Modular Design of Urban Traffic-Light Control Systems Based on Synchronized Timed Petri Nets

Yi-Sheng Huang, *Senior Member, IEEE*, Yi-Shun Weng, and MengChu Zhou, *Fellow, IEEE*

***Abstract*—Timed Petri nets (TPNs) have been utilized as visual formalism for the modeling of complex discrete-event dynamic systems. They illuminate the features in describing the properties of causality and concurrency. Moreover, it is well known that a synchronized TPN (STPN) allows us to present all of the concurrent states in a complex TPN. In this paper, we propose a new methodology to design and analyze an urban traffic network control system by using the STPN. In addition, the applications of the STPN to eight-phase, six-phase, and two-phase traffic-light control systems are modularized. The advantage of the proposed approach is the clear presentation of the behaviors of traffic lights in terms of the conditions and events that cause phase alternations. Moreover, the size of the urban traffic network control system can be easily extended with our proposed modular technique. An analysis of the control models is performed via a reachability graph method to demonstrate how the models enforce the transitions of the traffic lights.**

***Index Terms*—Modular, Petri net (PN), traffic control systems, traffic network.**

I. INTRODUCTION

WITH the growing number of vehicles, traffic congestion and transportation delay in urban arterial roads are increasing worldwide; hence, it is imperative to improve the safety and efficiency of transportation. Subsequently, several research teams focus their attention on the area of intelligent transport systems [14], [20]. In addition, they apply advanced communication, information, and electronics technology to solve transportation problems such as traffic congestion, and to improve transportation safety and efficiency [5]. Traffic light control systems regulate, warn, and guide transportation for the purpose of improving the safety and efficiency of pedestrians and vehicles. Much work has been done to develop various strategies [19], which are classified into two categories: fixed-time and traffic-response. Nowadays, most industrialized countries are using the former for urban traffic control. In addition, the topics of traffic signal control can be separated into two classes [15]: determining which signal indication sequence optimizes the overall system performance and ascertaining how to implement the signal control logic.

However, several fundamental problems remain open, among which are the modeling methodologies. This paper concentrates on the second class, with a predetermined traffic signal timing plan. In other words, traffic signals are used to manage conflicting requirements for the use of road space, which are often at road junctions, by allocating the right-of-way to different sets of mutually compatible traffic movements during distinct time intervals. There is only one type of phase transition discussed in an urban net [15]. The variation of a vehicle's direction at a four-way intersection has two phases [22], four phases [24], or eight phases [15]. Febbraro *et al.* [6] proposed an urban traffic controller including four-phase and three-phase transitions. However, they failed to address the complicated movement direction of the traffic flow at intersections. For example, a traffic light not only has red, yellow, and green lights, but it also has a left-turn arrow on green, a right-turn arrow on green, and a straight arrow on green. As a result, it is possible to have both a right-turn arrow on green and a red signal turned on in an eastbound traffic light, and a green signal turned on in the northbound traffic light. In some countries, there may be additional lights, i.e., usually a green arrow to authorize turns (called a lead light in the U.S. because it usually leads to the main green light). Traffic light control systems regulate, warn, and guide transportation for the purpose of improving the safety and efficiency of pedestrians and vehicles. Obviously, an unsuitable modeling tool could make it difficult to evaluate the performance indexes of traffic control systems, such as the number of vehicles that are stopped by red signals and the subsequent time loss due to them.

Petri nets (PNs) have been proven to be a powerful modeling tool for various kinds of discrete-event systems [7], [17], and their formalism provides a clear means for presenting simulation and control logic. Hence, they are suitably used in traffic control systems. However, conventional PNs cannot determine the exact time of transition firing. To enhance their capability, timed PNs (TPNs) are proposed. Recently, they have been successfully used to model railway level crossing [3], [9], [18] and urban traffic network control systems [10]–[12], [23]. Moreover, timed colored PNs (TCPNs) are utilized as visual formalism for the modeling of complex systems. Our prior work used a TCPN to model an intelligent urban traffic-light control system [8], [12]. In summary, there are several modeling

Manuscript received March 28, 2013; revised July 3, 2013; accepted September 2, 2013. Date of publication November 12, 2013; date of current version March 28, 2014. This work was supported in part by the National Science Council of Taiwan under Grant NSC 102-2221-E-197-022-MY2 and National Natural Science Foundation of China, under Grant 61374148. The Associate Editor for this paper was Z. Li.

Y.-S. Huang is with the Department of Electrical Engineering, National Ilan University, Ilan 260, Taiwan (e-mail: yshuang@niu.edu.tw).

Y.-S. Weng is with the Department of Electronic Engineering, Army Academy Republic of China, Zhongli 32093, Taiwan (e-mail: wys@aaroc.edu.com).

M. Zhou is with the Key Laboratory of Embedded System and Service Computing, Ministry of Education, Tongji University, Shanghai 201804, China and also with the Department of Electrical and Computer Engineering, Newark College of Engineering, New Jersey Institute of Technology, Newark, NJ 07102-1982 USA (e-mail: zhou@njit.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TITS.2013.2283034

tools applied in the traffic-light control field. However, they hardly present the alternation of the traffic-light phases in a clear manner. Therefore, this paper focuses on the use of a synchronized TPN (STPN) to model an urban traffic-light control system that may have eight-phase, six-phase, and/or two-phase traffic lights. The advantage of the new approach is that the behavior of the traffic lights in terms of the conditions and events that cause phase alternations can be presented clearly. Reachability graphs are adopted to analyze the proposed STPN models. More significantly, we propose a new method of modular design. It subdivides a system into smaller parts that can be independently created and then used in different systems to drive multiple functionalities. Its advantages include the ease of change to achieve technology transparency and the use of industry standards to design key interfaces.

Earlier contributions focused on finding optimal control strategies [2], [15]. A variety of mathematical programming methodologies [2], [4] and artificial intelligence techniques [1], [16] to model the traffic flow and the control logic are proposed. Nevertheless, they need to translate their control logic to computer codes. Unfortunately, previous work did not address the issue of how to implement signal control logic [15]. The proposed STPN can overcome this drawback.

The rest of this paper is organized as follows. Section II gives a brief description of STPN. Section III provides the description of an urban traffic network system. Section IV presents our STPN-based modular design technique. Section V depicts how to model and analyze it with STPN. Conclusions are given in Section VI.

II. Basics of STPNs

A classical PN is a particular kind of bipartite directed graphs populated by three types of objects [25]–[27]. They are the places, transitions, and directed arcs that connect places to transitions and transitions to places. Because the classical PN is not capable of handling quantitative time, a timing concept is introduced to its transitions, thereby leading to TPNs [25]. A transition fires by removing tokens from its input places, and after a certain period (time delay), it deposits the tokens to its output places. In a synchronized PN, an event is associated with each transition, and the firing of this transition will occur, i.e., if the transition is enabled when the associated event occurs [21]. Therefore, synchronized PNs are well suited for the modeling of the systems that are synchronized with external events at a hierarchical level. In this paper, we combine the TPN and synchronized PN models, resulting in STPN. STPN allows two types of transitions: the immediate transition, which is represented by a thin bar and is also called a slave transition, whose firing takes no time, and the deterministic transition, which is represented by an empty bar and is also called a master transition, whose firing takes a constant delay time. In a master module, an event is associated with a master transition. Therefore, once an external event happens, its corresponding slave transition fires in the slave modules.

Formally, STPN can be defined as

$$\text{STPN} = (P,\, T,\, I,\, O,\, M_0,\, \tau,\, E,\, H)$$

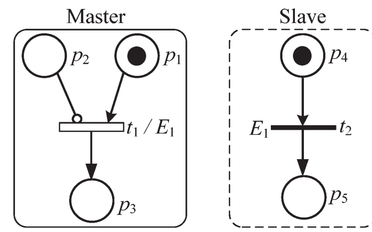


Fig. 1. STPN with an external event and an inhibitor arc.

where

$P=\{p_1,\, p_2, \ldots, p_m\}$	a finite set of places;
$T=\{t_1,\, t_2, \ldots, t_n\}$	a finite set of transitions, which are partitioned into two disjoint sets T^0 and T^+ that represent the immediate and deterministic transitions, respectively. $P \cup T \neq \phi$, and $P \cap T = \phi$;
$I : P \times T \rightarrow N$	input function that defines the directed arcs from places to transitions, where N is a set of nonnegative integers;
$O : P \times T \rightarrow N$	output function that defines the directed arcs from transitions to places;
$M_0 : P \rightarrow N$	initial marking;
$\tau : T \rightarrow R^+$	firing time function, where R^+ is the set of nonnegative real numbers;
$E=\{E_1,\, E_2, \ldots, E_i\}$	set of events associated with the transitions;
$H \subseteq P \times T$	set of inhibitor arcs from p to t.

Enabling Rule: A transition t is enabled at marking M if $\forall\, p \in P$, $M(p) \geq I(p,t)$ and $M(p) = 0$ if $(p,t) \in H$.

Firing Rule: First, an enabled transition t may or may not fire depending on the additional interpretation. Second, firing t removes $I(p,\, t)$ tokens from each input place p and deposits $O(p,\, t)$ tokens to each output place p of t. Finally, a transition in the slave can fire if and only if the corresponding external event occurs, and each of its preset contains sufficient tokens.

The liveness of a PN means that, for each marking $M \in R(M_0)$ that is reachable from M_0, it is finally possible to fire $t\ \forall t \in T$ through some firing sequence. Note that $R(M_0)$ is the set of all reachable markings from M_0. A marking M is reachable from M_0 if there is a fireable transition sequence that converts M_0 into M. A PN is said to be reversible if, for each marking $M \in R(M_0)$, M_0 is reachable from M. Thus, in a reversible net, it is always possible to go back to the initial marking (state) M_0.

An STPN with an inhibitor arc is shown in Fig. 1. In the presence of an inhibitor arc, a transition is regarded as enabled if input place p_1, which is connected to the transition by a normal arc (an arc that is terminated with an arrow), contains at least the number of tokens equal to the weight of the arc and if no tokens are present in input place p_2, which is connected to the transition by the inhibitor arc. Moreover, transition t/E_1 fires by removing a token from p_1, and it deposits the token to p_3 after a certain period τ. Event E_1 is associated with t_1 if t_1 fires and E_1 occurs simultaneously. In the slave side, t_2 is an immediate transition and is receptive to the same external event E_1. When E_1 occurs, t_2 fires by moving a token from p_4 to p_5 immediately.

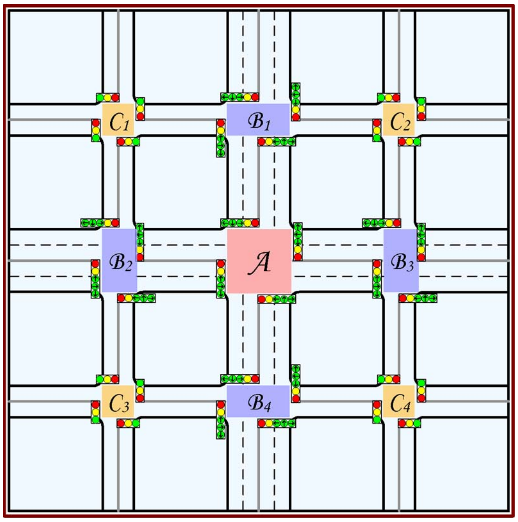


Fig. 2. Traffic network with nine intersections.

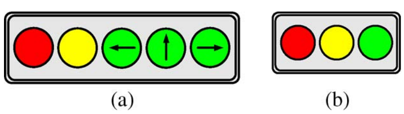


Fig. 3. (a) Five-light signal. (b) Three-light signal.

III. System Description and Methodology Development

The intersections are divided into three categories: main road, minor road, and tertiary road, which are called A, B, and C, respectively. In Fig. 2, only one intersection belongs to the main road intersection, four (B_1–B_4) belong to the minor one, and four (C_1–C_4) belong to the tertiary one. Here, two types of traffic lights are employed. The first type consists of five signal lights as shown in Fig. 3(a), i.e., a left-turn arrow on green (GL), a right-turn arrow on green (GR), a straight arrow on green (GS), a yellow light signal (Y), and a red light signal (R). There is either an eight-phase or six-phase transition available in this kind of traffic lights. The second type, as shown in Fig. 3(b), consists of three signal lights, i.e., G, Y, and R. Only two-phase transitions are needed. Considering the traffic flow, we assume that a traffic light has an eight-phase transition at A, a six-phase one at B, and a two-phase one at C.

This work proposes the following design procedure based on the idea of a modular design.

- *Inputs*: A traffic network with traffic-light requirements.
- *Outputs*: The modular design and implementation of a traffic-light control system.

1) Calculate how many traffic lights are needed in the traffic network system.
2) Identify the phase transition (i.e., two- phase, six-phase, and eight-phase transitions) for each traffic light.
3) Define the phase transition models for each phase transition.
4) Construct the STPN model for each phase transition.
5) Design the master–slave module for each phase transition.

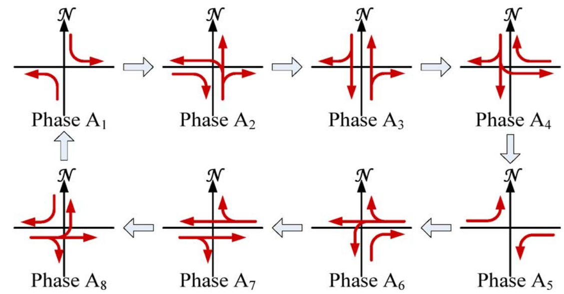


Fig. 4. Eight-phase transition and the phase alternating order.

6) Select one of the most complex modules (i.e., the eight-phase transition) as the master controller of the traffic-light control system.
7) Construct the traffic-light control system STPN model.
8) Analyze the control system STPN model.
9) Implement the traffic-light control system.

A. Phase Transition Models

Fig. 2 shows an urban traffic network with nine intersections that is taken from [13]. Consider the traffic area constructed in Fig. 2, which consists of intersections A, B, and C. The flows at A are ruled by an eight-phase traffic light. The rules are shown in Fig. 4(a).

1) Phase A_0. A traffic-light control system can be started if its traffic signal lights are all in a red state.
2) Phase A_1. GL signals turn on in the northbound and the southbound traffic lights. The R signals are displayed in the east–westward traffic directions.
3) Phase A_2. The northbound traffic-light signals GL, GR, and GS are on. The eastbound traffic lights indicate the GR signal. Notice that this phase is ignored in [6] and [24].
4) Phase A_3. Both the northbound and southbound traffic-light signals GS and GR are on.
5) Phase A_4. The southbound traffic-light signals GL, GR, and GS are on. At the same time, the GR signal is displayed in the westward traffic direction.

Notice that phases (A_5–A_8) are similar to phases (A_1–A_4), respectively (but in a reverse order). We assume that the phase transition time is 45 s for each phase transition. For convenience, we define "set cycle time" as the period for a set of traffic lights to operate in one cycle. Since there are nine sets of traffic lights in the example network, we further define "system cycle time" as the period for all sets to operate in one cycle. As a result, it needs 360 s for a set cycle time. It hints that phase A_1 should be changed to phase A_2 after 45 s.

Considering the second category of intersections, the vehicle flows at intersection B are required to be ruled by a six-phase traffic light. Fig. 5 shows the definitions of the phases and their alternating order.

1) Phase B_0. A traffic-light control system can be started if its traffic signal lights are all in the red state.
2) Phase B_1. GL, GS, and GL signals turn on in the northbound traffic light. The R signals are displayed in the other traffic directions.

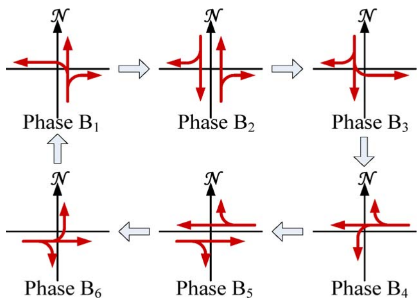


Fig. 5. Models of the six-phase transition.

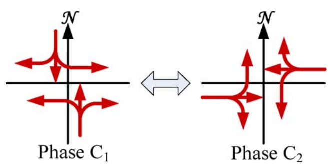
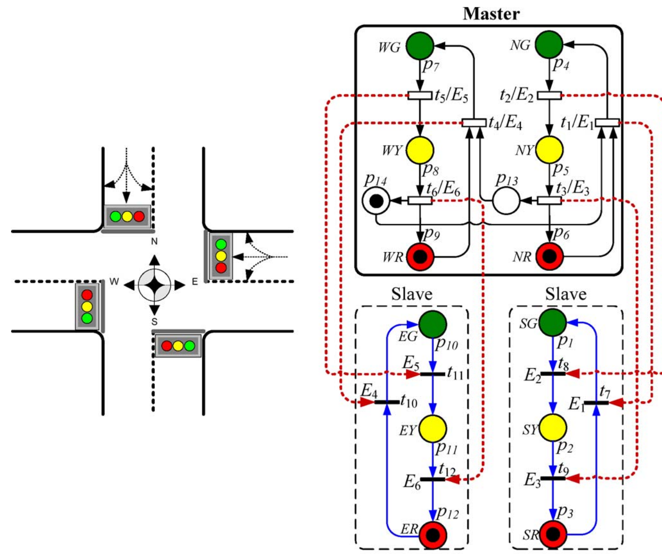


Fig. 6. Models of the two-phase transition.



(a) (b)

Fig. 7. (a) Two-phase traffic-light control system. (b) STPN model.

3) Phase B_2. Both the northbound and southbound traffic lights are showing GS and GR signals.
4) Phase B_3. GL, GS, and GL signals turn on in the southbound traffic lights. The R signals are displayed in the other traffic directions.

Finally, the two-phase transitions of the traffic lights that are located at intersections C are presented in Fig. 6.

1) Phase C_0. A traffic-light control system can be started if its traffic signal lights are all in the red state.
2) Phase C_1. Both the northbound and southbound traffic lights show G signals. This phase transition time is 90 s.
3) Phase C_2. Both the westbound and eastbound traffic lights show G signals. This phase transition time is also 90 s.

The three phase transition models illustrate the application of the urban traffic network control criteria.

B. STPN Model of an Example Traffic-Light Control System

A road signal for directing vehicular traffic using colored lights typically uses red for stop, green for go, and yellow for proceeding with caution. A traffic light with the three colored lights is called a two-phase traffic light. A two-phase system model is depicted in Fig. 7(a). Considering the safety of vehicles, some important rules are needed.

1) A traffic-light control system can be started if its traffic signal lights are all in a red light state.
2) A 2-s overlap is needed while the phase is changing from northward/southward to eastward/westward.
3) Only one green light is allowed on the two ways simultaneously.
4) A traffic light changes in the order of green, yellow, and red.

According to the specification of the traffic-light system, the STPN model of Fig. 7(a) can be constructed in Fig. 7(b). It describes a two-phase traffic-light control system with three traffic signal lights, i.e., green (G), yellow (Y), and red (R). More detailed information of Fig. 7(b) is listed in Table I.

TABLE I
ATTRIBUTION OF TRANSITIONS IN FIG. 7

t_i	Type	Parameter	value	units
t_1	deterministic	τ_1/E_1	2	*sec*
t_2	deterministic	τ_2/E_2	85	*sec*
t_3	deterministic	τ_3/E_3	3	*sec*
t_4	deterministic	τ_4/E_4	2	*sec*
t_5	deterministic	τ_5/E_5	85	*sec*
t_6	deterministic	τ_6/E_6	3	*sec*
t_7	immediate	E_1	0	*sec*
t_8	immediate	E_2	0	*sec*
t_9	immediate	E_3	0	*sec*
t_{10}	immediate	E_4	0	*sec*
t_{11}	immediate	E_5	0	*sec*
t_{12}	immediate	E_6	0	*sec*

Regularly, there are four sets of traffic lights in a system, i.e., the northbound, southbound, eastbound, and westbound traffic lights, which are placed at each intersection. Vehicles heading northward/southward and heading eastward/westward are denoted by N/S and E/W, respectively. The system model is divided into two parts, i.e., the master and slave models. A master model consists of places p_4/p_7, p_5/p_8, and p_6/p_9, and their associated arcs and transitions. A slave model consists of places p_1/p_{10}, p_2/p_{11}, and p_3/p_{12}, and their associated arcs and transitions. For convenience, p_4/p_7, p_5/p_8, and p_6/p_9 are given physical names, i.e., NG/WG, NY/WY, and NR/WR, respectively. They represent three northward/westward traffic lights G, Y, and R, respectively. Similarly, p_1/p_{10}, p_2/p_{11}, and p_3/p_{12} are given names, i.e., SG/EG, SY/EY, and SR/ER, respectively. They represent three southward/eastward traffic

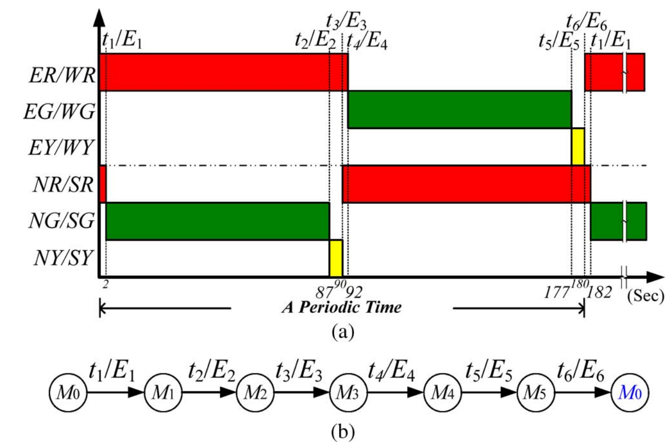


Fig. 8. (a) Duration diagram. (b) Reachability graph.

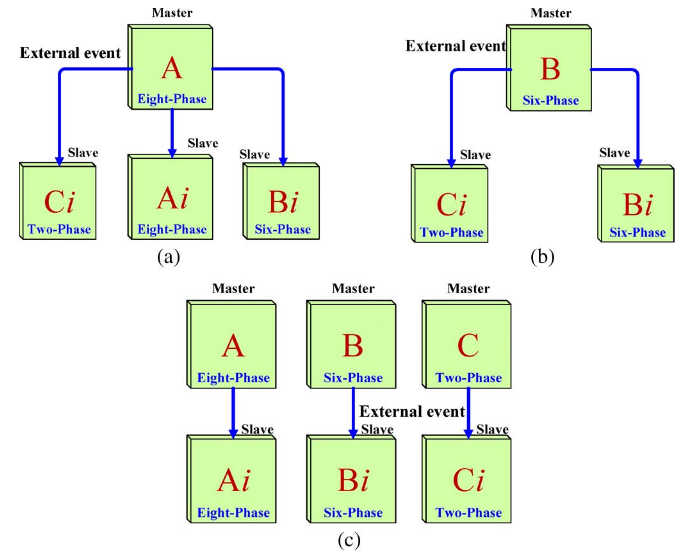


Fig. 9. All possible combinations for the master–slave traffic light controller.

lights G, Y, and R, respectively. The operations of the slave model are controlled by the master model via external events. For example, when t_1/E_1 fires (in the master model), then t_7 can be fired in the slave model.

Fig. 7(b) shows the initial state of the system model. To fit a physical system, it is natural to consider an upper limit of the number of tokens that each place can hold. This leads to a finite capacity net. Here, we apply the strict transition rule to our proposed models. For instance, Fig. 7(b) is a finite capacity net $(N,\ M_0)$, and each place p has an associated capacity $K(p) = 1$ [17]. We know that the initial state of traffic signal lights is in the R state. After 2 s (i.e., firing t_1/E_1), a token is moved to p_4. In the meantime, t_7 immediately fires as triggered by E_1 such that a token is moved to p_1 immediately. At this moment, NG turns on such that the way of vehicles can pass through the intersection. Next, NG should be off after the duration in t_2/E_2. Notice that event E_2 is associated with t_2. Therefore, E_2 will concurrently occur with the firing of t_2. It states that t_8 fires when E_2 occurs. It implies that SG has been on for 85 s. Then, NY is on for 3 s because the duration in t_3/E_3 is 3 s. E_3 is associated with t_3. Therefore, E_3 will occur concurrently. It states that t_9 fires when E_3 occurs. Then, the token is moved to place p_6/p_3 again. After 2 s (i.e., firing t_4/E_4), a token in p_9/p_{12} is moved to p_7/p_{10}. At this moment, WG/EG turns on. It means that NW/EG has been on for 85 s. Then, WY is on for 3 s because the duration in t_6/E_6 is 3 s. Finally, the token is moved to p_9/p_{12} again. According to the model, one can well describe the periodic property of the traffic-light control system. Fig. 8(a) shows its control process. It helps us understand the behavior of the system model.

A reachability graph can be generated to analyze the model. No conflict can be found, as shown in Fig. 8(b), as verified by its reachability graph. The reachability set $R(M_0) = \{M_0,\ M_1,\ M_2,\ M_3,\ M_4,\ M_5\}$ contains all the markings that are reachable from M_0. Here, $M_0 = (0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 1)$, $M_1 = (1, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0)$, $M_2 = (0, 1, 0, 0, 1, 0, 0, 0, 1, 0, 0, 1, 0, 0)$, $M_3 = (0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 1, 1, 0)$, $M_4 = (0, 0, 1, 0, 0, 1, 1, 0, 0, 1, 0, 0, 0, 0)$, and $M_5 = (0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0)$. Since this graph is a finite circuit containing all transitions, one can infer that the STPN model in Fig. 7(b) is live and reversible.

IV. STPN-Based Modular Design Technique for Master–Slave Traffic Signal Controller

In order to be able to integrate the different phases of traffic lights in different roadway intersections, all traffic lights (two-phase, six-phase, and eight-phase traffic lights) can independently operate and can be associated with each other. Fig. 9 shows all possible combinations for the master–slave traffic-light controller. For example, at a traffic network with three different intersections in Fig. 9(a), the most complex traffic light (i.e., the eight-phase traffic light) will be the master controller, and the others will be the slaves. If an urban traffic network only has the same phase traffic-light controller, all must serve as the master controller as shown in Fig. 9(c). In the proposed method, we need one model as the master that is able to trigger the slave models. Here, we leave out the details in Fig. 9(b). In addition, three capabilities offered by STPN are listed as follows.

1) *Master–slave configuration.* It is well known that STPN has external events that allow one to model the relation of the concurrence transitions, whatever the related transitions are in different levels. Therefore, we propose that the master–slave configuration is used to model the urban traffic-light system. Its advantage is that all external events (i.e., the transitions of the master) are able to trigger the corresponding transitions of the slaves.
2) *Timing specifications.* STPN is capable of handling quantitative time to model the timing specification in subnets. It is worth noting that such timing constraints can appear in subnets at any levels.
3) *Synchronization descriptions.* The ability of STPN to describe synchronization is crucial. In STPN, an event is associated with specific transitions. Once an event occurs, its corresponding transitions will fire. In Fig. 1, one can recall that t immediately fires when its external event E occurs. This representation is appropriate to model the occurrence transitions that depend on external events. Fig. 9 shows the relation of the traffic-light control system models with external events.

A. Master–Slave Module of Eight-Phase Traffic-Light STPN Model

All external events (i.e., E_i) are designed to connect to their corresponding pins. Hence, a master–slave module of the eight-phase traffic-light STPN model that is designed in an IC style is depicted in Fig. 10. Here, eight external events (i.e., E_1–E_8) are laid out in Fig. 10. This type of controllers can be only used in the main street intersection (i.e., the eight-phase traffic lights). We name it as Class A. It is the key to the urban traffic-light control system. There are four parts of traffic lights facing the four directions of an intersection. For example, the traffic light positioned at the southbound of intersection A is called the northbound traffic light, which is denoted as N_A. The others are S_A, W_A, and E_A for the southbound, westbound, and eastbound traffic lights, respectively. Note that each of them has five light signals.

For convenience, only the places that are modeling the traffic-light signals are included in the elements of a marking. In this master–slave model, the marking (ANR, ASR, AWR, AER) is called initial state M_{a0}, which means that the four-part traffic lights are signaling R. The initial state will hold for 2 s, i.e., four parts of the traffic light signals stay red for 2 s simultaneously. Once t_1/E_1 reaches 2 s, the initial marking configuration changing to $(ANGL \wedge ANR, ASGL \wedge ASR, AWR, AER)$ is called M_{a1}. Notice that, at the same time, the transitions of two slave models t_{19}/t_{33} in Figs. 12 and 14 immediately fire when E_1 occurs, such that marking M_{b0}/M_{c0} changes to M_{b1}/M_{c1}. In the meantime, phases A_1, B_1, and C_1 are derived. Once t_2 has been firing for 40 s, M_{a1} changes to $(ANGL \wedge ANR, ASY \wedge ASR, AWR, AER)$, which is called M_{a2}. During this period, the marking $ASGL$ of S_A is changed to ASR via ASY, which is called transient marking M_{a2}. It means that the left-turn arrow is green for 40 s, and then, the yellow light signal turns on for 3 s. Here, such kind of markings is called transient markings. The other markings involving (ANY, ASY, AWY, AEY) are also regarded as transient markings since one of the elements of the marking is a yellow light. When t_3 fires, the transient marking is changed to M_{a3}. Once t_4/E_2 fires, M_{a3} should immediately change to $(ANGS \wedge ANGR, ASGS \wedge ASGR, AWR, AER)$, which is called marking M_{a4}. Once t_5/E_3 fires, M_{a4} should immediately change to $(ANY, ASGS \wedge ASGR, AWR, AER)$, which is called the transient marking M_{a5}. It is worth noticing that a 2 s overlap between the phase changes is needed for safety reasons. The transient marking M_{a5} changes to $(ANR, ASGS \wedge ASGR, AWR, AER)$, which is called transition marking M_{a6}. After 2 s (firing t_7), M_{a6} should immediately change to $(ANR, ASGL \wedge ASGS \wedge ASGR, AWR \wedge AWGR, AER)$, which is called marking M_{a7}. After 42 s (firing t_8/E_4), M_{a7} should change to the transient marking (ANR, ASY, AWR, AER), which is called M_{a8}. Finally, M_{a8} should change to the initial marking M_{a0}. Phases A_5–A_8 proceed in the same way.

For clarity, a reachability graph of the STPN is depicted in Fig. 11. First, M_{a0} should be changed to M_{a1} when t_1/E_1 fires. In the meantime, the new marking should be held for 40 s. Then, the marking is changed to a transient marking, i.e., M_{a2}. When t_3 fires, M_{a2} is changed to M_{a3}. After 42 s (firing t_4/E_2), a new event E_2 is generated, and M_{a3} changes to M_{a4} and so forth. The reachability set $R(M_{a0}) = \{M_{a0}, M_{a1}, M_{a2}, M_{a3}, M_{a4}, M_{a5}, M_{a6}, M_{a7}, M_{a8}, M_{a9}, M_{a10}, M_{a11}, M_{a12}, M_{a13}, M_{a14}, M_{a15}, M_{a16}\}$ contains all the markings that are reachable from M_{a0}. As a result, we can conclude that the eight-phase model is live and reversible.

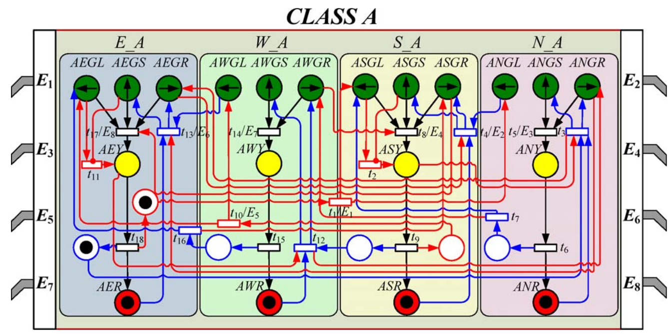


Fig. 10. Master–slave module of the eight-phase traffic-light STPN model.

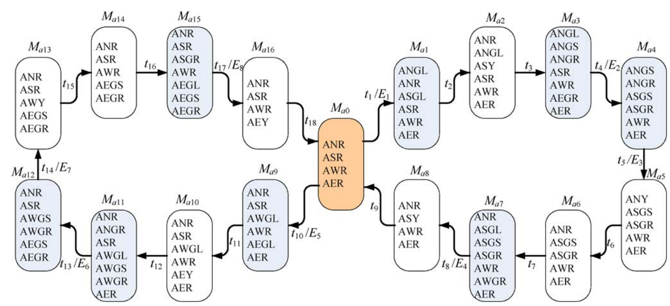


Fig. 11. Reachability graph of the eight-phase master–slave STPN model.

B. Master–Slave Module of the Six-Phase Traffic-Light STPN Model

One has seen that the eight-phase traffic-light STPN model can be designed in an eight-pin IC style. Similarly, a master–slave model of the six-phase traffic-light controller is constructed and is designed in a 16-pin IC style, as shown in Fig. 12. This type of controller can be only employed in the minor street intersection (six-phase traffic lights). Here, we name it as Class B. In details, the master–slave model consists of four parts: N_B, S_B, W_B, and E_B. They correspond to the four ways of traffic lights.

It is very important to notice that some external events are related to the eight-phase master–slave STPN model. We will use the external events as the path of the signal communication between the master and slave controllers. According to the preceding discussion, the six-phase traffic-light STPN model should change from M_{b0} to M_{b1}. Fig. 13 depicts the more detailed operations of the STPN model in a reachability graph.

It includes M_{b0}, six phase markings (M_{b1}–M_{b2}, M_{b5}, M_{b7}–M_{b8}, and M_{b11}), and six transient ones (M_{b3}–M_{b4}, M_{b6}, M_{b9}–M_{b10}, and M_{b12}). $R(M_{b0}) = \{M_{b0}, M_{b1}, M_{b2}, M_{b3}, M_{b4}, M_{b5}, M_{b6}, M_{b7}, M_{b8}, M_{b9}, M_{b10}, M_{b11}, M_{b12}\}$ contains all the markings reachable from M_{b0}. The six-phase master–slave STPN model is live and reversible.

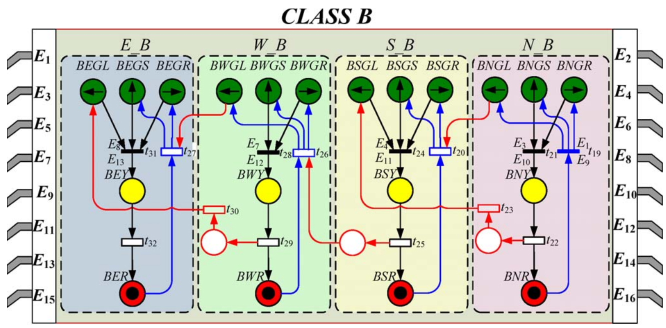


Fig. 12. Master–slave module of the six-phase traffic-light STPN model.

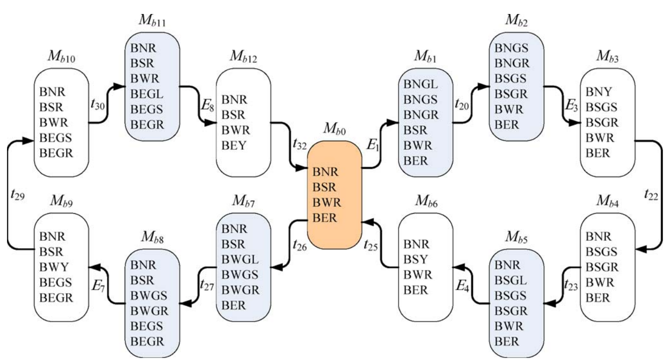


Fig. 13. Reachability graph of the six-phase master–slave STPN model.

C. Master–Slave Module of the Two-Phase Traffic-Light STPN Model

Next, we demonstrate the development the master–slave STPN model for two-phase traffic lights by a compact way. This kind of traffic lights is widely used at intersections of minor importance. The model of the two-phase traffic-light system is shown in Fig. 14. This kind of controllers can be only used in tertiary street intersections (i.e., the two-phase traffic lights). Here, we name it as Class C, as shown in Fig. 14. Likewise, a reachability graph of the master–slave STPN model is obtained, as shown in Fig. 15. It includes M_{c0}, two phase markings (i.e., M_{c1} and M_{c3}), and two transient ones (i.e., M_{c2} and M_{c4}). $R(M_{c0}) = \{M_{c0}, M_{c1}, M_{c2}, M_{c3}, M_{c4}\}$. As a result, we can infer that the two-phase model is live and reversible.

It is worth noting that the two-phase traffic light needs 180 s for a set cycle time. On the other hand, the system cycle of the urban traffic network system needs 360 s. Hence, the external event $(E_2/E_4)/(E_6/E_8)$ can bring the transient marking M_{c2}/M_{c4} to the initial marking M_{c0}, respectively.

V. Modeling and Analysis of Urban Traffic Network Controllers

In this section, we attempt to control an urban traffic network that consists of nine intersections. As aforementioned, the nine intersections are divided into three groups, i.e., main street, minor street, and tertiary street. To model the traffic network controllers of the nine intersections, one only needs one master eight-phase one, four slave six-phase one, and four two-phase ones. With a conventional methods, it is challenging to design a mechanism for coordinating the firing sequences of the traffic

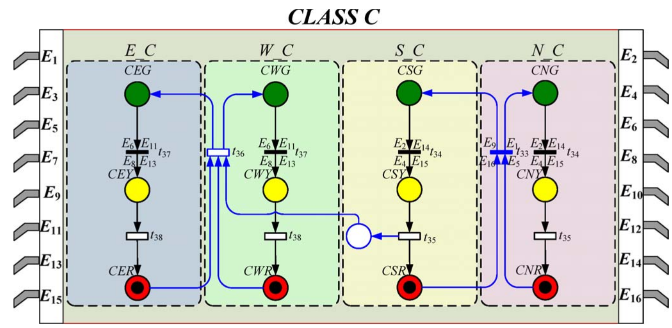


Fig. 14. Master–slave module of the two-phase traffic-light STPN model.

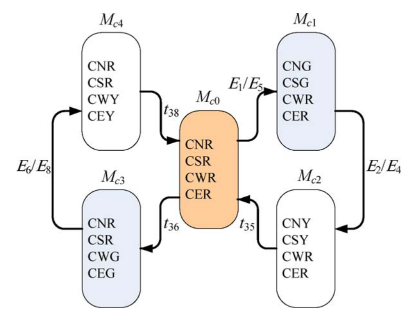


Fig. 15. Reachability graph of the two-phase master–slave STPN model.

lights as desired. However, it now becomes an easy job due to the developed modules.

A. Urban Traffic-Light Controller Designed by STPN Modular Model

Our proposed traffic network consists of three different types of intersections. We can construct the configuration of the master–slave modular models in Fig. 16, which allows one to have an overview of the entire urban traffic network system. The external events are connected according to their corresponding pins. In this urban traffic control system, one Class A, four Class B, and four Class C controllers are used for the main, minor, and tertiary street intersections, respectively. Here, Class A is the only master controller, whereas others are slaves. In detail, master A is decomposed into four parts, i.e., N_A, S_A, W_A, and E_A, as shown in Fig. 10. Likewise, slave models B_{1-4} and C_{1-4} are decomposed into four parallel behavioral components, respectively. The more detailed information of the slave models is shown in Figs. 11 and 12. It is worth noting that, when an external event occurs, it immediately causes the corresponding transition to fire. Here, we define the plant of the control system as the part of the system that is to be controlled. Traffic light statuses are changed by firing actions/triggering events. As the control objective, we want the system output (i.e., traffic-light signals) to follow the current phase state. In this way, each of the operation sequences can be regarded as a concatenation of operations that are performed at the traffic-light network system. The detailed information for the external events in Fig. 16 is given in Table II.

Fig. 17 helps one understand the operation of transition firing and event occurrences. We use three states (ANR, B_iNR, and C_iNR) to represent the part of the initial configuration. The traffic-light system can stay at the initial state

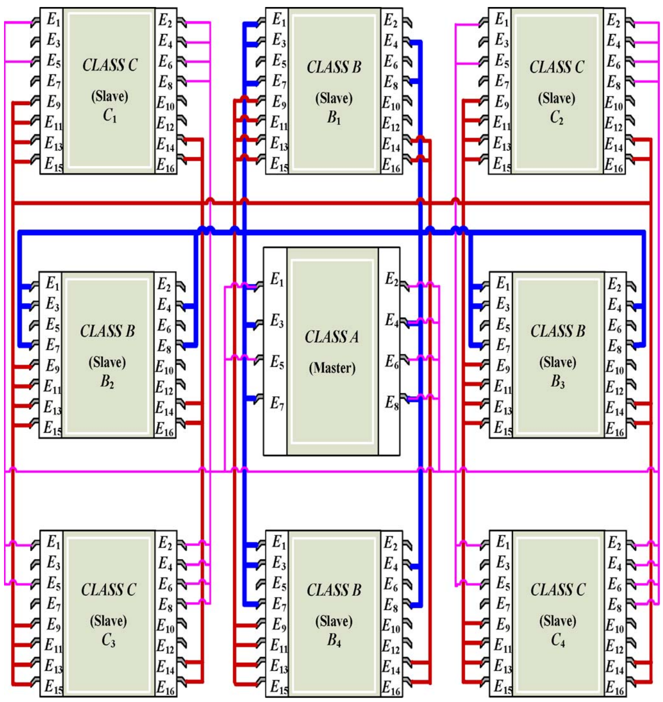


Fig. 16. Urban traffic network control system STPN model.

TABLE II
DETAILED INFORMATION OF THE EXTERNAL EVENTS IN FIG. 16

Master *CLASS A*	Slave *CLASS $B_{1\text{-}4}$*	Slave *CLASS $C_{1\text{-}4}$*
t_1/E_1	E_1/t_{19}	E_1/t_{33}
t_4/E_2		E_2/t_{34}
t_5/E_3	E_3/t_{21}	
t_8/E_4	E_4/t_{24}	E_4/t_{34}
t_{10}/E_5		E_5/t_{33}
t_{13}/E_6		E_6/t_{37}
t_{14}/E_7	E_7/t_{28}	
t_{17}/E_8	E_8/t_{31}	E_8/t_{37}

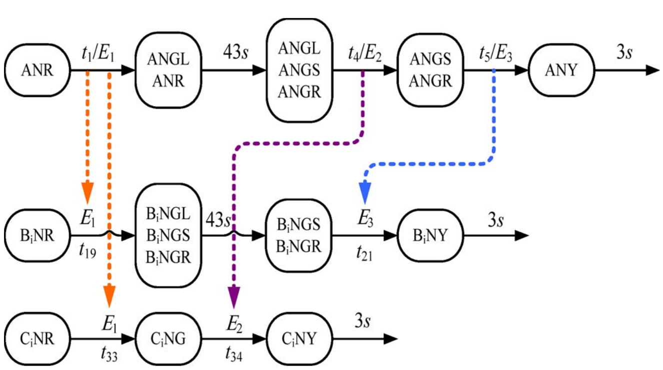


Fig. 17. Parts of the phase transitions of the controller.

(i.e., all the R signals are on). When t_1 has fired for 2 s, event E_1 is triggered such that the triggered transitions t_{19} and t_{33} fire immediately. Hence, event t_1/E_1 causes the initial state configuration (ANR, B_iNR, C_iWR) to change to ($ANGL$, ANR, $B_iNGL \wedge B_iNGS \wedge B_iNGR$, C_iNG). In the meantime, all the statuses of the traffic network are

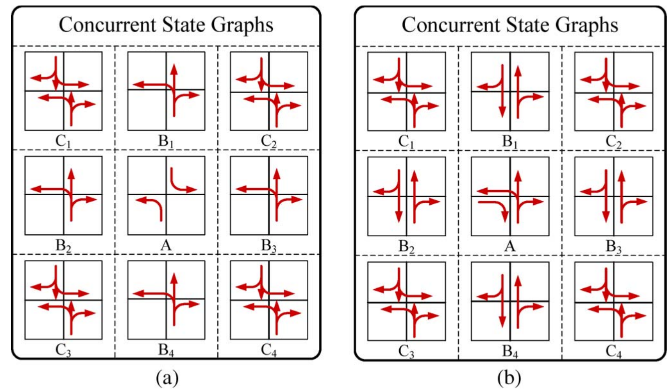


Fig. 18. Phase transition of the traffic network control system. (a) $Phase\_1$. (b) $Phase\_2$.

shown in Fig. 18(a). After 43 s, the state configuration ($ANGL$, ANR, $B_iNGL \wedge B_iNGS \wedge B_iNGR$, C_iNG) can be then changed to ($ANGL \wedge ANGS \wedge ANGR$, $B_iNGS \wedge B_iNGR$, C_iNG). At this moment, the nine pictures in Fig. 18(b) reflect the phase transitions of the traffic lights at nine intersections. Additionally, when t_4/E_2 has fired for 42 s, E_2 occurs such that t_{34} fires immediately. Because both state transitions are changed at the same time, the final state configuration becomes ($ANGS \wedge ANGR$, $B_iNGS \wedge B_iNGR$, C_iNY) after t_4/E_2 fires.

B. Analysis of Urban Traffic-Light Controller

Here, we attempt to analyze the urban traffic network. Based on the foregoing discussion, we know that the master–slave controller of the urban traffic system are all concurrent states. However, it is difficult to present all concurrent states together. Therefore, we try to construct the phase status configurations of the system model that contains the status of traffic-light signal Y called the transient state to present all the concurrent states. Here, an occurrence state shown in Table III is obtained from the urban traffic system state configuration. The occurrence state generates 17 phases in a system cycle. Each phase consists of nine elements, which are for the nine intersections of the urban traffic network. For example, the initial state, i.e., $phase\ state\_0 = (A, B_1, B_2, B_3, B_4, C_1, C_2, C_3, C_4)$, means that the states of the traffic lights at intersections A, B_1–B_4, and C_1–C_4 are in their initial states. It needs 360 s for the STPN to return to its initial state. It means that the urban traffic-light system needs 360 s for the system to return to the initial state. More precisely, the two-phase, six-phase, and eight-phase transitions need 180, 360, and 360 s for each set cycle time, respectively. Therefore, the eight-phase transition needs 45 s for each phase transition time.

Considering the STPN model of the urban traffic controller, the initial status configuration is represented by $phase\ state\_0$. It means that all the traffic lights signal R due to the initial status. According to the foregoing discussion, the system should be changed from $phase\ state\_0$ to $phase\ state\_1$ when event E_1 occurs. More precisely, $phase\ state\_1$ should be changed to $transient\ phase\ state\_2$ after 40 s. We have to emphasize that all transient states should be held active for

TABLE III
PHASE STATUS CONFIGURATIONS OF THE MODEL

Phase	Status configuration (N_A, S_A, W_A *and* E_A) of the intersections A, B_i and C_i (i.e. i = 1, 2, 3, 4)
State_0	(ANR, ASR, AWR, AER) , $(B_iNR, B_iSR, B_iWR, B_iER)$, $(C_iNR, C_iSR, C_iWR, C_iER)$
State_1	$(ANR, ANGL, ASR, ASGL, AWR, AER)$, $(B_iNGL, B_iNGS, B_iNGR, B_iSR, B_iWR, B_iER)$, $(C_iNG, C_iSG, C_iWR, C_iER)$
State_2	$(ANR, ANGL, ASR, ASY, AWR, AER)$, $(B_iNGL, B_iNGS, B_iNGR, B_iSR, B_iWR, B_iER)$, $(C_iNG, C_iSG, C_iWR, C_iER)$
State_3	$(ANGL, ANGS, ANGR, ASR, AWR, AEGR, AER)$, $(B_iNGS, B_iNGR, B_iSGS, B_iSGR, B_iWR, B_iER)$, $(C_iNG, C_iSG, C_iWR, C_iER)$
State_4	$(ANGS, ANGR, ASGS, ASGR, AWR, AER)$, $(B_iNGS, B_iNGR, B_iSGS, B_iSGR, B_iWR, B_iER)$, $(C_iNY, C_iSY, C_iWR, C_iER)$
State_5	$(ANGS, ANGR, ASGS, ASGR, AWR, AER)$, $(B_iNGS, B_iNGR, B_iSGS, B_iSGR, B_iWR, B_iER)$, $(C_iNR, C_iSR, C_iWR, C_iER)$
State_6	$(ANY, ASGS, ASGR, AWR, AER)$, $(B_iNY, B_iSGS, B_iSGR, B_iWR, B_iER)$, $(C_iNR, C_iSR, C_iWG, C_iEG)$
State_7	$(ANR, ASGS, ASGR, AWR, AER)$, $(B_iNR, B_iSGS, B_iSGR, B_iWR, B_iER)$, $(C_iNR, C_iSR, C_iWG, C_iEG)$
State_8	$(ANR, ASGL, ASGS, ASGR, AWGR, AWR, AER)$, $(B_iNR, B_iSGL, B_iSGS, B_iSGR, B_iWR, B_iER)$, $(C_iNR, C_iSR, C_iWG, C_iEG)$
State_9	(ANR, ASY, AWR, AER) , $(B_iNR, B_iSY, B_iWR, B_iER)$, $(C_iNR, C_iSR, C_iWY, C_iEY)$
State_10	$(ANR, ASR, AWR, AWGL, AER, AEGL)$, $(B_iNR, B_iSR, B_iWGL, B_iWGS, B_iWGR, B_iER)$, $(C_iNG, C_iSG, C_iWR, C_iER)$
State_11	$(ANR, ASR, AWR, AWGL, AER, AEY)$, $(B_iNR, B_iSR, B_iWGL, B_iWGS, B_iWGR, B_iER)$, $(C_iNG, C_iSG, C_iWR, C_iER)$
State_12	$(ANR, ANGR, ASR, AWGL, AWGS, AWGR, AER)$, $(B_iNR, B_iSR, B_iWGS, B_iWGR, B_iEGS, B_iEGR)$, $(C_iNG, C_iSG, C_iWR, C_iER)$
State_13	$(ANR, ASR, AWGS, AWGR, AEGS, AEGR)$, $(B_iNR, B_iSR, B_iWGS, B_iWGR, B_iEGS, B_iEGR)$, $(C_iNY, C_iSY, C_iWR, C_iER)$
State_14	$(ANR, ASR, AWGS, AWGR, AEGS, AEGR)$, $(B_iNR, B_iSR, B_iWGS, B_iWGR, B_iEGS, B_iEGR)$, $(C_iNR, C_iSR, C_iWR, C_iER)$
State_15	$(ANR, ASR, AWY, AEGS, AEGR)$, $(B_iNR, B_iSR, B_iWY, B_iEGS, B_iEGR)$, $(C_iNR, C_iSR, C_iWG, C_iEG)$
State_16	$(ANR, ASR, AWR, AEGS, AEGR)$, $(B_iNR, B_iSR, B_iWR, B_iEGS, B_iEGR)$, $(C_iNR, C_iSR, C_iWG, C_iEG)$
State_17	$(ANR, ASR, ASGR, AWR, AEGL, AEGS, AEGR)$, $(B_iNR, B_iSR, B_iWR, B_iEGL, B_iEGS, B_iEGR)$, $(C_iNR, C_iSR, C_iWG, C_iEG)$
State_18	(ANR, ASR, AWR, AEY) , $(B_iNR, B_iSR, B_iWR, B_iEY)$, $(C_iNR, C_iSR, C_iWY, C_iEY)$

3 s. Then, $transient\ phase\ state\_2$ should be changed into $phase\ state\_3$ after 3 s. The physical meaning is that the phase's change of traffic lights is needed via Y lights. For example, $phase\ state\_1\_A = (ANR, ANGL, ASR, ASGL, AWR, AER)$ is changed to $phase\ state\_2\_A = (ANR, ANGL, ASY, ASR, AWR, AER)$ and is then changed to $phase\ state\_3\_A = (ANGL, ANGS, ANGR, ASR, AWR, AER, AEGR)$. As aforementioned, we can infer that the occurrence state forms a cycle. As a result, the reversibility and liveness properties can be derived from the method.

VI. CONCLUSION

Modular design is basically a method that is used to allow a specific type of unit to be broken down into smaller and easier-to-manage units. It can be applied in numerous areas to create efficiency in other types of design, from manufacturing and construction to computer programming. In this paper, we have proposed the master–slave configuration to model an urban traffic-light system. The advantage of the proposed approach is the clear presentation of the system behavior and readiness for implementation. Moreover, the analysis of the control models is performed through a reachability graph method to demonstrate how the models enforce the transitions of the lights. Their liveness and reversibility are verified.

We believe that our research using STPN to model traffic-light systems will become more and more important in this field due to the increasing demands for more features on the traffic-light systems. The proposed STPN models can be extended for some other applications [28]–[34]. For example, our system model can be easily modified for an advanced traffic management system, which is able to interrupt the regularity of traffic lights if one allows emergency cars as the priority. The proposed method needs to be extended to deal with cases where the switching time of traffic lights is a function of the traffic intensity.

REFERENCES

[1] G. Abu-Lebdeh and R. F. Benekohal, "Genetic algorithms for traffic signal control and queue management of oversaturated two-way arterials," *J. Transp. Res. Board*, vol. 1727, no. 1, pp. 61–67, Jan. 2000.

[2] D. C. Chin, J. C. Spall, and R. H. Smith, "Evaluation of system-wide traffic signal control using stochastic optimization and neural networks," in *Proc. ACC*, Jun. 1999, vol. 3, pp. 2188–2194.

[3] S. C. Dutilleul, F. Deffossez, and P. Bon, "Safety requirements and p-time Petri nets: a level crossing case study," in *Proc. IEEE Multiconf. Comput. Eng. Syst. Appl./IMACS*, Beijing, China, Oct. 2006, pp. 1118–1123.

[4] V. Dinopoulou, C. Diakaki, and M. Papageorgiou, "Simulation investigations of the traffic-responsive urban control strategy TUC," in *Proc. IEEE Intell. Transp. Syst.*, Oct. 2003, pp. 458–463.

[5] L. Figueiredo, I. Jesus, J. A. Tenreiro Machado, J. R. Ferreira, and J. L. Martins de Carvalho, "Towards the development of intelligent transportation systems," in *Proc. IEEE Intel. Trans. Syst.*, Aug. 2001, pp. 1206–1211.

[6] A. D. Febbraro, D. Giglio, and N. Sacco, "Urban Traffic Control Structure Based on Hybrid Petri Nets," *IEEE Trans. Intell. Transp. Syst.*, vol. 5, no. 4, pp. 224–237, Dec. 2004.

[7] B. Hruz and M. C. Zhou, *Modeling and Control of Discrete Event Dynamic Systems*. London, U.K.: Springer-Verlag, 2007.

[8] Y. S. Huang and T. H. Chung, "Modeling and analysis of urban traffic lights control systems using timed CP-nets," *J. Inf. Sci. Eng.*, vol. 24, no. 3, pp. 875–890, 2008.
[9] Y. S. Huang, Y. S. Weng, and M. C. Zhou, "Critical scenarios and their identification in parallel railroad level crossing traffic control systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 4, pp. 968–977, Dec. 2010.
[10] Y. S. Huang and P. J. Su, "Modeling and analysis of traffic light control systems," *IET Control Theory and Applications*, vol. 3, no. 3, pp. 340–350, Mar. 2009.
[11] Y. S. Huang and T. H. Chung, "Modeling and Analysis of Air Traffic Control Systems Using Hierarchical Timed Coloured Petri Nets," *Trans. Inst. Meas. Control*, vol. 3, no. 1, pp. 30–49, 2009.
[12] Y. S. Huang and T. H. Chung, "Modeling and Analysis Urban Traffic Light Control Systems," *J. Chinese Inst. Eng.*, vol. 32, no. 1, pp. 85–95, 2009.
[13] Y. S. Huang, "Design of traffic light control system using statecharts," *Comp. J.*, vol. 49, no. 6, pp. 634–649, Nov. 2006.
[14] J. H. Lee and L. K. Hyung, "Distributed and cooperative fuzzy controllers for traffic intersection group," *IEEE Trans. Syst. Man. Cybern.*, vol. 29, no. 2, pp. 263–271, May 1999.
[15] G. F. List and M. Cetin, "Modeling traffic signal control using Petri nets," *IEEE Trans. Intell. Transp. Syst.*, vol. 5, no. 3, pp. 177–187, Sep. 2004.
[16] Q. Lin, B. W. Kwan, and L. J. Tung, "Traffic signal control using fuzzy logic," in *Proc. IEEE Syst., Man Cybern.*, Oct. 1997, pp. 1644–1649.
[17] D. Liu, Z. W. Li, and M. C. Zhou, "Hybrid Liveness-Enforcing Policy for Generalized Petri Net Models of Flexible Manufacturing Systems," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 43, no. 1, pp. 85–97, Jan. 2013.
[18] M. Medjoudj and P. Yim, "Extraction of critical scenarios in a railway level crossing control system," *J. Comput., Commun. Control*, vol. 2, no. 3, pp. 252–268, 2007.
[19] M. Papageorgiou, C. Diakaki, V. Dinopoulou, A. Kotsialos, and Y. Wang, "Review of road traffic control strategies," *Proc. IEEE*, vol. 91, no. 12, pp. 2043–2067, Dec. 2003.
[20] D. I. Robertson and R. D. Bretherton, "Optimizing networks of traffic signals in real time-the SCOOT method," *IEEE Trans. Veh. Technol.*, vol. 40, no. 1, pp. 11–15, Feb. 1991.
[21] D. Rene and A. Hassane, *Discrete, Continuous, and Hybrid Petri Nets*. Berlin, Germany: Springer-Verlag, 2005.
[22] C. Tolba, P. Thomas, A. Eimoudni, and D. Lefebvre, "Performances evaluation of the traffic control in a single crossroad by Petri nets," in *Proc. IEEE Emerging Technol. Factory Autom.*, Lisbon, Portugal, Sep. 2003, vol. 2, pp. 157–160.
[23] A. Tzes, K. Seongho, and W. R. McShane, "Application of Petri networks to transportation network modeling," *IEEE Trans. Veh. Technol.*, vol. 45, no. 2, pp. 391–400, May 1996.
[24] J. Wang, C. Jin, and Y. Deng, "Performance analysis of traffic networks based on stochastic timed Petri net models," in *Proc. IEEE Eng. Complex Comput. Syst.*, Oct. 1999, pp. 77–85.
[25] M. C. Zhou and K. Venkatesh, *Modeling, Simulation and Control of Flexible Manufacturing Systems: A Petri Net Approach*. Singapore: World Scientific, 1998.
[26] M. C. Zhou and F. DiCesare, *Petri Net Synthesis for Discrete Event Control of Manufacturing Systems*. London, U.K.: Kluwer, 1993.
[27] R. Zurawski and M. C. Zhou, "Petri nets and industrial applications: A tutorial," *IEEE Trans. Ind. Electron.*, vol. 41, no. 6, pp. 567–583, Dec. 1994.
[28] L. Qi, M. C. Zhou, and Z. Ding, "Real-time traffic camera-light control systems for intersections subject to accidents: A Petri net approach," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, Manchester, U.K., Oct. 2013, pp. 1069–1074.
[29] N. Wu, F. Chu, S. Mammar, and M. C. Zhou, "Petri net modeling of cooperation behavior of driver and co-pilot in advanced driving assistance systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 4, pp. 977–989, Dec. 2011.
[30] Y. Du, C. Jiang, and M. C. Zhou, "Modeling and analysis of real-time cooperative systems using Petri nets," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 37, no. 5, pp. 643–654, Sep. 2007.
[31] H. Hu, M. C. Zhou, and Z. Li, "Liveness enforcing supervision of video streaming systems using non-sequential Petri nets," *IEEE Trans. Multimedia*, vol. 11, no. 8, pp. 1446–1456, Dec. 2009.
[32] J. S. Lee, M. C. Zhou, and P. L. Hsu, "An application of Petri nets to supervisory control for human-computer interactive systems," *IEEE Trans. Ind. Electron.*, vol. 52, no. 5, pp. 1220–1226, Oct. 2005.
[33] G. Liu, C. Jiang, M. Zhou, and P. Xiong, "Interactive Petri nets," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 43, no. 2, pp. 291–302, Mar. 2013.
[34] N. Wu, F. Chu, C. Chu, and M. C. Zhou, "Petri net-based scheduling of single-arm cluster tools with reentrant atomic layer deposition processes," *IEEE Trans. Autom. Sci. Eng.*, vol. 8, no. 1, pp. 42–55, Jan. 2011.

Yi-Sheng Huang (SM'05) received the B.S. degree in automatic control engineering from Feng Chia University, Taichung, Taiwan, in 1989; the M.S. degree in electronic engineering from Chung Yuan Christian University, Zhongli, Taiwan, in 1991; and the Ph.D. degree in electrical engineering from National Taiwan University of Science and Technology, Taipei, Taiwan, in 2001.

He was a Professor with the Department of Electrical and Electronic Engineering, College of Science Engineering, Chung Cheng Institute of Technology, National Defense University, Daxi, Taiwan. He is currently a Full Professor with the Department of Electrical Engineering, National Ilan University, Ilan, Taiwan. His research interests include discrete-event systems, Petri nets, computer-integrated manufacturing, automation, reactive systems, air traffic control, intelligent transport systems, and motor control systems.

Dr. Huang serves as a Reviewer for *Automatica*, IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: SYSTEMS, IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING, IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, *Institution of Engineering and Technology* (IET) *Control Theory and Applications*, *IET Intelligent Transport Systems, International Journal of Production Research*, *The Computer Journal*, *International Journal of Advancement Manufacturing Technology*, *Asian Journal of Control*, *Journal of the Chinese Institute of Engineers*, and *Journal of Information Systems Education*.

Yi-Shun Weng received the B.S. degree in electrical engineering from National Taipei University of Technology, Taipei, Taiwan; the M.S. degree from National Taiwan Ocean University, Keelung, Taiwan, in 2000; and the Ph.D. degree in electrical and electronic engineering from National Defense University, Daxi, Taiwan, in 2011.

He is an Assistant Professor with the Department of Electronic Engineering, Army Academy Republic of China, Zhongli, Taiwan. His research interests include Petri nets, discrete-event systems, and intelligent transportation systems.

MengChu Zhou (S'88–M'90–SM'93–F'03) received the B.S. degree in control engineering from Nanjing University of Science and Technology, Nanjing, China; the M.S. degree in automatic control from Beijing Institute of Technology, Beijing, China; and the Ph.D. degree in computer and systems engineering from Rensselaer Polytechnic Institute, Troy, NY, USA.

Since 1990, he has been with New Jersey Institute of Technology, Newark, NJ, USA, where he is a Distinguished Professor of electrical and computer engineering. He is currently a Professor with Tongji University, Shanghai, China. His research interests include Petri nets, automation, and sensor networks. He is the author or coauthor of over 500 publications, including 11 books and more than 240 journal papers (the majority in IEEE TRANSACTIONS).

Dr. Zhou is a Fellow of the International Federation of Automatic Control and the American Association for the Advancement of Science.