Computing and Informatics, Vol. 41, 2022, 590–608, doi: 10.31577/cai_2022_2_590

MODULAR DESIGN AND ADAPTIVE CONTROL OF URBAN SIGNALIZED INTERSECTIONS SYSTEMS USING SYNCHRONIZED TIMED PETRI NETS

Hajar Lamghari Elidrissi

LISA Laboratory, Cadi Ayyad University
Marrakesh, Morocco
e-mail: `hajar.lamgharielidrissi@ced.uca.ma`

Ahmed Nait Sidi Moh

Jean Monnet University Saint Etienne
LASPI Laboratory – IUT of Roanne
Campus Pierre Mendes France
20 Av. de Paris, 42300 Roanne Cedex, France
e-mail: `ahmed.nait@univ-st-etienne.fr`

Abdelouahed Tajer

LISA Laboratory, Cadi Ayyad University
Marrakesh, Morocco
e-mail: `a.tajer@uca.ac.ma`

Abstract. Traffic flow at urban intersections varies randomly during the day. It depends on several dynamic factors and requires efficient regulation and flexible control strategies in particular for traffic light regulation. The proposed strategy allows managing the green light time autonomously. The dynamic behavior of traffic signals at intersections can be seen as a discrete event system. Through this paper, a modular Timed Synchronized Petri Net (TSPN) model is developed and a real-time adaptive control strategy of urban signalized intersections is proposed. The control is shared between two communicant actors. The master-slaves approach is adopted in this control strategy. The master (controller) decides the next phase to be served with green light and its duration. While, the slaves (TSPN modules) control the traffic signals displays, phases transitions, and model traffic flow fluctuations. Thanks to the used modularity approach, the developed models reduce the system complexity in terms of combinatorial explosion, and they could be adapted easily for any real intersection. Using the developed models, some interesting properties of the system are checked, and some simulations are performed and analyzed in order to validate the proposed control approach.

Keywords: Traffic lights, control, regulation, modeling, simulation, Petri nets, SUMO

1 INTRODUCTION

Road traffic problems continue to increase and constitute a significant challenge for researchers and transport operators. Traffic management systems mostly inscribe the purpose of reducing congestion, vehicles delay time, fuel consumption, and pollution. The critical cause leading to traffic congestion is the high number of vehicles, which is mainly caused by the increasing population and economic development. Indeed, this heads to queuing phenomena and corresponding delay problems while the infrastructure ability is fixed. They are admittedly becoming a menace to the free flow of traffic in cities. These problems cause an increasing fuel consumption, and raising of atmospheric pollution emission. The request for qualified traffic services continues to rise, making the lack of an adequate management of mobility stronger, especially inside the urban areas.

Several control strategies of traffic lights were developed. The currently available traffic control strategies can be divided into two principal categories [1]. In the first one, fixed timing strategies are determined by the control system using an off-line optimization method implemented by computer programs. The second category includes traffic-responsive strategies which use actuated signal timing plans and offer an on-line optimization and synchronization of traffic signals. These adaptive strategies regulate the traffic lights according to the occurring vehicles at the intersections. Indeed, sensors detect traffic evolution on the road intersections and deliver information to a real-time control.

Traffic signal control can be split into two methods:

1. optimizing the system performance by determining the optimal signalizing plans [2] and
2. establishing how to implement the signal control logic [3].

In this paper we explore both methods by implementing the signal control logic with responsive traffic signals timing plans.

In the last decades, various tools have been developed for modeling, optimizing, and controlling traffic flow at intersections. The most well-known and generally utilized are SCOOT (Split cycle offset optimization techniques) [4], SCAT (Sydney coordinated adaptive traffic) [5], and OPAC (Optimization policies for adaptive control) [6]. Unfortunately, the cost and duration of installation of these systems are a hindrance to their development. The second category of control strategies using Artificial Intelligence (AI) techniques has been developed for traffic flow management. Such as the fuzzy logic [7], bee colony optimization [8], and neural networks [9]. Furthermore, adaptive signal control policies based on AI have gained increasing consideration due to their undeniable improvement of traffic control performance compared to the pre-timed signal control. However, these methods are quiet under research and are not yet ready for immediate implementation due to the hindrances, such as specific hardware specifications and high computational cost.

According to a certain point of view, the traffic signal control at the intersections can be observed as a discrete event system (DES). The events in the road traffic are interpreted by the arrivals and departures of vehicles at the intersection, and beginning or completion of the signal timing plans to control the traffic junction [10]. Thanks to its ability to represent the dynamic behavior of traffic flow at intersections, Petri nets (PNs) are chosen to be used in this work as a powerful tool to model this kind of DES. Moreover, the solid graphical and mathematical bases of PNs allow DES to be ruled by a set of mathematical equations, therefore presenting a formal way for system analysis and checking of desirable specifications. It is worth noting that a PN models in traffic control could represent the flow of vehicles, traffic lights, and decision making. Also, PN can be used as a modeling tool, whereas the decision-making is ensured by a complementary control tool while considering more specifications such as (max, +) algebra [11].

Petri nets were first applied for traffic modeling and control in [12]. Thereafter, other researchers investigated this formal tool. The authors of [13] have discussed the use of PNs for modelling traffic signal control and perform a structural analysis of the developed control model. Besides, Febbraro and Giglio have presented in [14] a traffic model using timed PN formalism. There are other research works where time is introduced into the model to represent phase duration, such as in [15]. The work developed in [2] solves a traffic congestion problem with a regulatory traffic light control technique that prevents vehicles from crossing traffic congestion zones; using synchronized timed Petri nets. The durations of traffic signals in these works are based on fixed and predetermined durations represented in timed PNs.

In order to enrich all the aforementioned researches and bring our contribution in this field, we focus in this paper on the enhancement of system performances in terms of minimizing waiting delays of vehicles, decreasing the waste time within the intersection, and ensuring vehicles safety, and finally environment issue (gas emission) at a given intersection. Classical signal control methods based on ordinary Petri nets are not sufficient because usually, traffic flow and signal settings are mutually interdependent and dynamic. Moreover, there is a wasted time when the traffic light remains green, whereas all vehicles are evacuated from the intersection lane. Our paper addresses these issues by using a new modular TSPN based model to control the signal lights and phase transitions. In fact, the added-value of our solution is double: model the dynamics of traffic intersections using a modular TSPN (slaves) and manage conflicts using a controller (master) with combinatorial optimization.

The remainder of the paper is organized as follows. Section 2 illustrates the problem statement and the modeling approach. In Section 3 we address the developed control approach. The signal control analysis based on PNs basis is given in Section 4. Performed simulations using a traffic simulator and discussions about obtained results are reported in the same section. Finally, in Section 5, we present our conclusion and some perspectives.

2 STUDIED SYSTEM AND MODELLING APPROACH

2.1 System Description and Aims

The studied system in this paper is illustrated in Figure 1. This intersection is composed of four directions: north, south, east, and west; three lanes for each direction. As shown in this figure, each road of the intersection includes sensors to detect the arrival of vehicles. Traffic flow information in each intersection road could be obtained via these sensors implemented at a given distance from the stop line.

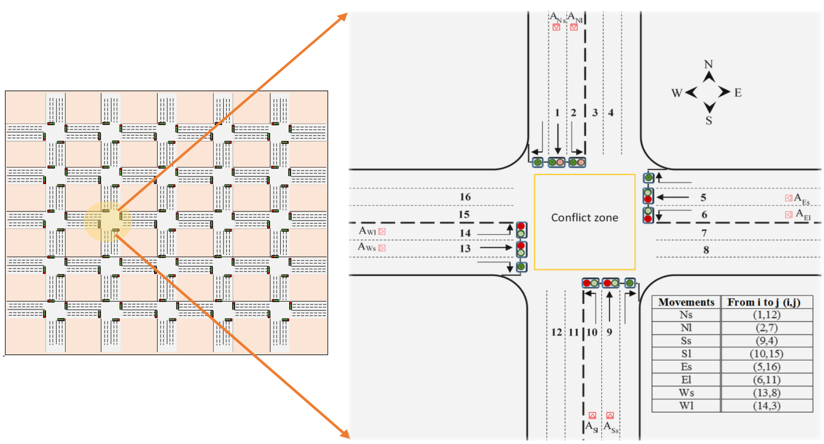


Figure 1. An intersection with four roads

We recall that from Figure 1, we can read these two terms direction and movement. The directions represent the four roads at the intersection, while a vehicle has three movements: left-turn, straight-ahead, and right-turn. Ordinarily, the vehicles of the right-turn movement do not have conflicts with vehicles coming from other movements and directions. So we assume that right-turn vehicles are directed to precise lanes, and the lights for them are green all the time. Consequently, in this study, we consider four directions and two movements in each direction. Based on the movement notion we define the term "phase" that will frequently appear in the rest of the paper. A phase is a safe combination of two movements (absence of vehicles crash). That means the green light could be given to combined movements (even of different directions) in a phase simultaneously without losing safety at the conflict zone (common and shared area of the fourth lanes of the intersection). For simplification reasons and without impacting the desired functioning of the system, we suppose that the time of amber light is included in the red time. So the sequence of active color becomes from green to red, and then back to green. As earlier mentioned, the proposed solution in this paper consists in controlling traffic flow within the intersections using variable temporizations. This control consists of satisfying these three specifications:

1. Minimize the downtime of vehicles at the intersection.
2. Decrease the waste time between phases (evacuate the maximum number of vehicles in a minimum time).
3. Ensure safety in the collision zone (avoid vehicles collision) by managing conflicts between different movements.

To this end, and as we already mentioned, we adopt in this work a modular approach by modeling each movement by a PN model as we detail in the next subsection.

2.2 Petri Net Based Models

The studied system is considered as a discrete event system characterized by occurrence of synchronized and timed events. Therefore, a specific type of PNs is used for modeling and control issues. The Timed Synchronized Petri Net model represents each movement. These models, called and playing the role of "salves", represent the traffic signals displays, phases transitions, and traffic flow fluctuations. A controller, called and playing the role of "master" could manage conflicts between slaves and calculates their variable temporizations (see Section 3).

2.2.1 Structure of the Timed Synchronized Petri Net (TSPN)

This subsection presents the class of PNs used to model traffic flow and lights. We call this class Timed Synchronized Petri Nets. It is a bipartite graph described by the seven-tuple $\mathrm{TSPN} = (P, T, A, f_w, M_0, \tau, \mathrm{Evt})$:

- P : is a finite set of discrete places.
- $T = T_s \cup T_t \cup T_{st}$ is a finite set of transitions consisting of synchronized transitions subset T_s, the temporized transitions subset T_t, and synchronized temporized transitions subset T_{st}. $T_s \cap T_t \neq \varnothing$
- $A \subseteq (P \times T) \cup (T \times P)$ is a finite set of arcs.
- $f_w : A \longrightarrow \{1, 2, \dots\}$ is the arc-weight function.
- $M_0 : P \longrightarrow \{0, 1, 2, \dots\}$ is an initial marking.

- $\tau : T \longrightarrow R^+$ firing time function, where R^+ is the set of non negative real numbers.
- Evt $= \{A_{\langle ij \rangle}, E_{\langle ij \rangle}\}$ is a set of external events that are linked to certain transitions. Where the index i represents directions: north, south, east, and west, respectively ($i \in \{N, S, E, W\}$); and j represents straight-ahead and left-turn, respectively ($j \in \{s, l\}$).

2.2.2 TSPN Model for Traffic Flow and Lights

In this subsection, we develop the traffic flow and the signal control models for each intersection movement using TSPN. For more understanding, we are going to describe each part of the global model (Figure 2) separately using the modular approach.

Lanes and Traffic Flow Models. It is worth noting that the PNs structures of the eight movements are similar. So, only the TSPN model of the first movement north, and straight-ahead ($\langle Ns \rangle$)) is detailed in what follows (Figure 2). The transition T_{Ns1} is used to generate tokens (arrival of vehicles to the intersection) when the sensor A_{Ns}, located at a given distance from the stop line, detects vehicles coming from the directions north and want to go straight ahead (the movement $\langle Ns \rangle$). Tokens in the place P_{Ns1} represent vehicles waiting to enter into the intersection. The timed transition T_{Ns2} can be fired every t_{vNs} time units. Its firing moves all tokens from the place P_{Ns1} into the place P_{Ns2}, via arcs with variable weight. The time t_{vNs} is a delay which prohibits the firing of T_{Ns2} until the end of the evacuation of all vehicles coming from the movement $\langle Ns \rangle$) ($M(P_{Ns2}) = 0$).

The role of this delay is to save the number of vehicles, fixed at the decision making, to be evacuated from the intersection when the traffic light is green. The timed transition T_{Ns3} represents the stop line, and the place P_{Nsv} represents the green light (north- straight-ahead). A token in the place P_{Nsv} (common-place for the movement Ns between traffic flow and traffic light (Figure 2)) allows the firing of the timed transition T_{Ns3}; bidirectional arrows are used here to represent a reading of the place P_{Nsv} (reading of a place means withdrawal and addition of a token in the same place by firing its output/input transition). In our case, reading the place P_{Nsv} means the evacuation of vehicles (one by one) as long as the light is green. With this solution, the first constraint (specification) is respected, so the light must be green in order to allow a vehicle to cross the stop line. The time t_n associated with the timed transition T_{Ns3} represents the necessary time for each vehicle to cross the stop line. While firing the transition T_{Ns3} (if there are tokens in P_{Ns2}, and P_{Nsv}), the number of tokens in P_{Ns2} will be reduced by one every t_n time unit until this place becomes empty. Reaching this state means that all vehicles authorized (limited by the position of the sensor) in this cycle are fully evacuated. The number of tokens in the place P_{Nsc} represents the capacity (c) of the lane. This capacity is the possible number of vehicles

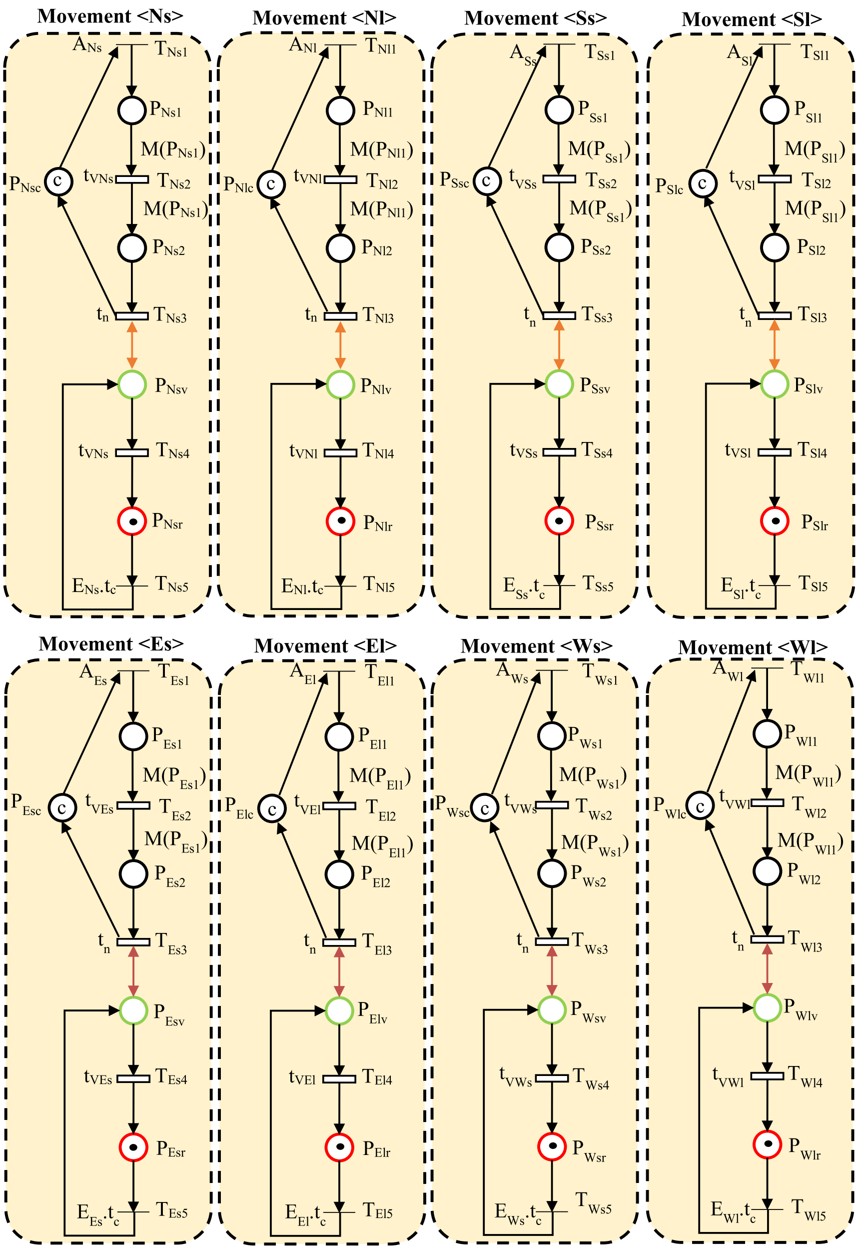


Figure 2. Global T-timed and Synchronized Petri Net of an intersection with eight movements

between the sensor and the stop line. For every firing of the transition T_{Ns1}, the number of tokens in P_{Nsc} reduces by one. On the other hand, every firing of the transition T_{Ns3} the number of tokens in P_{Nsc} increases by one. If there are no tokens in the place P_{Nsc} (the lane is fully occupied by vehicles), the transition T_{Ns1} cannot be then fired. Its next firing will be done whenever a vehicle leaves the intersection. Similarly, we develop the TSPN models of other considered movements as depicted in Figure 2.

Traffic Lights Model. Similar to the lanes and traffic flow models, only the TSPN model of the movement $\langle Ns \rangle$ (Figure 2) is explained in what follows. The TSPNs traffic light models of the other movements are similar to the TSPN model of the movement $\langle Ns \rangle$. If the decision is given to the movement $\langle Ns \rangle$, the external event E_{Ns} occurs and takes one as a value, which allows the firing of the transition T_{Ns5}. This external event takes zero if there is no decision or in the case of the absence of vehicles in this movement. Such external event acts as an interrupter with two states on/off enabling to fire or not the transition T_{Ns5}. A delay t_c is added to the transition T_{Ns5} to ensure that the last vehicle for one cycle of the green light has left the conflict zone before changing the light from the red to green for the next chosen movement. When this duration elapses the light becomes green for t_{vNs} time units, for the movement $\langle Ns \rangle$. This duration corresponds exactly to the required time for vehicles to leave the stop line (to clear out the places P_{Ns2} (see Figure 2)). At the end of these duration, the transition T_{Ns4} will be fired, so signal light becomes red for the movement $\langle Ns \rangle$, and the controller (master) restarts to make a new decision.

TSPN Global Model.

Once TSPN sub-models representing all considered movements through the modular approach are obtained, we aggregate them to obtain the global TSPN model of Figure 2 (see Table 2 in Section 4 for some scenarios). In order to facilitate the global model readability and understanding, we define the meaning of all its components. Table 1 addresses all the model components with: $i \in \{N, S, E, W\}$, and $j \in \{s, l\}$. The index i represents directions north, south, east, and west respectively, and j represents straight-ahead and left-turn, respectively. Five places and five transitions compose the PN sub-model of each movement. Associated weights with arcs are equal to one, except two directed arcs (P_{ij1}, T_{ij2}) and (T_{ij2}, P_{ij2}) aiming to move all tokens (vehicles) presented in P_{ij1} to P_{ij2} by firing T_{ij2} only once.

The events in this TSPN model are managed by two parameters: temporizations and external actions (sensors and master control). For each movement $\langle ij \rangle$, there are just two external events associated with synchronized transitions. The first external event (A_{ij}) is associated with the transition T_{ij1} which represents the arrival of vehicles at the movement $\langle ij \rangle$, the second one (E_{ij}) is associated with the transition T_{ij5} which represents a decision given by the master to the movement $\langle ij \rangle$ of the chosen phase. The role of temporizations associated with some transitions is to delay their firing in order to respect given specifications.

Transitions	**Meaning**
T_{ij1}	Detection of a vehicle in the movement $\langle ij \rangle$.
T_{ij2}	New decision (The previous green light time has expired).
T_{ij3}	Departure of a vehicle from the movement $\langle ij \rangle$.
T_{ij4}	Light switch from green to red.
T_{ij5}	Light switch from red to green.
Places	**Meaning**
P_{ij1}	Vehicle entrance to the movement $\langle ij \rangle$.
P_{ij2}	Vehicles to be evacuated from the movement $\langle ij \rangle$.
P_{ijv}	Green light.
P_{ijr}	Red light.
P_{ijc}	Capacity of the lane in the movement $\langle ij \rangle$.
Temporizations	**Meaning**
t_n	Necessary time for a vehicle to leave the stop line.
t_{vij}	Time of the green light for the movement $\langle ij \rangle$.
t_c	Crossing time of the last vehicle reaching the intersection (red safety).
External events	**Meaning**
A_{ij}	Vehicle detection sensor in the movement $\langle ij \rangle$.
E_{ij}	Controller decision for the movement $\langle ij \rangle$.

Table 1. Description of the PN model components of Figure 2 ($i \in \{N, S, E, W\}$, and $j \in \{s, l\}$)

The controller is the responsible to make a decision via the external events E_{ij} and calculates the variable green light time t_{vij} of the chosen movement(s). In fact, the adopted control strategy will be presented in Section 4.

3 ADAPTIVE CONTROL APPROACH

The second part of this contribution concerns the development of a controller (master). Indeed, this controller role is to allow managing of conflicts between the slaves (TSPN sub-models) in an optimal way. Based on background of PNs, some calculations will be used to implement the control strategy as a computing program.

3.1 Traffic Control Principle

This subsection deals with the development of a traffic control method within the intersections. The target of this control method is to manage the smooth evacuation of all presented vehicles at different movements while avoiding their collision and delays. It is based on the algorithm of Figure 3.

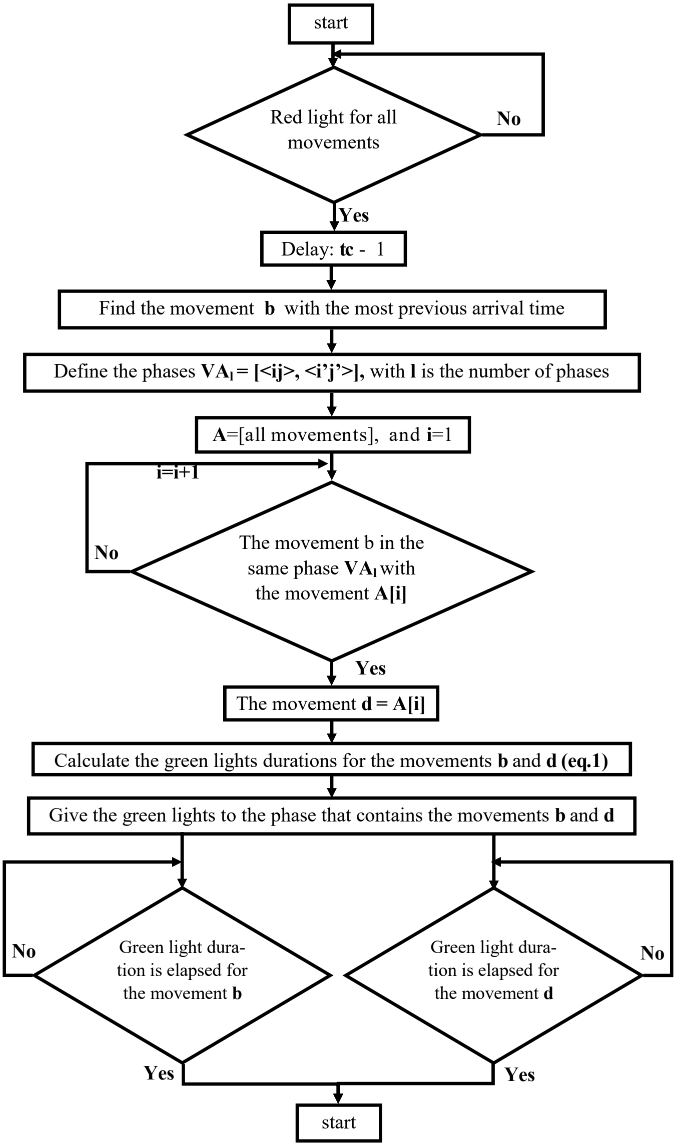


Figure 3. Traffic control algorithm

This algorithm runs as follows. At the end of the green lights for the previous phase and the red safety duration, the controller makes a new decision according to the arrival time of vehicles and their number at each movement. It gives priority to the movement "b" containing the vehicle with a higher delay, which corresponds to the most anterior arrival time at the intersection. As aforementioned, a phase is a combination of two movements $\langle ij \rangle$ and $\langle i'j' \rangle$ (with $\langle ij \rangle \neq \langle i'j' \rangle$). In this algorithm, a phase is represented by a list VA_l, with $l = \{1, 2, 3, \ldots, m\}$ represents the phase and m is the number of phases (depends on the studied scenario). Once the list of phases VA_l and the list of movements A are defined, we make a test allowing to find from A the movement "d" that belongs with the chosen movement "b" to the same phase VA_l. The next step is to calculate the green lights times for both movements "b" and "d". These durations are obtained by using the number N of vehicles authorized to leave the intersection (it is fixed when the controller starts to make a new decision), the headway time h, and the time t_a corresponds to the necessary time to evacuate the first vehicle waiting at the queue (Equation (1)).

$$t_{vij} = (N - 1) \times h + t_a \tag{1}$$

The headway time h_i of the vehicle i arrived at time t_i represents the inter-arrival time of the vehicles i and $i - 1$, as expressed in the Equation (2). We assumed that the headway in our case is the same for all vehicles and fixed at three seconds in the studied scenario later in this paper.

$$h_i = t_i - t_{i-1} \tag{2}$$

with: $i = 1, 2, 3, \ldots, N$.

Once the controller defines the chosen movements "b" and "d" and their green lights durations, it communicates with the system represented by the TSPN model, via the external events E_{ij} ($E_{ij} = 1$ for the chosen movement of a given phase and $E_{i'j'} = 0$ otherwise $(i, i' \in \{N, S, W, O\}$, $j, j' \in \{s, l\}$, and $i \neq i'$, $j \neq j'$)).

The considered safety constraint, while making a decision, concerns the following three points:

1. Movements in each phase are chosen while taking into account the safety issue;
2. The controller gives the priority to only one phase each time;
3. The delay t_c is added to ensure that the last vehicle leaves the intersection before turning on the green light for a new chosen phase.

4 A CASE STUDY: COMPARISON, ANALYSIS AND DISCUSSION

4.1 Simulation Context

In what follows, we proceed to the validation of our modeling and control method by both a dedicated simulator for traffic flow simulation, named SUMO [16], and using Matlab programming. Our aim is to validate the proposed control method using two different simulation tools, Matlab and SUMO. Indeed, the modeling and control logic proposed were implemented using Matlab programs for simulation and analysis issues. This allows generating the marking evolution, the sequences of firing, the states of external events E_{ij} according to the taken decision, and the state of the traffic lights for all movements.

In order to compare both simulation methods, we use the same input data (the vehicle arrival frequency and their arrival times). These data are generated from SUMO simulator using a Python program. With SUMO, we represent the digital model of the studied intersection (Figure 1). Some screen-shots of this virtual model are given in Table 2). In order to detect vehicles arrival into the intersection, eight induction loops are installed at a distance from the stop lines. The values returned by these loops are the states of the traffic flow within the last simulation step.

4.2 Verification and Validation of System Properties Through PN's Analysis

By analyzing the established TSPN model (marking generated by Matlab program), some PNs properties about the well-functioning of the system are checked in what follows. We refer to [17] for more details about these properties.

Accessibility. In our system, vehicles are authorized to cross the intersection only if the traffic light is green; this specification is verified by simulation. All vehicles have access to the intersection and to be evacuated whenever they are authorized (switching from the red to green light and vice-versa). So all desired markings are accessible. This means that the propriety of accessibility is achieved.

Boundedness. This property is verified in our model because, for each model marking, all places contain a limit number of tokens (traffic light, vehicles, decision, lane capacities). So, each place is bounded, and an infinity of vehicles cannot be sojourned (accumulation of vehicles) for a long time within the intersection.

Vivacity and deadlock-free. Based on the simulations results and the marking evolution, the arrival and departure of vehicles is ensured. Also, traffic lights vary for all movements. So, all transitions in the proposed TSPN model are alive, which demonstrates that the PN is alive.

The performed simulation runs for more than one hour and can go beyond, with the absence of a blocking situation. As long as the vehicles arrive at the intersection, the deadlock-free is ensured. Through the verification and validation of these properties, we prove a safe working of the studied system.

4.3 Adaptive Control Validation: Simulation and Comparison

Performed simulations and obtained results about the analysis of the system functioning are presented in this section. It is important to note that the proposed control method in this work could be applied to all types of intersections and all phases. However, a case study with just four phases is given in this paper (Figure 4) to prove the applicability of the proposed control method. A comparison is performed using the SUMO simulator to enrich the discussion with qualitative parameters.

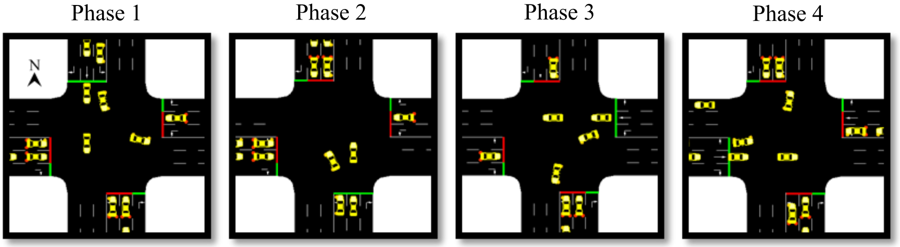


Figure 4. Intersection phases

4.3.1 Validation with Simulation

Obtained results based on analytic simulation with Matlab in Figure 5 illustrates the arrival time of each vehicle to the eight movements and the decision making instant to evacuate vehicles from the chosen phase.

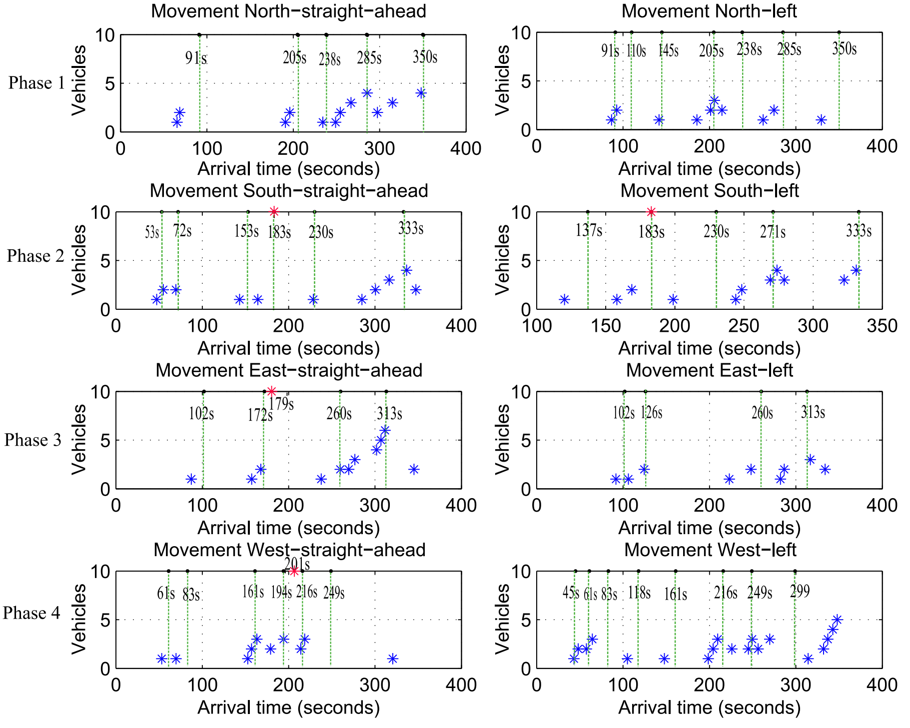


Figure 5. The number of vehicles versus their arrival times and decision making times

It shows that the evacuation of vehicles can be authorized just for the two movements of the same phase; if both of them contain vehicles. For the eight movements, the controller gives the green light to the phase that contains vehicles

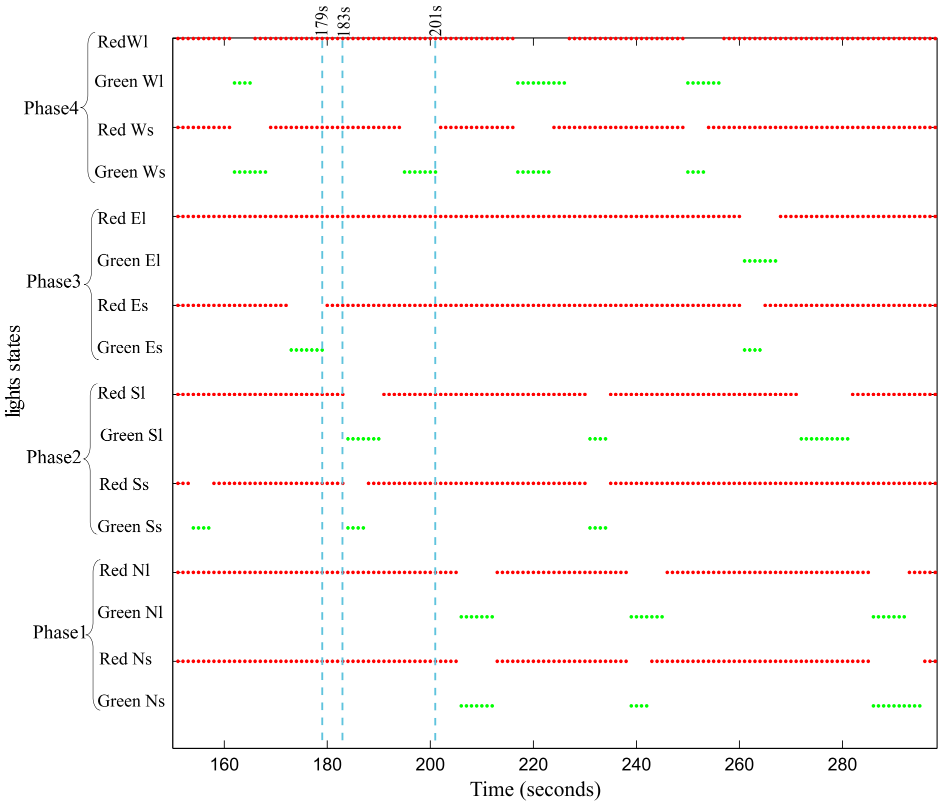


Figure 6. Green and red lights switching over the time for the four phases

with the most anterior arrival time. During the time interval $[50\,\mathrm{s}, 350\,\mathrm{s}]$, a total of 28 decisions for all the four phases are taken. A record of the traffic lights states is shown in Figure 6. It represents the bi-color state for the eight movements over time.

In order to link between the simulations results in Figures 5, 6, and what happens at the intersection, three simulation steps $k = 179\,\mathrm{s}, 183\,\mathrm{s}$, and $201\,\mathrm{s}$ are considered in Table 2. These steps are shown: by red asterisks in Figure 5 and by the blue dashed lines in Figure 6. As previously mentioned, a vehicle detected at the moment of decision making or after is not be considered until the next decision making. Hence in Figure 5, the decision is given to the phase 4 – movement $\langle Ws \rangle$ (no vehicle in the movement $\langle Wl \rangle$) at the step 194 s, the vehicle detected at this instant is not considered (blue asterisk).

Table 2 shows two views of the intersection: virtual model (with SUMO) versus TSPN model (with PNs) for three different moments during the simulation (at $k = 179\,\mathrm{s}$, 183 s and 201 s). This illustration shows a perfect similarity between the two used approaches for simulation and conducts to the same results. For

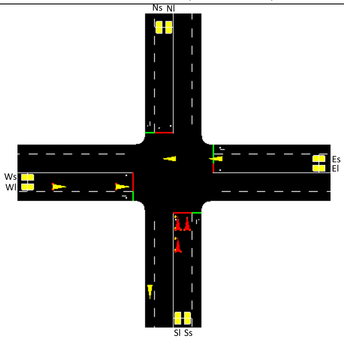
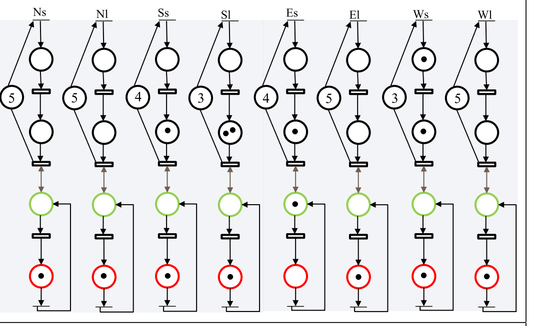
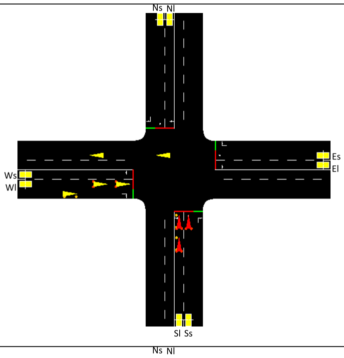
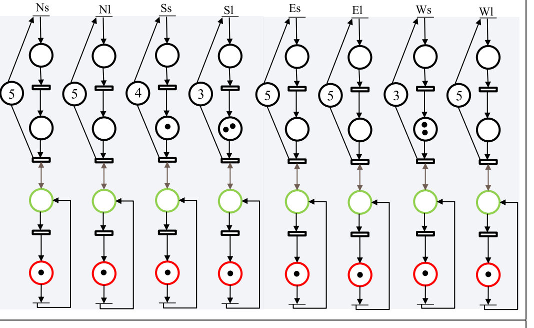
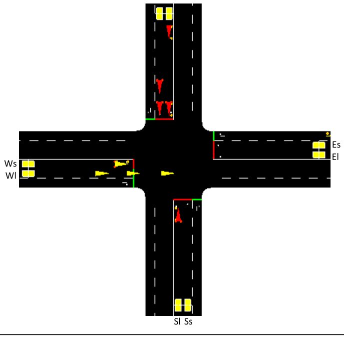
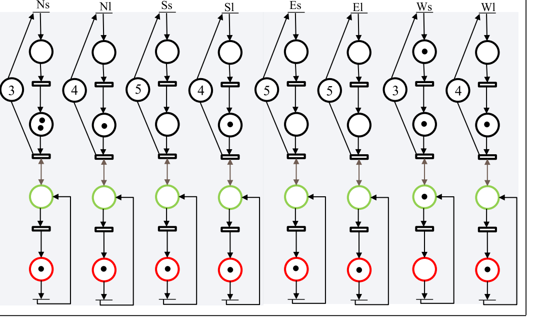
k	Virtual models (SUMO)	Corresponding PN model states
179		
183		
201		

Table 2. The state of the intersection in three different simulation steps: SUMO models vs. TSPN models

example, at the moment $k = 197\,\mathrm{s}$, virtual model shows that the light is red for the movements $\langle Ns \rangle$, $\langle Nl \rangle$, $\langle Ss \rangle$, $\langle Sl \rangle$, $\langle El \rangle$, $\langle El \rangle$, $\langle Ws \rangle$, and $\langle Wl \rangle$. The only movement with green light is $\langle Es \rangle$. The same traffic light situations are observed in the TSPN models (marking of TSPN models places). Also, the number of vehicles at the intersection (on each movement) corresponds to number of tokens in the TSPN places.

Through all these analysis, we conclude that the green light can be given just to the movements of the same phase. Also, the green light duration is variable and depends on the number of vehicles at the instant of decision making. Moreover, the master gives a decision according to the most anterior arrival time of vehicles at the intersection. Consequently, the wasted time and the delay of vehicles are decreased while ensuring safety. Finally, we can confirm the efficiency of the proposed adaptive control through all the obtained and discussed results.

5 CONCLUSION

This paper presents a general timed synchronized Petri Net model for representing the dynamic evolution of traffic lights at road intersections. A modular approach is used to model each movement separately (slaves) respecting the system constraints. In order to avoid collisions of vehicles at the common area of the intersection and manage durations of green light according to the number of vehicles, we developed a dynamic control strategy based on an on-line decision-making policy by a controller (master). The following dilemma is taken into account between minimizing the waiting time of vehicles, evacuation of a maximum number of vehicles in a minimum time, and ensure safety at the intersection.

The proposed model in this paper is based on Petri nets for which variable weights are associated with some arcs. Using the robust background of PNs, some properties about the well-functioning of the intersection are proved by simulating and analysing and the TSPN model. In order to simulate and validate the proposed model, the whole parts, including the obtained TSPN model and the control algorithm are translated into an executable program in Matlab tool. The Matlab input data (the arrival frequency of vehicles to the intersection, their arrival times and their number) are generated using SUMO 1.0.0 open-source traffic simulator. The control logic is also implemented through SUMO to validate the proposed solution and compare it with the analytic results obtained from the Matlab program. In the future work, using SUMO simulator, next comparisons between our adaptive control strategy and the existing control approaches in the literature will be investigated with the focus on waiting time, gas emissions, and fuel consumption. Also, we will investigate the possibility how to enhance the adaptive control method to prioritize the movements with incoming emergency vehicles using the technology of vehicle to infrastructure to vehicle (V2I2V) communications.

REFERENCES

[1] Diakaki, C.—Papageorgiou, M.—Aboudolas, K.: A Multivariable Regulator Approach to Traffic-Responsive Network-Wide Signal Control. Control Engineering Practice, Vol. 10, 2002, No. 2, pp. 183–195, doi: 10.1016/S0967-0661(01)00121-6.

[2] Huang, Y. S.—Weng, Y. S.—Zhou, M.: Design of Regulatory Traffic Light Control Systems with Synchronized Timed Petri Nets. Asian Journal of Control, Vol. 20, 2018, No. 1, pp. 174–185, doi: 10.1002/asjc.1536.

[3] List, G. F.—Mashayekhi, M.: A Modular Colored Stochastic Petri Net for Modeling and Analysis of Signalized Intersections. IEEE Transactions on Intelligent Transportation Systems, Vol. 17, 2016, No. 3, pp. 701–713, doi: 10.1109/TITS.2015.2483324.

[4] Robertson, D. I.—Bretherton, R. D.: Optimizing Networks of Traffic Signals in Real Time – the SCOOT Method. IEEE Transactions on Vehicular Technology, Vol. 40, 1991, No. 1, pp. 11–15, doi: 10.1109/25.69966.

[5] Sims, A. G.—Dobinson, K. W.: The Sydney Coordinated Adaptive Traffic (SCAT) System Philosophy and Benefits. IEEE Transactions on Vehicular Technology, Vol. 29, 1980, No. 2, pp. 130–137, doi: 10.1109/T-VT.1980.23833.

[6] Gartner, N. H.—Tarnoff, P. J.—Andrews, C. M.: Evaluation of Optimized Policies for Adaptive Control Strategy. Transportation Research Record, Vol. 1324, 1991, pp. 105–114.

[7] Shiri, M. S.—Maleki, H. R.: Maximum Green Time Settings for Traffic-Actuated Signal Control at Isolated Intersections Using Fuzzy Logic. International Journal of Fuzzy Systems, Vol. 19, 2017, No. 1, pp. 247–256, doi: 10.1007/s40815-016-0143-7.

[8] Jovanović, A.—Nikolić, M.—Teodorović, D.: Area-Wide Urban Traffic Control: A Bee Colony Optimization Approach. Transportation Research Part C: Emerging Technologies, Vol. 77, 2017, pp. 329–350, doi: 10.1016/j.trc.2017.02.006.

[9] Castro, G. B.—Hirakawa, A. R.—Martini, J. S.: Adaptive Traffic Signal Control Based on Bio-Neural Network. Procedia Computer Science, Vol. 109, 2017, pp. 1182–1187, doi: 10.1016/j.procs.2017.05.394.

[10] Tzes, A.—Kim, S.—McShane, W. R.: Applications of Petri Networks to Transportation Network Modeling. IEEE Transactions on Vehicular Technology, Vol. 45, 1996, No. 2, pp. 391–400, doi: 10.1109/25.492914.

[11] Idel Mahjoub, Y.—Nait Sidi Moh, A.—Chakir El Alaoui, E.—Tajer, A.: Petri Nets Conflicts Resolution for Performance Evaluation and Control of Urban Bus Networks: A (max,+)-Based Approach. Transportmetrica A Transport Science, Vol. 16, 2020, No. 2, pp. 164–193, doi: 10.1080/23249935.2018.1527864.

[12] DiCesare, F.—Kulp, P. T.—Gile, M.—List, G.: The Application of Petri Nets to the Modeling, Analysis and Control of Intelligent Urban Traffic Networks. In: Valette, R. (Ed.): Application and Theory of Petri Nets 1994 (ICATPN 1994). Springer, Berlin, Heidelberg, Lecture Notes in Computer Science, Vol. 815, 1994, pp. 2–15, doi: 10.1007/3-540-58152-9_2.

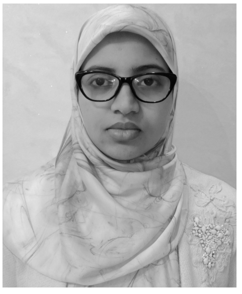
[13] List, G. F.—Cetin, M.: Modeling Traffic Signal Control Using Petri Nets. IEEE Transactions on Intelligent Transportation Systems, Vol. 5, 2004, No. 3, pp. 177–187, doi: 10.1109/TITS.2004.833763.

[14] Di Febbraro, A.—Giglio, D.: On Representing Signalized Urban Areas by Means of Deterministic-Timed Petri Nets. Proceedings. of the 7th International IEEE Conference on Intelligent Transportation Systems, 2004, pp. 372–377, doi: 10.1109/ITSC.2004.1398927.

[15] Luo, J.—Huang, Y. S.—Weng, Y. S.: Design of Variable Traffic Light Control Systems for Preventing Two-Way Grid Network Traffic Jams Using Timed Petri Nets. IEEE Transactions on Intelligent Transportation Systems, Vol. 21, 2020, No. 7, doi: 10.1109/TITS.2019.2925824.

[16] Lopez, P. A.—Behrisch, M.—Bieker-Walz, L.—Erdmann, J.—Flötteröd, Y. P.—Hilbrich, R.—Lücken, L.—Rummel, J.—Wagner, P.—Wiessner, E.: Microscopic Traffic Simulation Using SUMO. 2018 21st IEEE International Conference on Intelligent Transportation Systems (ITSC), 2018, pp. 2575–2582, doi: 10.1109/ITSC.2018.8569938.

[17] Murata, T.: Petri Nets: Properties, Analysis and Applications. Proceedings of the IEEE, Vol. 77, 1989, No. 4, pp. 541–580, doi: 10.1109/5.24143.



Hajar Lamghari Elidrissi received the Master degree in automation, control, and computing in 2016. She is currently Ph.D. student and member of Systems Engineering and Applications Laboratory at the University of Cadi Ayyad, Morocco. Her research interests are discrete event systems (DES), road traffic, control, regulation, modeling, and simulation.

Ahmed Nait Sidi Moh is Full Professor at the Jean Monnet University (UJM), Saint Etienne, France. He received the Ph.D. degree in computer science and automatic control from the University of Technology of Belfort (UTBM), Belfort, France, in 2003. He was Assistant Professor at the UTBM from 2004 to 2011. After, he joined as Associate Professor the University of Picardie Jules Verne (UPJV), Amiens, France, where he obtained his Habilitation à diriger des Recherches (HDR) in computer engineering in 2016. He is the author of several articles published in international journals, conferences, and workshops. He is involved in several national and international events such as conferences and workshops organization, technical program committee member of many international journals and conferences; participation to national research groups. His research has been supported by many research and development projects including European grants, regional projects, and EU EACEA Erasmus Mundus projects. He is interested to research topics in the field of healthcare and industrial engineering, with problems of modeling, analysis and control, performance evaluation, resources sharing, optimization, scheduling and interoperability for service composition, information and communication technologies.

Abdelouahed Tajer achieved his Master degree in systems optimization and safety at the University of Reims Champagne Ardenne/University of Technology of Troyes in France and his Ph.D. degree in control of discrete-events systems at the University of Reims Champagne-Ardenne in France in 2005. His research interests are: discrete event systems, fault diagnosis, modeling, supervisory control theory, optimal control, transportation and manufacturing systems. Currently, he is Full Professor at the National School of Applied Sciences of Marrakech, Cadi Ayyad University, Morocco.