Toward an Advanced Level Crossing Control Enabled by Train–infrastructure Communication

Mohamed Ghazel
Univ Gustave Eiffel, COSYS-ESTAS
Campus de Lille, F-59650 Villeneuve d'Ascq, France
mohamed.ghazel@univ-eiffel.fr

***Abstract*—This paper proposes a novel level crossing (LC) control architecture designed for railway systems operating under ERTMS Levels 2 and 3. The approach aims to prevent identified hazardous scenarios and to facilitate the integration of LCs into the ERTMS/ETCS framework by moving from passive control schemes to a supervised, communication-based paradigm. The proposed strategy specifically addresses two critical issues: excessive level crossing closure time (ELCCT) and overly short opening durations (OSOD), both of which may lead to unsafe road user behavior. A formal behavioral model based on Time Petri Nets (TPN) is developed using a modular approach to represent the interactions between LC control and railway traffic. The model is enriched with observers to express safety requirements as temporal logic properties, which are verified using model-checking techniques with the TINA tool. The results show that the proposed architecture effectively prevents the considered risky scenarios.**

***Index Terms*—Level crossing safety, control-command, Petri net, model checking.**

I. Introduction and motivation

Safety at level crossings (LCs) remains a major challenge for railway stakeholders. LC accidents represent approximately one third of all railway accidents in Europe, resulting in more than 300 fatalities each year [1], [2]. These accidents generally stem from a complex interplay of technical, organizational, operational, and human-related factors. Collisions at LCs lead to significant material damage and severe traffic disruptions.

Furthermore, they negatively impact the safety perception of railways, even though the primary causes are largely attributed to road user errors, which account for about 95% of LC accidents. It is important to note that, although human factors play a predominant role, such errors are often influenced or exacerbated by technical conditions. In particular, certain technical configurations of LC control systems may directly affect road user behavior.

In parallel, the European Rail Traffic Management System (ERTMS) is the standardized railway control-command and signalling system currently being deployed in Europe and beyond. Its objective is to ensure interoperability while maintaining high safety standards, thereby enhancing the competitiveness of the railway sector. However, with respect to LCs, the ERTMS specifications provide only a limited description of control mechanisms.

As previously highlighted, Certain accident scenarios emerge not from a single cause, but from the interplay between technical system configurations and human mistakes. Accounting for human errors remains a particularly difficult task, as human behavior is influenced by a wide range of cognitive, organizational, and contextual factors that cannot be fully captured by deterministic models. Unlike technical failures, which generally follow well-defined mechanisms and can often be anticipated through systematic analysis, human errors exhibit a high degree of variability and uncertainty. Their occurrence depends on the specific operational context, the operator's perception and interpretation of the situation, workload, stress, experience, and interactions with the system, making them inherently difficult to predict and model accurately.

Nevertheless, safety improvements at LCs can be achieved by refining technical parameters in order to anticipate or even prevent hazardous behaviors by road users.

A functional control architecture for automatic level crossings is proposed in the context of ERTMS Levels 2 and 3. These ERTMS configurations provide near-continuous tracking of train positions by means of GSM-R communications between the trains and the Radio Block Centre (RBC).

The proposed LC control scheme leverages real-time train location information to optimize control decisions, with the objective of preventing certain hazardous scenarios and enhancing overall LC safety. In particular, the study focuses on two critical scenarios that have been identified as major contributors to train–road vehicle collisions: (i) excessively long LC closure durations, and (ii) insufficient opening intervals between successive closure cycles.

Analysis of accident scenarios reveals recurrent situations in which road user errors are predominant. One notable example is the circumvention of half-barriers ("zigzagging") to cross an LC while it is closed to road traffic. This behavior is especially observed at LCs equipped with two half-barriers, which constitute the majority of automatic protected crossings, particularly in France. Studies on human factors indicate that such behavior is often associated with prolonged closure durations. Indeed, train detection sensors are positioned to ensure a minimum warning time prior to train arrival, typically based on the maximum authorized speed of the track section. However, trains with varying speeds—such as freight and passenger trains—or operational constraints (e.g., stops at nearby stations) may result in significantly longer actual closure times. Consequently, slow-moving trains can lead to

unnecessarily long closure periods.

As an illustration, considering a maximum authorized speed of 150 km/h and a nominal closure interval of 25 seconds, a freight train moving at 45 km/h would activate the crossing closure approximately 83 seconds prior to reaching the crossing. Such extended closures may encourage impatient drivers to engage in risky behaviors, such as bypassing the barriers. Previous studies have identified impatience and risk-taking as key contributors to LC incidents. Empirical evidence has repeatedly highlighted the influence of closure duration on risky behaviors at level crossings. As an illustration, a report published in 2007 by the Australian National Railway Level Crossing Behavioral Coordination Group (BCG) revealed that approximately 25% of surveyed drivers acknowledged having violated crossing regulations on at least one occasion [3]. Consistent with these findings, field observations carried out within the PANSAFER project [4] identified excessive closure times as a key contributor to zigzagging behaviors exhibited by road users attempting to bypass the barriers.

Another hazardous scenario occurs at LCs located on double-track lines. While trains traveling in the same direction are typically well spaced, trains approaching from opposite directions may arrive independently. As a result, successive closure cycles may occur within short time intervals: the LC reopens after the passage of one train, only to close again shortly thereafter due to the arrival of another train from the opposite direction. This situation has been identified as a source of confusion and panic among road users, potentially leading to unsafe behavior.

Numerous systems have been developed worldwide to enhance LC safety. For instance, [5] proposes a control system for four-half-barrier LCs designed to prevent vehicles from becoming trapped between entry and exit barriers; this system has been tested in Illinois (USA) and subsequently deployed more widely. Other approaches focus on obstacle detection at LCs. A stereo vision-based system is presented in [6] and tested in Japan, while [7] introduces an ultra-wideband radar solution for detecting vehicles within the crossing area. In [8], a multi-sensor architecture combining infrared and ultrasonic sensors is proposed to detect obstacles on the tracks, with data fusion techniques ensuring high reliability. Additionally, [9] explores train travel time prediction methods to mitigate congestion at LCs, and [10] evaluates innovative treatments for pedestrian crossings. These efforts are part of broader initiatives led by the Federal Railroad Administration (USA) to improve LC safety through a combination of technological, infrastructural, and educational measures [11].

II. Methodology

A generic methodology is adopted to devise a control model capable of mitigating the two hazardous scenarios identified earlier. It begins with the construction of a formal behavioral representation based on Time Petri Nets (TPNs). Beyond their expressive power for capturing the temporal evolution of dynamic systems in a rigorous manner, TPNs offer the important advantage of being supported by dedicated verification tools that can automatically check a wide range of behavioral requirements.

To capture the dynamics of the level crossing (LC) system, a modular modeling approach—similar to that proposed in [12]—is employed. This approach consists of decomposing the overall system into modules, developing elementary behavioral models for each module, and subsequently integrating them while accounting for their interactions. This methodology is particularly well suited for complex systems, as it breaks down the modeling task into more manageable sub-problems. In the present case, the LC system is decomposed into two modules. Such a methodology was also adopted in [13] which serves as a foundation for the present work. Compared with that earlier study, the present paper introduces several enhancements to the underlying models and resolves a number of identified issues.

The first module corresponds to the local control system, responsible for managing barrier movements, road traffic lights, and acoustic alarms. The second module represents rail traffic monitoring, which triggers LC closing and opening cycles through the activation of train approach and departure sensors. For modeling purposes, the local LC control unit is assumed to act solely in response to railway traffic events, with all pertinent operational constraints embedded in the traffic model itself. Without affecting the generality of the approach, the study considers a two-track level crossing protected by two half-barriers (2-HB) and accommodating traffic in opposite directions. The extension to four-half-barrier installations would only require limited adaptations of the control logic. In addition, the track layout is assumed to preclude any overlap between consecutive closure cycles initiated by trains traveling in the same direction. Lastly, the control architecture is analyzed under the assumption of fault-free operation.

Having outlined the configuration of the LC control model, the next step consists in determining the appropriate parameters of the control architecture so as to satisfy operational and safety requirements, with the objective of mitigating the aforementioned risky scenarios. The various distances involved in the control scheme are determined in a realistic way. To illustrate our approach, we will consider the LC setting as depicted in Figure 1.

The last stage of the process focuses on validation, whose objective is to demonstrate that the system complies with the requirements established during the design phase and to provide evidence supporting this compliance. For safety-critical applications, validating the proposed design before operational deployment is a fundamental requirement. This activity typically consists of assessing the system against various classes of requirements, including safety, responsiveness, and liveness properties. The choice of validation technique depends on several factors, such as the form in which the system is represented (e.g., static versus dynamic models) and the expertise available to the analysts.

When the system is described by means of a dynamic behavioral model, validation is commonly performed using either simulation-based approaches or analytical techniques.

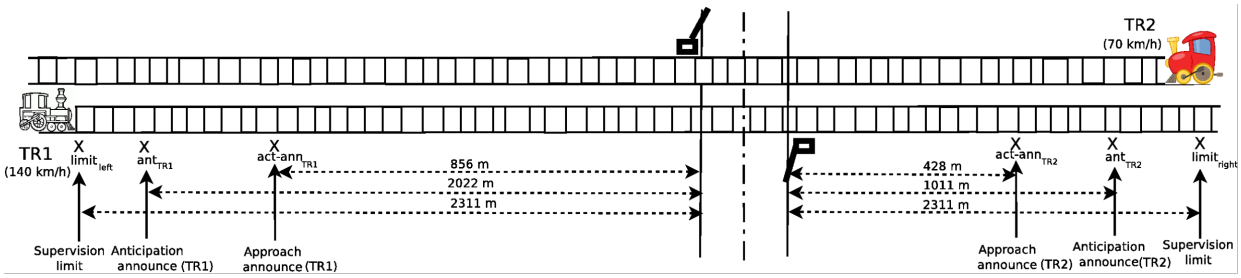

Fig. 1. The generic setting for an automatic LC

While simulation offers flexibility and ease of implementation, its exploratory nature prevents it from covering all possible execution scenarios. Consequently, defects that only manifest under rare or specific conditions may remain undetected. To overcome these limitations, formal methods have increasingly been advocated for the assessment of complex systems [14]. Among them, Model Checking (MC) provides an automated means of exhaustively exploring system behaviors. The outcome of the verification process is definitive, establishing whether a given property is satisfied or violated. In addition, several mature and freely available model-checking environments are capable of handling systems with large state spaces. In practice, these tools automatically evaluate temporal-logic specifications against a discrete-event representation of the system under study. In the present work, the verification of the proposed control architecture is carried out using the model-checking functionalities offered by the TINA toolbox [15].

III. Results

A. Behavioral model for the LC dynamics

For space reasons, the various elementary models developed to describe the behavior of the identified subsystems involved in the LC control are not presented here.

As discussed earlier, the dynamic behavior within the level crossing (LC) area is represented using the Time Petri Net (TPN) formalism. The formal definition of a TPN adopted in this work is given below.

Let $\mathbb{T} \subset \mathbb{Q}^+$ be a temporal domain. A TPN [16] defined over $\mathbb{T}$ is a tuple $N = \langle P, T, B, F, M_0, SIM \rangle$ such that:

- $N = \langle P, T, B, F, M_0 \rangle$ is a marked Petri net (PN), where B and F denote the backward and forward incidence functions, respectively;
- $SIM : T \to \mathbb{T} \times \mathbb{T}_\infty$ is the Static Interval Mapping, with $\mathbb{T}_\infty = \mathbb{T} \cup \{\infty\}$, which associates with each transition its static firing interval, with a rational lower bound (since $\mathbb{T} \subset \mathbb{Q}^+$).

In order to derive a finite-state abstraction of a TPN, the state class paradigm proposed in [17] can be employed. This approach constitutes the core of enumerative methods used for reachability analysis. A state class represents the largest set of states $E = (M, I)$ that are equivalent up to a time shift, meaning that any state in the class can be reached from any other through a suitable translation of time. For bounded TPNs, this leads to a finite abstraction of the state space known as the State Class Graph (SCG). Details on the computation of the SCG using the enumerative approach can be found in [18].

It is worth noting that the tool TINA (TIme Petri Net Analyzer) [15] implements this enumerative approach and allows the automatic construction of the SCG from either a textual or graphical specification of a TPN. The SCG preserves linear-time properties of the system. Furthermore, TINA provides alternative state space abstractions derived from the SCG that additionally preserve branching-time properties, such as CTL and CTL* specifications. Since each abstraction retains different behavioral characteristics, the choice of representation depends on the class of properties to be verified. Beyond reachability analysis, TINA also offers model-checking capabilities for properties expressed in LTL, CTL*, and μ-calculus, as well as path analysis functionalities.

The overall behavioral model combines the elementary dynamics of the various subsystems while explicitly representing the dependencies and interactions among them. To ensure the correct execution of the opening sequence, the following high-level requirements are imposed:

1) A train-approach notification shall be issued sufficiently in advance so that the time interval between its emission and the train entering the intersection zone remains equal to the predefined delay α.
2) Whenever the level crossing is open to road users, the detection of a train-approach notification shall automatically initiate the closing procedure without delay.
3) Once it is guaranteed that the train has cleared the intersection zone (based on the departure sensor), the opening cycle is initiated only if:
   a) no train is currently crossing from the opposite direction;
   b) no train approaching from the opposite direction is expected to trigger a closing cycle within less than γ seconds, as ensured by the anticipated announcement mechanism.

The specified requirements are embedded within the global behavioral representation by augmenting the elementary models with supplementary places, transitions, and arcs that establish the necessary interactions among them. In addition, certain modifications are made to the original models so that the resulting behavior complies with the imposed constraints.

The resulting global model is illustrated in Fig. 2, where transitions represented by thin lines correspond to immediate transitions, i.e., transitions associated with a firing interval $[0, 0]$. The modeling choices are discussed in the following.

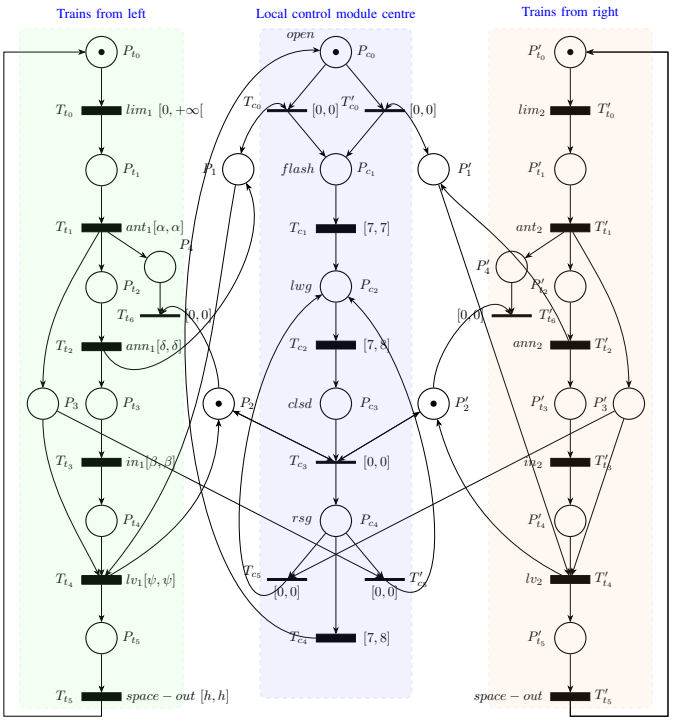

Fig. 2. The model of the dynamics within the LC zone

The global behavioral model is symmetric, as trains approaching from both directions are handled identically by the LC local control module. Consequently, the discussion is restricted to the interaction between the LC control module and a train approaching from the left-hand side; analogous interactions can be derived for trains approaching from the right-hand side. For brevity, certain technical modeling artifacts are not explicitly represented. Furthermore, the individual elements of the model are not described in detail, as the objective is to provide an overall view of how the LC dynamics are captured, thereby establishing a foundation for the subsequent validation phase.

B. Validating the control model

For the purposes of validation, the two hazardous scenarios previously identified are translated into temporal logic specifications, enabling their automatic assessment through model-checking (MC) procedures.

When referring directly to the behavioral model, expressing these scenarios in a straightforward formal manner proves to be challenging. To address this issue and enable a more intuitive specification of properties, the behavioral model is augmented with observer components acting as watchdogs. This is illustrated in Figures 3 and 4, where the colored parts correspond to the monitoring artifacts (observers) established

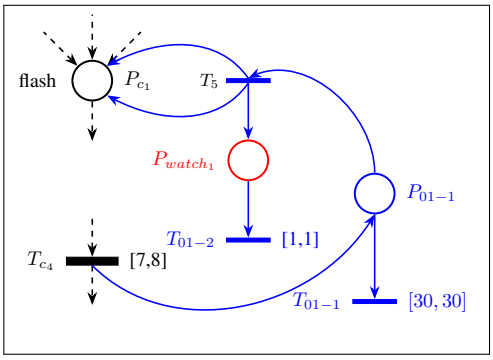

Fig. 3. Established observer for monitoring the OSOD scenario

for the two considered risky scenarios OSOD and ELCCT, respectively.

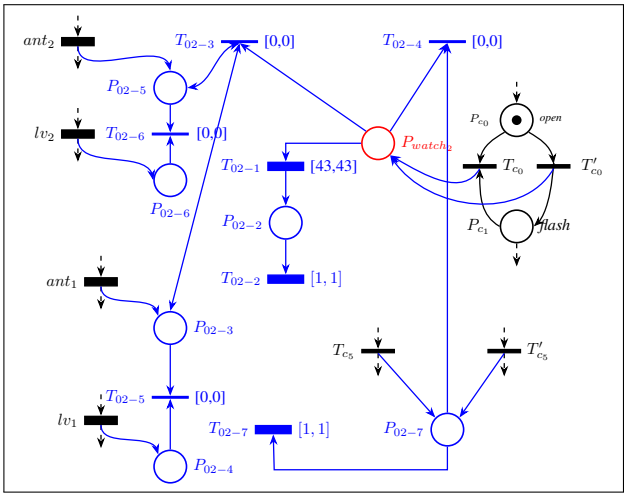

Fig. 4. Established observer for monitoring the ELCCT scenario

For the sake of space, we do not detail all the technical aspects regarding the set observers. Yet, the underlying behind is that an observer is an ad-hoc petri net model that is grafted to the original model and interacting with this latter, so as to monitor some specific property. Such a monitoring facility is designed in such a way that when the monitored property is violated, some specific place in it becomes marked. Therefore, in practice property verification reduces to checking whether the dedicated violation state is reachable. This greatly simplifies properties' formulation. namely, the formulation of the considered properties is illustrated below.

- **Overly Short Opening Duration (OSOD):** On the basis of the introduced observer artifact, examining whether two successive closing cycles may occur with an interval shorter than 30 seconds amounts to checking whether place P_{watch_1} can be reached, i.e., can hold some token. The requirement is formalized as the following temporal logic specification and verified using TINA's model-checking capabilities, more specifically through its SELT model checker:

$$prop_1 : AG(\neg P_{watch_1})$$

which, in TINA syntax, is written as: $[](\neg P_{watch_1})$
This property is verified on the state class graph recomputed after integrating the observer. The results provided by SELT confirm that the examined feature is well fulfilled.

- **Excessive Level Crossing Closure Time (ELCCT):** As previously mentioned, the use of an observer significantly simplifies the formulation of the property to be verified, and amounts to monitor whether place P_{watch_2} can be marked. In other terms, the requirement is expressed as:

$$prop_2 : AG(\neg P_{watch_2})$$

which, in TINA syntax, is written as: $[](\neg P_{watch_2})$
This property is also verified on the recomputed state class graph after integrating the corresponding observer. The results obtained with SELT indicate that the property is satisfied.

IV. Conclusion

The study presented in this paper aims to develop a novel level crossing (LC) control and command architecture within the context of ERTMS Levels 2 and 3 operations, with the objective of preventing a set of identified hazardous scenarios. The rationale behind the proposed control strategy lies in introducing a paradigm shift in LC operation, moving from a conventional passive control approach to a supervised control-command architecture that explicitly accounts for the overall dynamics within the LC area. Beyond its expected safety benefits, the proposed architecture also paves the way to the elimination of certain trackside train detection physical devices, thereby reducing maintenance costs and improving overall system reliability. This approach is consistent with the ongoing trend in railway systems toward minimizing trackside equipment in order to lower installation and maintenance costs.

To support the design of the proposed LC control architecture, a formal framework has been established, allowing the specification of the various operational and safety constraints to be satisfied. The study demonstrates how formal modeling techniques and analytical methods can be effectively employed for the design and validation of control architectures in safety-critical railway systems. In particular, the debugging phase highlights the ability of such models to reveal subtle behavioral features that would be difficult to detect in the absence of automated verification support.

It is worth recalling that the use of formal methods is gaining recognition as a valuable means of supporting the design and assurance activities associated with complex and safety-critical systems, especially in the railway domain. Although additional developments are still required before deployment in operational environments, the proposed contribution provides a foundation for the development of advanced LC control architectures that are fully integrated within the ERTMS/ETCS operational architecture. It should be emphasized that the proposed control paradigm is not specific to the considered use case and may be deployed in other railway control and command systems, as long as continuous train–trackside communication is ensured.

References

[1] A. W. Evans, "Fatal train accidents on Europe's railways: 1980–2009," *Accident Analysis & Prevention*, vol. 43, no. 1, pp. 391–401, 2011.

[2] E. U. A. for Railways, "Report on Railway Safety and Interoperability in the EU 2020 — European Union Agency for Railways," https://www.era.europa.eu/content/report-railway-safety-and-interoperability-eu-2020_en, Jul. 2020.

[3] K. Taylor, "Addressing road user behavioural changes at railway level crossings," in *Proceedings of the ACRS-travelsafe National Conference*, Brisbane, Australia, 2008, pp. 368–375.

[4] L. Khoudour, P. Feltz, D. Bertrand, M. Ghazel, M. Heddebaut, S. Collart-Dutilleul, Y. Ruichek, and A. Flanquart, "PANsafer Project: Towards a safer level crossing," in *11th World Level Crossing Symposium (GLX'2011)*, Tokyo, 2010.

[5] F. Coleman and Y. J. Moon, "Design of gate delay and gate interval time for four-quadrant gate system at railroad-highway grade crossings," *Transportation Research Record*, vol. 1553, pp. 124–131, 1996.

[6] M. Ohta, "Level crossings obstacle detection system using stereo cameras," *Quarterly Report of RTRI*, vol. 46, no. 2, pp. 110–117, 2005.

[7] S. P. Lohmeier, R. Rajaraman, and V. C. Ramasami, "Development of an ultra-wideband radar system for vehicle detection at railway crossings," in *IEEE Conference on Ultra Wideband Systems and Technologies*, 2002, pp. 207–211.

[8] J. J. Garcia, J. Urena, A. Hernandez, M. Mazo, J. A. Jimenez, F. J. Alvarez, C. De Marziani, A. Jimenez, M. J. Diaz, C. Losada, and E. Garcia, "Efficient multisensory barrier for obstacle detection on railways," *IEEE Transactions on Intelligent Transportation Systems*, vol. 11, no. 3, pp. 702–713, 2010.

[9] M. Li, G. Wu, S. Johnston, and W. B. Zhang, "Analysis toward mitigation of congestion and conflicts at light rail grade crossings and intersections," California PATH, Tech. Rep. UCB-ITS-PRR-2009-9, 2009.

[10] Federal Railroad Administration, "Engineering design for pedestrian safety at highway-rail grade crossings," U.S. Department of Transportation, Tech. Rep. DOT/FRA/ORD-16/24, 2016.

[11] Federal Railroad Administration, Office of Research, Development and Technology, "Highway-rail grade crossing safety and trespass prevention research program," in *2015 National Highway-Rail Grade Crossing Safety Training Conference*, 2015.

[12] M. Ghazel and E.-M. El-Koursi, "Two-half-barrier level crossings versus four-half-barrier level crossings: A comparative risk analysis study," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 3, pp. 1123–1133, 2014.

[13] M. Ghazel, "A control scheme for automatic level crossings under the ERTMS/ETCS level 2/3 operation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 10, pp. 2667–2680, 2017.

[14] ——, "Formalizing a subset of ERTMS/ETCS specifications for verification purposes," *Transportation Research Part C: Emerging Technologies*, vol. 42, pp. 60–75, 2014.

[15] B. Berthomieu, P. Ribet, and F. Vernadat, "The tool TINA - Construction of abstract state spaces for Petri nets and time Petri nets," *International Journal of Production Research*, vol. 42, no. 14, pp. 2741–2756, 2004.

[16] P. M. Merlin, "A study of the recoverability of computing systems," Ph.D. dissertation, University of California, Irvine, 1974.

[17] B. Berthomieu and M. Diaz, "Modeling and verification of time dependent systems using time Petri nets," *IEEE Transactions on Software Engineering*, vol. 17, no. 3, pp. 259–273, 1991.

[18] M. Diaz, *Les Réseaux de Petri - Modèles Fondamentaux*. Hermes, 2001.