* and *tdz1*. This means that the tokens has to remain in the input places of this transitions at least *dti* and at most *dTi*. Place *trn7* and *trn12* represent the train out of the danger zone. Transition *tr7* can be fired after a green time duration *dg* to specify a non-finite behaviour of the track. Transition *tr14* is fired if the train detects that it can not arrive at the

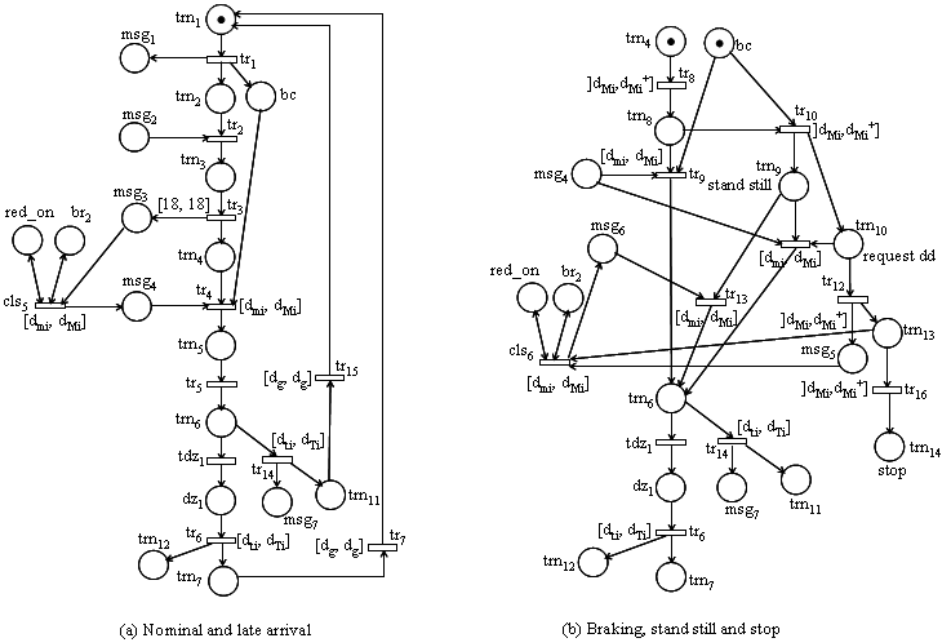

Figure 10: Petri net model of the train

level crossing within the maximum supervised arrival time of 240 seconds (late arrival) and still is able to stop before the danger point. It sends a deactivation order to the level crossing (place *msg7*) and discards any information received so far from the level crossing and supervises a breaking curve (firing transition *tr15* after a green time duration *dg*).

Braking and stand still or stop

Figure 10b represents the case in which the train does not receive the status report with the safe state of the level crossing before entering its breaking curve. Note that places *trn4*, *bc*, *msg4*, *red_on*, *br2*, *trn6*, *trn12*, *dz1*, *msg7* and *trn11* are the same as in Figure 10a. The temporal interval *]dMi, dMi+]* represents the fact that the train does not receive the status report before entering its breaking curve. In this case the on-board system apply the breaks (place *trn8*) until the status report will be received. The transition *tr9* will be fired to release the breaks and continue the run if the status report is received before a stand still (place *trn9*). Place *trn10* represents the request prompted on the driver's display to make sure that the level crossing can be passed safely. Transition *tr11* is fired if meanwhile the status report has been received. Otherwise transition *tr12* will be fired to confirm the safe state by sending the message *msg5*. If the level crossing is in its safe state the transition (place *msg6*), the transition *tr13* will be fired otherwise the train will stop (place *trn14*).

4.8 Petri net model of the road user

Places *road_user* and *dz2* in Figure 11 represent respectively the road user in the entrance of the danger zone and crossing the danger zone. The road user may pass the level crossing only if the red traffic light is not in its activated mode (place *red_on* is not marked) or the level crossing is still open. It means that road users may pass when the red traffic light is off (place *red_off*) or in its defective mode (place *red_ko1* or *red_ko2*) even if the half barriers are lowered as they can pass in the opposite lane or when the half barriers are not yet lowered (place *dely2*, *dely3* or *dely4* marked). To simplify, note that we are focussing on the red traffic light as the yellow traffic light is included in this cases: the yellow traffic

light can be activated when the red traffic light is in its deativated mode or defected before activation (*red_ko1*). the failure of the yellow light is also taken into account as in this case place *dely3* or *dely4* will be marked. The transition *usr* represents the non-finite behaviour of the road users.

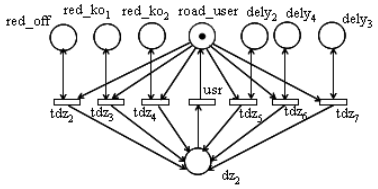

Figure 11: Petri net model of the road user

4.9 The whole Petri net model of the system

Places labelled with "*N*" in Figure 12, modelling the whole system, represent normal operating and transitions labelled with "*F*" will be added to forbidden transitions and can not be fired. This transitions concern repair and non-finite behaviour as repair of defective devices will not take place when there is a train approaching or passing the level crossing and we are interesting in this paper only to one round.

We will seek the feared scenarios corresponding to the presence of both train and road user in the danger zone, i.e. all the scenarios which lead to the marking of the place *S_fail*.

5 Extraction of feared scenarios

A general view of ESA_PetriNet and TINA tools is given in Figure 13. To use ESA_PetriNet, we first edit the Petri net model of the system on the graphic editor of TINA tool to generate two input files: a descriptive file of the Petri net model and a file containing the invariant of places. Generated scenarios can be illustrated in the form of a precedence graph. ESA_PetriNet generates a total of 196 scenarios (nominal and feared) in which 88 are feared. Note that the actual version of ESA_PetriNet generates non minimal scenarios, so most of the generated scenarios are redundant. This explains the important number of the scenarios generated. Note also that this version of ESA_PetriNet support continuous dynamics and temporal constraints and an important number of incoherent scenarios are yet eliminated. We have chosen these parameters: *dmi = 0, dMi = 4, dti = 1, dTi = dg = 10, dri= 250*

The 88 feared scenarios (collision) correspond to the following situations:

1) Crossing of both the road user (*tdz4*) and the train (*tdz1*) the danger zone when the red traffic light fails after activation (*fail4*). In this case, just after the train has received the safe state of the level crossing, the red traffic light fails and the road user passes at the same time the danger zone thinking that the train has already passed. We note three categories of scenarios according to the way of the train crossing.

- The train is crossing without braking: *sc10, sc13, sc19, sc22, sc16, sc25, sc30, sc33, sc36, sc41, sc46, sc53, sc61, sc65, sc69, sc76, sc80, sc84, sc91, sc95, sc102, sc108, sc112, sc116, sc123, sc127, sc131, sc138, sc142, sc149*. These scenarios are represented by *sc1a: {tr4, tr5, tdz1, fail4, tdz4, E_fail}*.

- The train is crossing with braking before stand still: *sc50, sc57, sc62, sc66, sc73, sc77, sc81, sc88, sc92, sc99, sc104, sc109, sc113, sc120, sc124, sc128, sc135, sc139, sc146, sc154, sc157, sc162, sc165, sc168, sc173, sc176, sc181*. These scenarios are represented by *sc1b: {tr8, tr9, tdz1, fail4, tdz4, E_fail}*.

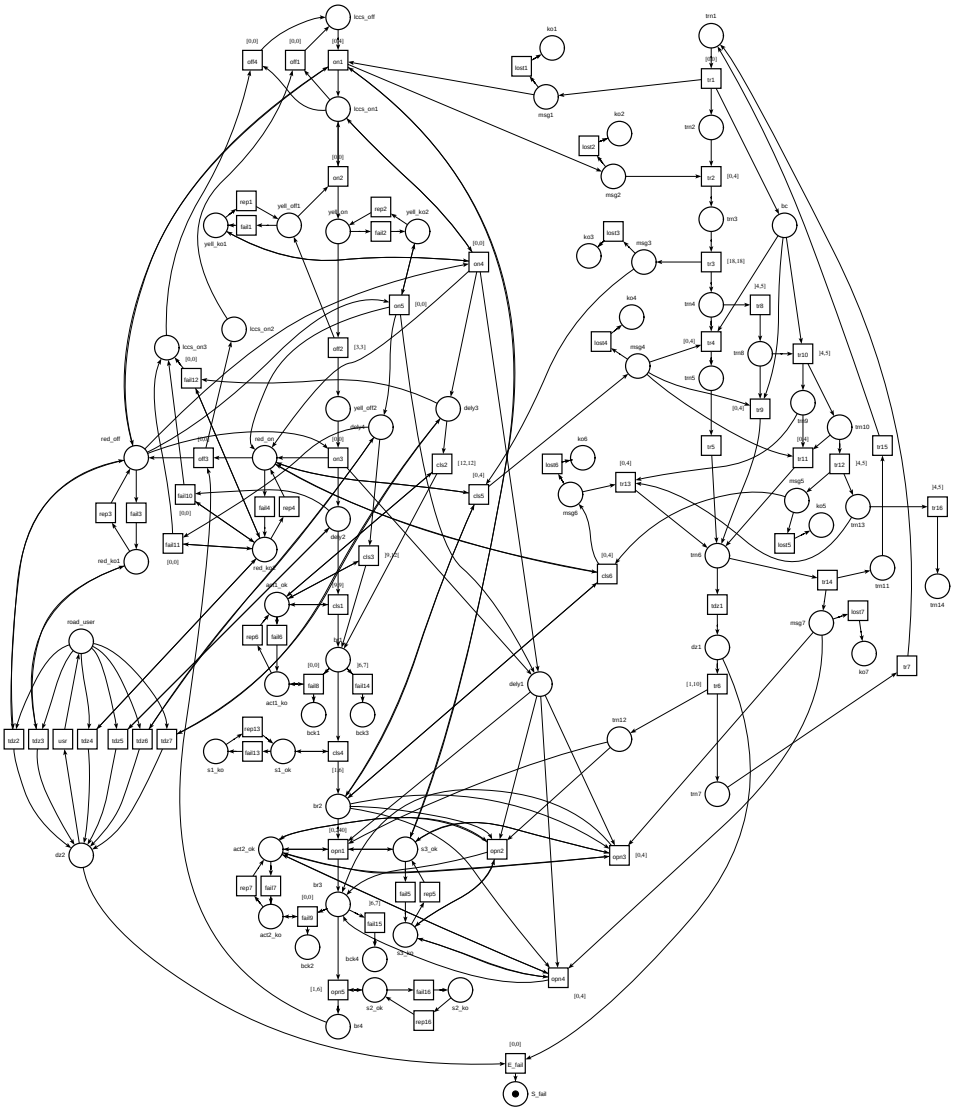

Figure 12: Whole Petri net model of the system

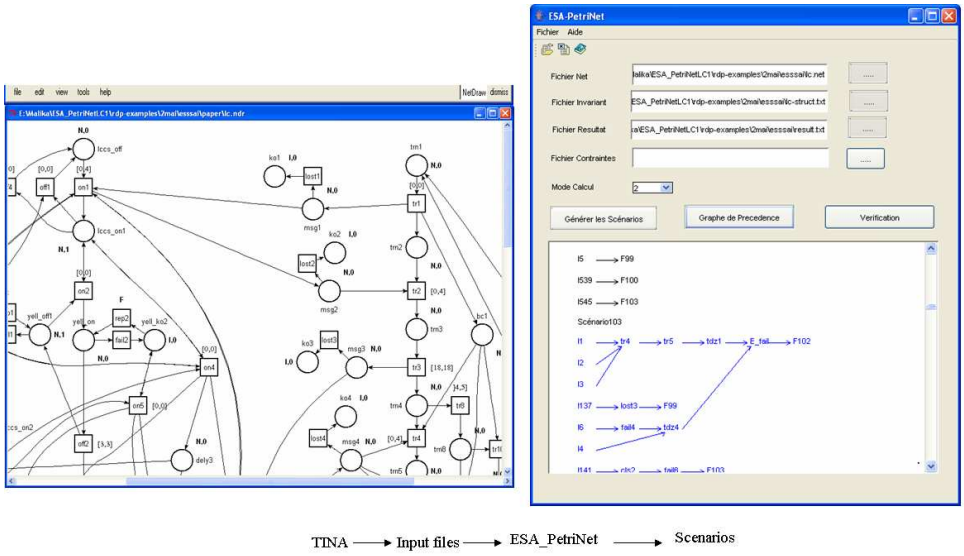

Figure 13: Screen shots of TINA and ESA_PetriNet tools

- The train is crossing after stand still: *sc54, sc71, sc86, sc172, sc180, sc161, sc97, sc118, sc133, sc144*. These scenarios are represented by *sc1c: {lost4, tr13, tdz1, fail4, tdz4, E_fail}*.

The precedence graph of these three scenarios is given in Figure 14. *Ii* and *Fi* represent respectively initial and final events.

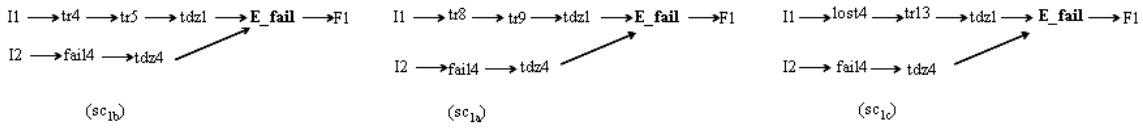

Figure 14: Precedence graph of the scenarios

2) Crossing of the road user the danger zone before the barriers to be lowered (*tdz5*, *tdz6* or *tdz7*) then, crossing of the train (*tdz1*). In this situation, the road user may be slow down stopped or break down on the danger zone. What often arrives on this zone, in general difficult to cross compared to the normal road as attested by the statistics of this field. We note three categories of scenarios according to the mode of activation of the red traffic light and each category contain deferent scenarios according to the way of train crossing.

- In the case of the activation of the red traffic light after the yellow light period (place dely2 marked), we find the scenario *sc21a: {tr4, tr5, tdz5, tdz1, E_fail}*, *sc21b: {tdz5, tr8, tr9, tdz1, E_fail}* and *sc21c: {lost4, tr13, tdz5, tdz1, E_fail}*. The scenario *sc21a* corresponding to the crossing of the train without braking regroups scenarios *sc3, sc5* and *sc11*. The scenario *sc21b* corresponding to the crossing of the train with braking before stand still regroups scenarios *sc23, sc28*, and *sc44*. The scenario *sc21c* representing to the crossing of the train after stand still corresponds to the scenario *sc37*.

- In the case of the activation of the red traffic light after the yellow traffic light become defective in its activated mode (place *dely4* marked), we find the scenario *sc22a: {tr4, tr5, tdz6, tdz1, E_fail}*, *sc22b: {tdz6, tr8, tr9, tdz1, E_fail}* and *sc22c: {lost4, tr13, tdz6, tdz1, E_fail}*. The scenario

sc22a corresponding to the crossing of the train without braking regroups the scenarios *sc1, sc2* and *sc7*. The scenario *sc22b* corresponding to the crossing of the train with braking before stand still regroups the scenario *sc17, sc20* and *sc34*.The scenarios *sc22c* representing the crossing of the train after stand still corresponds to the scenario *sc26*.

- In the case of the activation of the red traffic light when the traffic light become defective in its deactivated mode (place *dely3* marked), we find the scenario *sc23a: {tr4, tr5, tdz7, dz1, E_fail}*, *sc23b: {tdz7, tr8, tr9, tdz1, E_fail}* and *sc23c: {lost4, tr13, tdz7, tdz1, E_fail}*. The scenario *sc23a* corresponding to the crossing of the train without braking regroups the scenarios *sc6, sc9* and *sc14*. The scenario *sc23b* corresponding to the crossing of the train with braking before stand still regroups the scenario *sc31, sc39* and *sc51*.The scenarios *sc23c* representing the crossing of the train after stand still corresponds to the scenario *sc47*.

To facilitate the identification of the feared scenarios among the scenarios of normal operating, ESA_PetriNet tool illustrates them with a different colour.

6 Summary and Conclusions

Two objectives have been reached in this paper. The first is a whole modelling of the level crossing by a temporal Petri net model. The second is the extraction of the critical scenarios using ESA_PetriNet tool. The analysis of these scenarios permitted to propose a solution to improve the safety of the level crossing. This simplest solution consists on the importance of adding a sensor to allow the detection of road users by the train in the danger zone. Among the perspectives of this work: the quantification of these scenarios by a Monte Carlo simulation [1] that has been implemented in ESA_PetriNet, checking different temporal constraints and taking into account the minimality of the scenarios to eliminate the unnecessary events and the redundancies. These analyses can be extended to a level crossing used in the intersection area between a multiple track railway line and a road.

Bibliography

[1] Kalos, M.H., Whitlock, P.A., *Mont Carlo methods*, Vol. 1: basics, John Wiley and Sons, New York, 1986.

[2] Laprie, J.C., *Dependability: basic concepts and terminology*, Vol. 5, Springer, 1992.

[3] Medjoudj, M., Khalfaoui, S., Demmou, H., Valette, R., "A method for deriving feared scenarios in hybrid systems," *Probabilistic Safety Assessment and Management (PSAM7-ESREL04)*, Berlin, Germany, 14-18 June 2004.

[4] Medjoudj, M., Demmou, H., Valette, R., "ESA_PetriNet tool : Extraction Scenarios & Analyzer by Petri Net model : Application to the extraction of feared scenarios in a landing gear system," *European Simulation and Modeling Conference (ESM2006)*, LAAS, Toulouse, France, pp. 375-382, 23-25 October 2006.

[5] Murata, T., "Petri nets: Propreties, analysis and applications," *IEEE Proc*, Vol. 77, pp. 541-580, April 1989.

[6] Demmou, H., Khalfaoui, S., Riviere, N., Valette, R., "Extracting critical scenarios from a Petri net model using linear logic," *Journal Européen des Systèmes Automatisés (APII-JESA)*, Vol. 36, N7, pp. 987-999, 2002

[7] Girard, J.Y., "Linear Logic," *Theoretical Computer Science*, Vol. 50, pp. 1-102, 1987.

[8] Berthomieu, B., Ribet, P.O., Vernadat, F., "The tool TINA - Construction of abstract state spaces for Petri nets and time Petri nets," *International Journal of Production Research*, Vol. 42, N14, pp.2741-2756, 15 July 2004.

[9] Collart-Dutilleul, S., Deffossez, F., Bon, P., "Safety requirements and p-time Petri nets: a level crossing case study," *IMACS-IEEE Multiconference on Computational Engineering in Systems Applications*, pp. 1118-1123, oct 2006.

[10] Jansen, L., Schnieder, E., "Traffic Control Systems Case Study: Problem Description and a Note on Domain-based Software Specification," *Technical rapport*, Technical University of Braunschweig, 2000.

[11] List, G.F., Cetin, M., "Modeling traffic signal control using Petri nets," *IEEE Trans. on Intelligent Transportation Systems*, Vol. 5, N3, pp. 177- 187, 2004.

[12] Febbraro, A.Di., Giglio, D., Sacco, N., "Urban Traffic Control Structure Based on Hybrid Petri Nets," *IEEE Trans. on Intelligent Transportation Systems* Vol. 5, N4, pp. 224-237, 2004.

[13] Padberg, J., Gajewsky, M., "Rule-Based Refinement of Petri Nets For Modeling Train Control Systems," *IFAC Conference on Control Systems Design (CSD2000)*, Elsevier Science, pp, 299-304. 2000.

Malika Medjoudj, Pascal Yim
LAGIS, Ecole Centrale de Lille
Cité Scientifique, BP 48
Villeneuve d'Ascq, 59651, France
E-mail: {malika.medjoudj, pascal.yim}@ec-lille.fr
Received: June 27, 2007
This work was supported by the pole ST2 and the region Nord-Pas de Calais

Malika Medjoudj was born in Tizi-ouzou (Algeria) on February 21, 1977. She received the Engineer Diploma degree in Electronics (Control) and the Diploma of Higher Education Applied in Technical English from Mouloud Mammeri University of Tizi-ouzou (Algeria) in 2001. She obtained the Master in Industrial Systems from UPS-LAAS-CNRS of Toulouse (France) in 2002 and the PhD in Industrial Systems from the same university and laboratory in March 2006. She is actually a Post Doctorate at the Ecole Centrale de Lille in collaboration with INRETS after a scientific stay of six months in the nuclear metrology service of the Université Libre de Bruxelles (FNRS-Belgium). Her research is related to the reliability of hybrid and dynamic systems (computer-controlled systems), checking of temporal constraints, extended Petri Nets for safety (transportation systems), feared scenarios and simulation.

Pascal Yim, Married, 3 children, was born in Papeete (French Polynesia) on November 3, 1963. He is Professor at the Ecole Centrale de Lille, a French Technical University. His research is based on the cross fertilization of concepts from discrete automatic control, and computer science. He has a particular interest in Petri Nets, Constraint Programming and Information Systems. His main application field is the design and optimisation of transportation systems (mainly railways). He published numerous papers in international journals and conferences. He has also supervised several industrial contracts for various customers: SNCF (the French Railway Company), the Fluvial Port of Lille, the "3 Suisses" (a large retail French Company) and other companies. Professor Pascal Yim is a member of scientific board of French "competitivity poles" on Intelligent Transportation Systems (I-Trans) and Distribution (PICOM). He is also correspondent for the North of France of the European Excellence Network on railways (EURNEX).