

Modelling and analysis of traffic light control systems

Y.-S. Huang¹ P.-J. Su²

¹Department of Electrical and Electronic Engineering, National Defense University, Taoyuan 335, Taiwan, Republic of China

²School of Defense Science, National Defense University, Taiwan, Republic of China

E-mail: yshuang@ndu.edu.tw

Abstract: An urban traffic network of signalised intersections can be suitably modelled as a discrete event system, in which the traffic light alternations are described by means of timed coloured Petri nets (TCPNs). A module of basic traffic TCPN model with a signal timing plan for a day is successfully constructed. The traffic operations are ruled by the control logic of TCPN and an analysis method using place invariant is verified. From the basic TCPN model, some of the complicated traffic signal models will be easily obtained. Moreover, a real-world supervisor of the urban traffic light system is implemented by the new methodology. Finally, an urban traffic light control system with five intersections is realised. Also, the performance of the supervisor of the urban traffic light system can be confirmed by the simulation results.

1 Introduction

With the growing number of vehicles, traffic congestion and transportation delay on urban arterials are increasing worldwide. Therefore it is practically important to develop, verify and validate simple yet powerful models that help in designing and improving the safety and efficiency of transportation. It is a significant issue to control traffic lights in road-vehicle systems. The main reason is that traffic signals are used to manage conflicting requirements for the use of road space – often at road junctions – by allocating the right side of a way to different sets of mutually compatible traffic movements during distinct time intervals.

The traffic light control systems regulate, warn and guide transportation for the purpose of improving the safety and efficiency of pedestrians and vehicles. There are many literatures to develop various strategies [1–10] and they are classified into two categories [11]: (1) fixed-time strategies and (2) traffic-response strategies. Nowadays, most of the industrialised countries are using fixed-time strategies for urban traffic control. In addition, the topic of traffic signal control can be separated into two categories [12]: (1) determining which signal-indication sequence optimises the system performance and (2) ascertaining how to implement the signal control logic. This paper centralises on the

second category with a traffic signal timing plan that is predetermined.

Petri nets (PNs) have been proven to be a powerful modelling tool for various kinds of discrete event systems [13, 14], and its formalism provides a clear means for presenting simulation and control logic. Hence, the PN is applied in traffic control. In fact, traffic control has been accomplished by using PNs [15–18]. Wang *et al.* [19] chose deterministic and stochastic Petri nets (DSPNs) as the modelling tools. The behaviour of the pre-timed two-phase signal is depicted. The green periods and the cycle lengths are pre-determined and have a fixed duration. The approach via programmable logic control (PLC) and PN synthesis is proposed in [20]. Its traffic lights contain three-coloured lights, and each lane has a series light whose signal will be changed by the regular time intervals. And the model of traffic light is used by PNs. In [21], a signal timing plan is proposed by timed Petri nets (TPNs). The streams allow us to proceed with the eight phases of the signal timing plan. Recently, an urban traffic light controller using statecharts is proposed [22], which includes eight, six and two phases.

As mentioned earlier, the models of traffic light systems have only one set of phase duration. The authors believe that the timing plan of traffic light plays an important role

in urban traffic light systems. And it can determine the optimal splits and the optimal cycle time. It hints that there are many sets of phase duration varied with traffic flow in a day. As a result, the authors propose a new modelling methodology to tackle this difficult problem. To sum up, a traffic light control system with multi-set of phase duration has been designed based on TCPN.

This paper is organised as follows. Section 2 briefly introduces the basic definitions of TCPN that are related to this paper. The traffic intersection control systems are presented in Section 3. Section 4 depicts how to model the traffic lights by using TCPN. Section 5 presents the analysis of the TCPN models. Finally, some conclusions are given in Section 6.

2 The basic definitions of TCPN

We first introduce the new methodology, which is based on a global clock. The global clock values represent the system model time where they may either be discrete or continuous. More precisely, each token has a time stamp on it. The time stamp describes the earliest model time at which the token can be moved by a binding element. Note here that one can define the time stamp unit as seconds, microseconds or millennia and so on. It depends completely on the designer. Please note that in this paper, the authors assume that one time unit is equal to one second and the definitions of TCPN are going to be presented here in a compact way and follow the original definitions of TCPN by Jensen [23, 24].

Definition 1: A timed non-hierarchical CP-net is a tuple $TCPN = (CPN, R, r_0)$ such that:

1. $CPN = (\Sigma, P, T, A, N, C, G, E, I)$ satisfying the requirements below:

- Σ is a finite set of non-empty types, called colour sets.
- P is a finite set of places.
- T is a finite set of transitions.
- A is a finite set of arcs such that $P \cap T = P \cap A = T \cap A = \emptyset$.
- N is a node function. It is defined from A into $P \times T \cup T \times P$.
- C is a colour function. It is defined from P into Σ .
- G is a guard function. It is defined from T into expressions.
- E is an arc expression function. It is defined from A into expressions.

- I is an initialisation function. It is defined from P into closed expressions that is an expression without variables.

2. R is a set of time values, also called *time stamps*. It is closed under $+$ and containing 0.

3. r_0 is an element of R called the *start time*.

Definition 2: A binding of transition t is a function b defined on $Var(t)$, which is called the set of variables of t . $B(t)$ denotes the set of all bindings for t .

Definition 3: A binding element is a pair (t, b) where $t \in T$ and $b \in B(t)$. The set of all binding elements is denoted by BE .

Definition 4: A step Y is enabled in a marking M iff the following properties are satisfied:

$$\forall p \in P: \sum_{(t,b) \in Y} E(p, t) < b > \leq M(p)$$

The expression evaluation $E(p, t) < b >$ yields the multi-set of token colours, which are removed from p when t occurs with the binding b . For a time concept, the enabled definition is modified as follows:

A step Y is enabled in a state (M_1, r_1) at time r_2 iff the following properties are satisfied:

1. $\forall p \in P: \sum_{(t,b) \in Y} E(p, t) < b >_{r_2} \leq M_1(p)$
2. $r_1 \leq r_2$
3. r_2 is the smallest element of R for which there exists a step satisfying the above two restrictions.

Next, the definition is related with place invariant.

Definition 5: For a non-hierarchical CP-net, a set of place weights with range $A \in \Sigma$ is a set of functions $W = \{W_p\}_{p \in P}$ such that $W_p \in [C(p)_{WS} \rightarrow A_{WS}]_L$ for all $p \in P$.

1. W is a place flow iff

$$\forall (t, b) \in BE: \sum_{p \in P} W_p(E(p, t) < b >) = \sum_{p \in P} W_p(E(p, t) < b >)$$

2. W determines a place invariant iff:

$$\forall M \in [M_0 >: \sum_{p \in P} W_p(M(p)) = \sum_{p \in P} W_p(M_0(p))$$

Note that the invariant property in (2) is a dynamic property, while the flow property in (1) is a static property. It hints that the static property can be checked without considering the set of all reachable markings.

3 The description of the basic traffic system

It is observed quite often that the vehicle goes straight and turns right in a junction with two-way roads. In this paper, a traffic system with two phases (shown in Fig. 1), called a basic traffic system, will be introduced. The two phases consist of phase_NS and phase_EW. And the operations are described as follows. In phase_NS, the northbound and southbound vehicles are allowed to go straight and turn right. In phase_EW, the westbound and eastbound vehicles are allowed to go straight and turn right.

A schedule of the traffic signals is given in Table 1 for this intersection. At first, the definition of 'period cycle' that represents one cycle of a period and 'day cycle' that represents one cycle of a day are introduced. There are several periods in a day cycle and they represent the intervals that have continuous execution times. The duration of the phase_NS represents the time when the vehicles can go and pass the intersection in NS direction. Similarly, the duration of the phase_EW represents the time when the vehicle can pass through the intersection in EW direction. The duration of the two phases include both the duration of green and the duration of yellow light. A cycle time is defined as the summation of the two durations. It is worth noticing that there are many repetition cycles in a period. For example, the number of repetition cycles are 30 in 1 h if its cycle time is 120 s.

4 A TCPN model

In this section, the paper focuses on how to model the traffic lights using TCPN. It is an interesting work for the traffic light system model. It is worthy to notice that three TCPN models are constructed in Fig. 2 by three type lines. One is the basic traffic system model, which is modelled by normal lines. The other one is an extended model, which is added with dashed lines. Another one is also an extended model, which is added with heavy lines. The two types of extended models will be explained later. The basic traffic system model consists of two main parts: the left part, which is called the NS traffic lights, describes the states of both the northern and southern traffic lights; the right part, which is called the EW traffic lights, depicts the states of both the eastern and western traffic lights. More precisely, each part can be divided into two sub-models. One sub-model is for the signal indication and the other is for the signals' timing whose model is shared between the signal

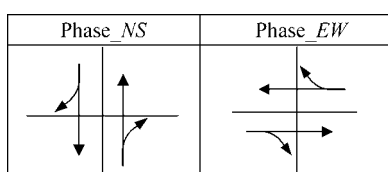


Figure 1 Two-phase transitions

Table 1 A schedule of traffic light signals

Period	t_1	t_2	...	t_i
Execution time	$0-h_1$	h_1-h_2	...	$h_{i(-)1}-h_i$
Phase_NS duration	τ_{11}	τ_{12}	...	τ_{1i}
Phase_EW duration	τ_{21}	τ_{22}	...	τ_{2i}
Cycle time	ct_1	ct_2	...	ct_j

indication models. The duration of the yellow is assumed as 3 s and the lost time [11] is neglected in this paper. Detailed information is given in Sections 4.1 and 4.2.

4.1 Signal indication models

Usually, green light means the vehicles can go straight and turn right. And the vehicles do not permit turning left. In the pioneer works [17–20], this situation is hardly presented in detail. However, this case is easily modelled by the new methodology. A signal indication model can show the traffic light in red, yellow or green. The traffic lights are modelled as places EW_G , EW_Y , EW_R , NS_G , NS_Y and NS_R (i.e. Fig. 2). The places of the traffic lights are triggered by tokens in the TCPN model. Based on the rules of TCPN, the tokens are defined as 4-tuple (d, n, t, ct) with time stamp. The first element d represents the

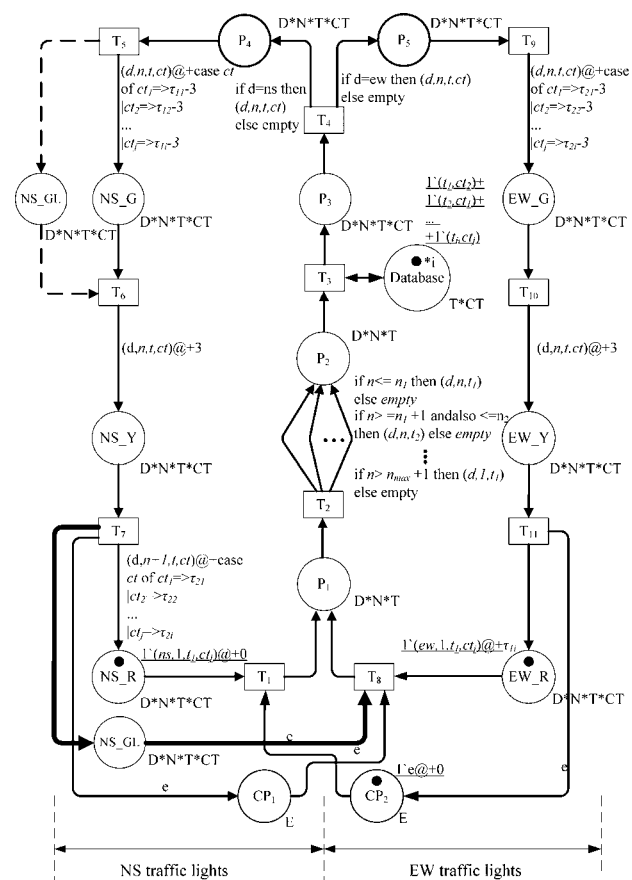


Figure 2 An intersection TCPN model

traffic lights in *NS/EW*. Thus, it can be written as $d \in \{ns, ew\}$. The second one n (i.e. $n \in I^+$) represents the number of the repetition cycles. The element t stands for the period time, where $t \in I^+$. The final element ct represents the cycle time, where $ct \in I^+$. Here, I^+ is a non-negative integer.

Places NS_G , NS_Y , NS_R , EW_G , EW_Y , EW_R , CP_1 and CP_2 belong to this sub-model. A token in place NS_G means the green lights turn on in *NS* traffic lights. And then place NS_Y (NS_R) with a token represents the yellow (red) lights turn on in *NS* traffic lights. Similarly, the places EW_G , EW_Y and EW_R with tokens means the green lights, yellow lights and red lights turn on in *EW* traffic lights, respectively. Places CP_1 and CP_2 play the guard roles, which guarantee that only one direction is allowed to proceed for the vehicle streams. For example, when the green light turns on in *NS*, the red light turns on in *EW*. More specifically, the two places ensure the safety of the traffic. The transitions of the system model will be introduced next.

The transitions T_5 and T_9 stand for the green durations. Because the duration of the phase involves green duration and yellow duration (i.e. 3 s), the time inscription $\tau - 3$ that deducts yellow duration from phase duration is a green duration. The transitions T_6 and T_{10} represent the yellow duration (i.e. 3 s). The transition T_7 stands for the red durations in *NS*. And then, the red duration in *EW* is controlled by the token of place CP_1 . The physical means the red light of the traffic lights in *EW* is going on till the enabled binding element (T_8, e) is taken. Obviously, the red duration of traffic lights in *EW* is τ_{1i} seconds. It reveals that the vehicle streams only proceed in either *NS* or *EW*.

Based on the basic sub-model, two types of extended signal indication models that involve a left turn arrow on green are obtained. The first one is added with the dashed lines in Fig. 2, which describes the left turn arrow on green to be triggered with the right turn arrow on green and go straight arrow on green at the same time. In other words, the three green lights are turned on concurrently. In this extended signal indication model, the authors add a place NS_GL to the basic traffic model. And it represents the left turn green light. The second one is added with heavy lines in Fig. 2, which indicates only one of the three green lights (left turn green light, go straight green light and right turn green light) goes on. It means that the left turn green light (NS_GL) is going on after the other two green lights (NS_G) turn off. For this purpose, the authors add a place NS_GL to the basic model. It also represents the left turn green light. It is worthy to notice that the two types of extended signal indication models in Fig. 2 describe only one part of *NS* traffic lights. The attached places and transitions can also be constructed on the part of *EW* traffic lights. The extended signal indication model is useful in the multi-phase transitions.

4.2 The signal timing plan model

The sub-model of the signal timing plan model is the kernel of the TCPN model (i.e. Fig. 2). It is the model's duty to assign the cycle times to the different periods. Especially, there are no durations and delay times in the sub-model. Indeed, it is used to estimate the cycle time of a period. In the sub-model, one has to count the number (i.e. n_i) of repetition cycles. Once the number of repetition cycles meets the number n_i , the current period t_i will enter the next period t_{i+1} . And then, a new cycle time will be given in terms of period t_{i+1} .

The definitions of the elements of tokens in the sub-model are the same as the signal indication model. It is worthy to notice that the component elements of token in this sub-model consist of two (i.e. (t, ct)), three (i.e. (d, n, t)), or four (i.e. (d, n, t, ct)) elements. And the signal indication model consists of four elements (i.e. (d, n, t, ct)). A more explicit expression is described as follows.

There are several arc expressions between transition T_2 and place P_2 . The arc expressions determine when the current period goes to the next period. The repetition cycle time in a 'period cycle' is represented by r_i . And values n_i are determined from the formula, where $n_i = \sum_{i=1}^n r_i$, $n \in I^+$. For example, $n_2 = r_1 + r_2$. Once the number of repetition cycles is equal to n_i , the current period (t_i) will go to the next period (t_{i+1}). And then, the current token (d, n, t_i) will change to (d, n, t_{i+1}) . In addition, a maximum number of repetition cycles $n_{\max}$ for the 'day cycle' is required, because the system model needs a mechanic to judge what is the day off. On the new day, the number $n_{\max}$ is enforced to reset. It means the state is coming back to the initial state.

In summary, when a token is removed from P_1 to P_2 , the token is with a t_i . Once the token is removed from P_2 to P_3 , the token is now with a ct_i . Note that the ct_i is offered by the Place Database. The function of Place Database is that the model can show both the cycle time and the current period together. Therefore, we have to create a database of (t_i, ct_i) by tokens in Database. Meanwhile, the output token is changed and it is with four elements like as (d, n, t_i, ct_i) .

5 Analysis of the urban traffic net model

A traffic light control system model must have correct and readable features. For example, the controller should not lock up (deadlock) due to some unexpected combination of actions, should not allow conflicting movements to have right side of a way simultaneously, should be able to serve all signal phases and return to some initial state. A major strength of TCPN is the availability of methods for analysing the properties of the model. Those properties of TCPN model reveal whether the model is reliable or not.

There are three methods to analyse a TCPN model: (1) invariant method, (2) occurrence graphs method and (3) simulation method [23]. In this section, the basic traffic light control system model is analysed by the occurrence graphs method and invariant method. In addition, a real-world urban traffic net, which consists of three intersections, is used to verify the model by simulation.

5.1 Invariant method

The basic idea behind place invariants creates equations that are satisfied in all reachable markings [24]. In TCPN nets, the sets of removed tokens are not fully determined by the binding elements. Based on the rules of TCPN, a transition can be fired if the global time is greater than or equal to the time stamp. It hints that the system acts according to the time stamps of the binding elements. In fact, the system models require only the time stamps to be small enough, instead of requiring them to have some exact time values. This means that the linearity of weight functions is insufficient to guarantee that each flow determines an invariant. However, our traffic light TCPN model is pre-determining time. Therefore, it is necessary to use invariants in the analysis of TCPN models. The basic model (Fig. 2) is used to analyse the place invariance. Based on Definition 5, we can obtain several equations from Fig. 2. And the performance of the system should be verified by the equations. The detailed information is given as follows.

$$NS_G + NS_Y + NS_R + P_1 + P_2 + P_3 + P_4 = 1 \quad (1)$$

This invariant states that there can be only one token in any one of the places involved in (1). And it indicates that the firing sequence of the binding elements should be in order at *NS* traffic light. For example, the variations of traffic lights are red, green and yellow in turn. Note that places P_1 , P_2 , P_3 , and P_4 are necessary control places. Similarly, the other invariant can be obtained, i.e. $EW_G + EW_Y + EW_R + P_1 + P_2 + P_3 + P_5 = 1$. This invariant asserts that at any given time, there is only one token on the right-hand side of the display indication modules.

$$NS_G + NS_Y + NS_R + P_1 + P_2 + P_3 + P_4 = EW_R, \quad \text{where } EW_R = 1 \quad (2)$$

This invariant depicts that if there is one token in place EW_R , then there must be a token in either place NS_G , NS_Y , NS_R , P_1 , P_2 , P_3 or P_4 . It means that once a red signal goes on in the *EW* traffic lights, a green, a yellow or a red signal turns on in the *NS* traffic lights.

In the other way, the invariant $EW_G + EW_Y + EW_R + P_1 + P_2 + P_3 + P_5 = NS_R$ asserts that if there is one token in place NS_R , then there must be a token in either place EW_G , EW_Y , EW_R , P_1 , P_2 , P_3 or P_5 . It shows that once a red signal is going on in the *EW* traffic

lights, a green, a yellow or a red turns on in the *NS* traffic lights.

$$NS_G + NS_Y + EW_G + EW_Y + P_1 + P_2 + P_3 + P_4 + P_5 + CP_1 + CP_2 = 1 \quad (3)$$

As mentioned earlier, this invariant means that the direction of the vehicles movement is either in *NS* or *EW*. This invariant hints that the system model guarantees the safety of the traffic.

The three equations (1), (2) and (3) show the key invariants in the TCPN model. And the three key invariants show that the system model is with invariants.

5.2 Occurrence graphs method

The basic idea behind occurrence graphs (OGs) is to construct a graph containing a node for each reachable marking and an arc for each occurring binding element. Also, it is intuitive in this approach to see that there is no possibility of deadlock in the system model.

The OG of the basic TCPN model is shown in Fig. 3. It is worthy to notice that each node represents a marking, and the content of the marking is described by the text inscription of the node. And each arc represents the occurrence of a binding element, and the content of this binding element is described by the text attached to the arc. For example, EW_R and NS_G are used to describe the node N_5 . This means that the red lights turn on in the *EW* traffic light and the green lights turn on in the *NS* traffic lights. After a binding element T_6 firing, the marking will be changed to node N_6 . It is worthy to notice that the initial marking (i.e. node $N_{INITIAL}$) shows all the red lights turn on in a set of traffic lights. For the sake of simplicity, the place Database is not shown in the OG.

Based on this analysis, it can be concluded that: (1) there are no dead-end nodes in the OG, therefore the net is live; (2) the net is reversible because we can always find an

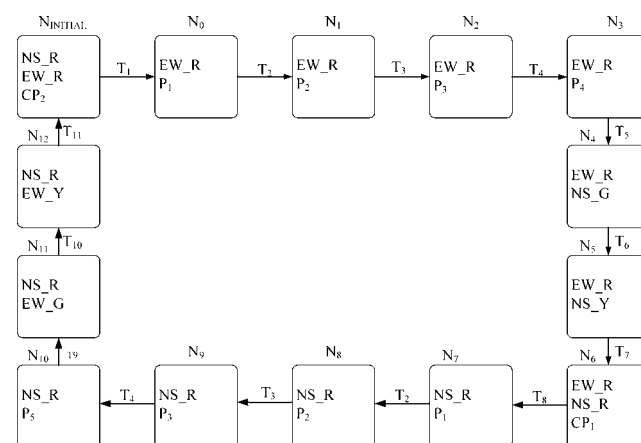


Figure 3 The OG of the basic TCPN model

occurring sequence that brings the system back to the initial marking.

5.3 The simulation results

Considering the simulation, a real-world urban traffic net is used to verify the model. The real traffic net consists of five intersections, which are shown in Fig. 4. Some notations are employed to represent the five intersections. For example, $1NS_G$ means the green light goes on in the NS traffic light, which is placed at intersection I_1 . Fig. 5 shows the information of the phase transitions in this real case. Also Table 2 shows the signal time plan of traffic light model. The urban traffic net model will be constructed based on the basic traffic model. For convenience, the five intersections are divided into three types. One consists of intersection I_1 , I_4 and I_5 whose models are the same as in the basic model. The other one is with intersection I_2 whose model is obtained from the first extending model. Another one is with intersection I_3 whose model is also from the second extending model. The detailed information of the urban net model is described as follows.

1. Modelling of intersection I_1 , I_4 and I_5 (type I)

The model of type I is the same as that of the basic traffic systems model. Hence, the module of basic traffic system could be applied directly to type I. From the schedule of the signal timing plan, there are four kinds of cycle times given. For fitting the basic models, the notations $ct_1 = 90$, $ct_2 = 120$, $ct_3 = 180$ and $ct_4 = 200$ are assigned. Then, the values of r_i and n_i have to be determined. The intersection I_1 is used to illustrate how to obtain the values r_i and n_i . Table 2 shows the execution time and the cycle time for every period of the intersection I_1 . From the table, the number of repetitions (i.e. r_i) is easy to count. For example, the numbers of repetitions of r_1 is 30 for the period t_1 in the intersection I_1 . As a result, all the numbers of repetitions (i.e. r_i) are obtained and are shown in Table 3. Based on Table 3, data the values n_i shown in Table 4 are determined from the formula, where $n_i = \sum_{j=1}^n r_j$, $n \leq 9$. For example, $n_2 = r_1 + r_2 = 30 + 160 = 190$. It is worthy to notice that the values of r_i and n_i in Table 3 can be applied to intersections I_2 and I_3 since the three intersections have the same cycle time in a period.

Finally, three TCPN models of type I will be derived if the values n_i , τ_{1i} , τ_{2i} and ct_j are put in the original model (i.e.



Figure 4 A real-world urban traffic net

	Phase1	Phase2	Phase3	Phase4
1				
2				
3				
4				
5				

Figure 5 The phase transitions of the traffic net

Fig. 2). And then, the three models (i.e. I_1 , I_4 and I_5) are given in Figs. 6–8.

Based on the place-invariant method, the three models should be verified by the following equations:

Intersection I_1 :

$$1NS_G + 1NS_Y + 1NS_R + 1P_1 + 1P_2 + 1P_3 + 1P_4 = 1 \quad (4)$$

$$1EW_G + 1EW_Y + 1EW_R + 1P_1 + 1P_2 + 1P_3 + 1P_4 = 1 \quad (5)$$

$$1NS_G + 1NS_Y + 1NS_R + 1P_1 + 1P_2 + 1P_3 + 1P_4 = 1EW_R, \text{ where } 1EW_R = 1 \quad (6)$$

$$1EW_G + 1EW_Y + 1EW_R + 1P_1 + 1P_2 + 1P_3 + 1P_4 = 1NS_R, \text{ where } 1NS_R = 1 \quad (7)$$

$$1NS_G + 1NS_Y + 1EW_G + 1EW_Y + 1P_1 + 1P_2 + 1P_3 + 1P_4 + 1P_5 + 1CP_1 + 1CP_2 = 1 \quad (8)$$

Intersection I_4 :

$$4NS_G + 4NS_Y + 4NS_R + 4P_1 + 4P_2 + 4P_3 + 4P_4 = 1 \quad (9)$$

$$4EW_G + 4EW_Y + 4EW_R + 4P_1 + 4P_2 + 4P_3 + 4P_4 = 1 \quad (10)$$

$$4NS_G + 4NS_Y + 4NS_R + 4P_1 + 4P_2 + 4P_3 + 4P_4 = 4EW_R, \text{ where } 4EW_R = 1 \quad (11)$$

$$4EW_G + 4EW_Y + 4EW_R + 4P_1 + 4P_2 + 4P_3 + 4P_4 = 4NS_R, \text{ where } 4NS_R = 1 \quad (12)$$

$$4NS_G + 4NS_Y + 4EW_G + 4EW_Y + 4P_1 + 4P_2 + 4P_3 + 4P_4 + 4P_5 + 4CP_1 + 4CP_2 = 1 \quad (13)$$

Table 2 The schedule of the signal timing plan

	Period	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9
	Execution time	00:00–01:00	01:00–05:00	05:00–07:00	07:00–09:00	09:00–13:00	13:00–16:30	16:30–19:00	19:00–23:00	23:00–24:00
I_1	Phase 1	60	46	60	110	110	110	110	95	60
	Phase 2	60	44	60	90	90	90	90	85	60
	cycle time	120	90	120	200	200	200	200	180	120
I_2	Phase 1	80	60	80	140	140	140	140	120	80
	Phase 2	40	30	40	60	60	60	60	60	40
	cycle time	120	90	120	200	200	200	200	180	120
I_3	Phase 1	55	30	55	85	85	85	85	80	55
	Phase 2	10	10	10	15	15	15	15	15	10
	Phase 3	45	40	45	85	85	85	85	70	45
	Phase 4	10	10	10	15	15	15	15	15	10
	cycle time	120	90	120	200	200	200	200	180	120
I_4	Phase 1	60	45	60	100	100	100	100	90	60
	Phase 2	60	45	60	100	100	100	100	90	60
	cycle time	120	90	120	200	200	200	200	180	120
I_5	Phase 1	90	60	90	160	160	160	160	140	90
	Phase 2	30	30	30	40	40	40	40	40	30
	cycle time	120	90	120	200	200	200	200	180	120

Table 3 The values of r_i

	r_1	r_2	r_3	r_4	r_5	r_6	r_7	r_8	r_9
r_i	30	160	60	36	72	63	45	80	30

Table 4 The values of n_i

	n_1	n_2	n_3	n_4	n_5	n_6	n_7	n_8	n_9
n_i	30	190	250	286	358	421	466	546	576

Intersection I_5 :

$$5NS.G + 5NS.Y + 5NS.R + 5P_1 + 5P_2 + 5P_3 + 5P_4 = 1 \quad (14)$$

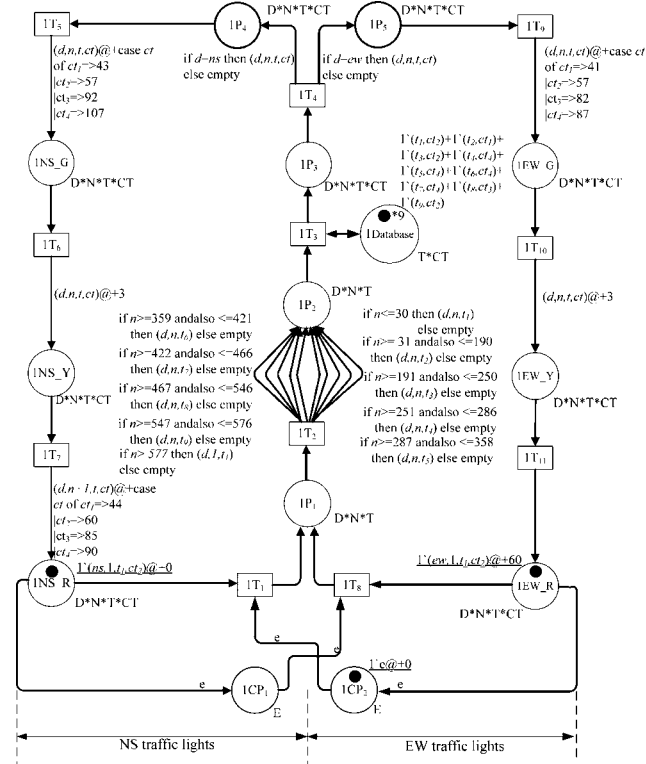
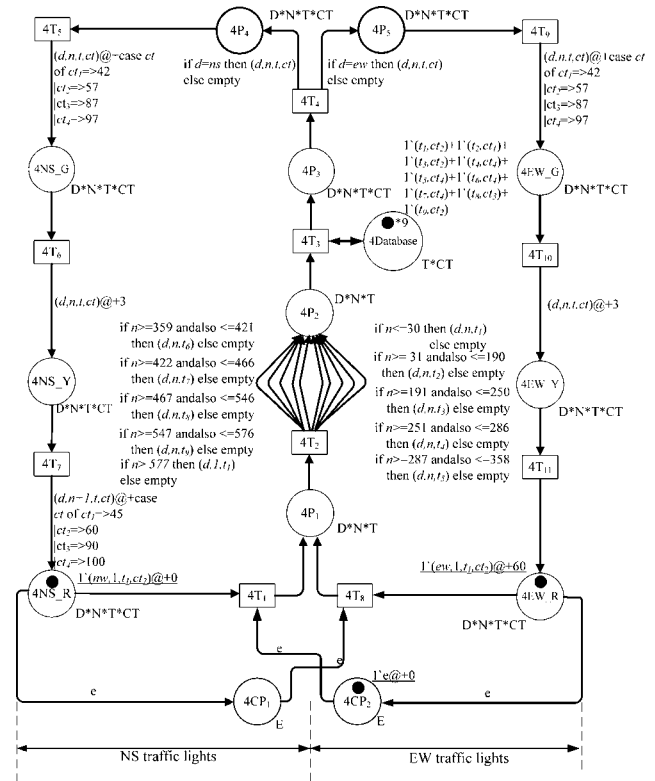
$$5EW.G + 5EW.Y + 5EW.R + 5P_1 + 5P_2 + 5P_3 + 5P_4 = 1 \quad (15)$$

$$5NS.G + 5NS.Y + 5NS.R + 5P_1 + 5P_2 + 5P_3 + 5P_4 = 5EW.R, \text{ where } 5EW.R = 1 \quad (16)$$

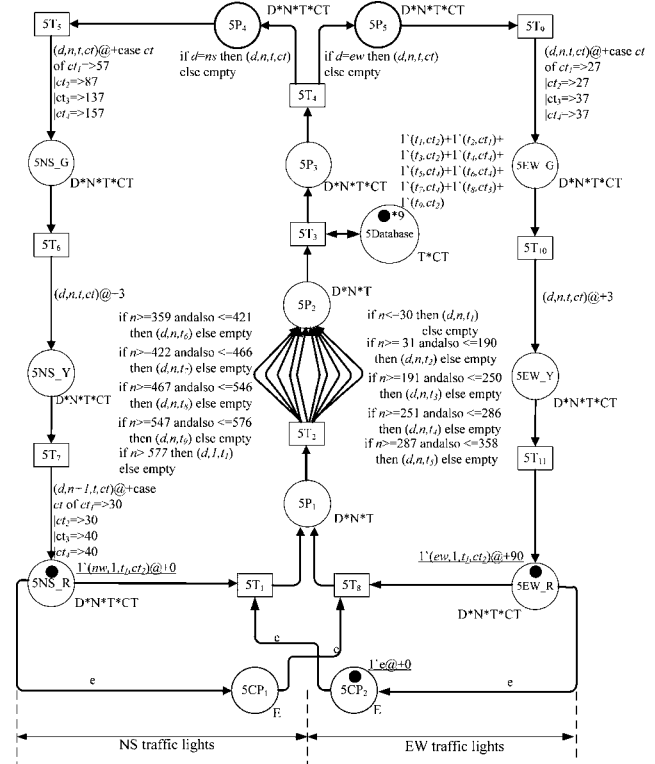
$$5EW.G + 5EW.Y + 5EW.R + 5P_1 + 5P_2 + 5P_3 + 5P_4 = 5NS.R, \text{ where } 5NS.R = 1 \quad (17)$$

$$5NS.G + 5NS.Y + 5EW.G + 5EW.Y + 5P_1 + 5P_2 + 5P_3 + 5P_4 + 5P_5 + 5CP_1 + 5CP_2 = 1 \quad (18)$$

2. Modelling of intersection I_2 (type II)


Figure 6 TCPN model of the intersection I_1

Figure 7 TCPN model of the intersection I_4

A T-type intersection is usually presented in a real-world urban traffic system. A feature of the T-type intersection is that one way of the vehicle stream cannot be allowed to go


Figure 8 TCPN model of the intersection I_5

straight. For this reason, a left turn arrow on green is needed to present in the traffic light system. As a result, the function of the left turn green light can be presented when a place NS_GL is added on the original model. It is interesting that the completed TCPN model of the T-type intersection (i.e. Fig. 9) can be derived after a minor revision of the first extended traffic system model.

From the definition of the phases (i.e. Fig. 5), a right turn movement is absent in northern traffic light (i.e. phase 1 of I_2). For convenience, i.e. in the extended model (i.e. Fig. 9), the place NS_G used to represent the vehicle stream can be allowed to go straight in the northbound traffic light and the other way can be allowed to go straight and turn right. In addition, the vehicle stream cannot be allowed to go straight in the eastbound (i.e. phase 2 of I_2). This situation is different from the phase of the basic model (i.e. phase 2 of I_1). For the same reason, the place E_G used to represent the vehicle stream can be allowed to turn right and turn left in the eastern traffic light. Notice that the place EW_G/EW_GL is used instead of the place E_G/E_GL in the new extended TCPN model.

Finally, the TCPN model of T-type will be derived if the values n_i , τ_{1i} , τ_{2i} and ct_j are put in the original model. And then, the model is constructed and is shown in Fig. 9.

Based on the place-invariant method, the model of intersection I_2 should be verified by the following equations:

$$2NS_G + 2NS_Y + 2NS_R + 2P_1 + 2P_2 + 2P_3 + 2P_4 = 1 \quad (19)$$

$$2W_G(2W_GL) + 2W_Y + 2W_R + 2P_1 + 2P_2 + 2P_3 + 2P_4 = 1 \quad (20)$$

$$2NS_G + 2NS_Y + 2NS_R + 2P_1 + 2P_2 + 2P_3 + 2P_4 = 2W_R, \text{ where } 2W_R = 1 \quad (21)$$

$$2W_G(2W_GL) + 2W_Y + 2W_R + 2P_1 + 2P_2 + 2P_3 + 2P_4 = 2NS_R, \text{ where } 2NS_R = 1 \quad (22)$$

$$2NS_G + 2NS_Y + 2W_G(2W_GL) + 2W_Y + 2P_1 + 2P_2 + 2P_3 + 2P_4 + 2P_5 + 2CP_1 + 2CP_2 = 1 \quad (23)$$

3. Modelling of intersection I_3 (type III)

From Fig. 5, the phase of intersection I_3 is with four phases. Obviously, one of the phases is with a left turn movement of intersection I_3 . The TCPN model of the intersection I_3 can be easily obtained because the phase is already included in the second extended model. As a result, the TCPN model of the intersection I_3 can be constructed if the relation values are put in the original model. The model is constructed and is shown in Fig. 10.

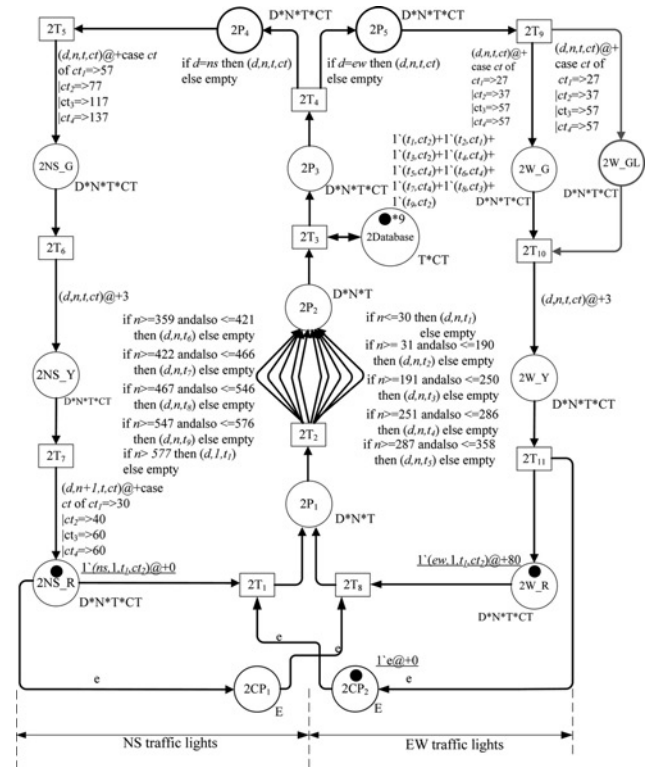


Figure 9 TCPN model of the intersection I_2

Based on the place-invariant method, the model of intersection I_3 should be verified by the following equations:

$$3NS_G + 3NS_Y + 3NS_R + 3P_1 + 3P_2 + 3P_3 + 3P_4 = 1 \quad (24)$$

$$3EW_G + 3EW_Y + 3EW_R + 3P_1 + 3P_2 + 3P_3 + 3P_4 = 1 \quad (25)$$

$$3NS_G + 3NS_Y + 3NS_R + 3P_1 + 3P_2 + 3P_3 + 3P_4 = 3EW_R, \text{ where } 3EW_R = 1 \quad (26)$$

$$3EW_G + 3EW_Y + 3EW_R + 3P_1 + 3P_2 + 3P_3 + 3P_4 = 3NS_R, \text{ where } 3NS_R = 1 \quad (27)$$

$$3NS_G + 3NS_Y + 3NS_GL + 3EW_G + 3EW_Y + 3EW_GL + 3P_1 + 3P_2 + 3P_3 + 3P_4 + 3P_5 + 3CP_1 + 3CP_2 = 1 \quad (28)$$

The TCPN models are implemented and simulated by the CPN tools [25]. The aim of the simulation is to observe the relationship between the cycle time and the execution time in each period. Based on the schedule of the signal timing plan (i.e. Table 2), the global time of the supervisor is set as 86 400 time units (i.e. one time unit is equal to 1 s) for the nine periods (i.e. $t_1, t_2, \dots, t_9$) in the CPN tools. In summary, the simulation results are the same as the pre-determining time (i.e. Table 2) and the cycle time of each period is consistent with Table 2. Moreover, the traffic performance can be confirmed by the simulation results.

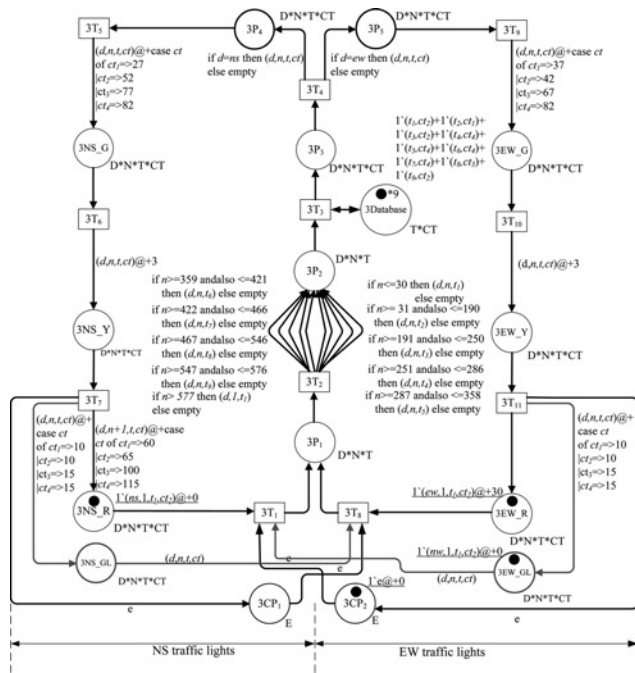


Figure 10 TCPN model of the intersection I_3

6 Conclusion

This paper presents the modelling, analysis and implementation of an urban traffic lights system using TCPN models. This paper also particularly proposed the module of basic traffic light system model, which can assist in designing the extended models. Based on the operational flow of the traffic lights systems, the authors derive the associated TCPN model by looking into the schedule of the signal timing plan of the traffic systems. The advantage of the proposed approach is the clear presentation of the system behaviour and readiness for implementation. To summarise, this paper has the following contributions:

- How to use TCPN to model the traffic lights of the urban network. Also, the applications of TCPN to urban traffic lights have been realised.
- Structural analysis of TCPN models.
- The traffic systems with signal timing plan for a day are successful in converting TCPN models. These examples are helpful to us to obtain a TCPN model for a complex urban traffic lights system.

The authors presume that our research using TCPN to model traffic light systems will become more important in this field because of the increasing demands in many features of the traffic light systems.

7 Acknowledgment

This work was supported by the National Science Council of Taiwan, ROC., under Grant NSC 96-2221-E-606-029.

8 References

- ALLSOP R.E.: 'SIGSET: a computer program for calculating traffic capacity of signal-controlled road junctions', *Traffic Eng. Control*, 1971, **12**, pp. 58–60
- ALLSOP R.E.: 'SIGCAP: a computer program for assessing the traffic capacity of signal-controlled road junctions', *Traffic Eng. Control*, 1976, **17**, pp. 338–341
- LITTLE J.D.C.: 'The synchronization of traffic signals by mixed integer-linear-programming', *Oper. Res.*, 1966, **14**, pp. 568–594
- LITTLE J.D.C., KELSON M.D., GARTNER N.H.: 'MAXBAND: a program for setting signals on arterials and triangular networks', *Transp. Res. Record*, 1981, **795**, pp. 40–46
- LI M.T., GAN A.C.: 'Signal timing optimization for oversaturated networks using TRANSYT-7F', *Transp. Res. Record*, 1999, **1683**, pp. 118–126
- ROBERTSON D.I.: 'TRANSYT method for area traffic control', *Traffic Eng. Control*, 1969, **10**, pp. 276–281
- HUNT P.B., ROBERTSON D.I., BRETHERTON R.D.: 'The SCOOT on-line traffic signal optimization technique', *Traffic Eng. Control*, 1982, **23**, pp. 190–192
- GARTNER N.H.: 'OPAC: a demand-responsive strategy for traffic signal control', *Transp. Res. Record*, 1983, **906**, pp. 75–81
- LIM J.H., HWANG S.H., SUH I.H., BIEN Z.: 'Hierarchical optimal control of oversaturated urban traffic networks', *Int. J. Control*, 1981, **33**, pp. 727–737
- DAVISON E.J., OZGUNER U.: 'Decentralized control of traffic networks', *IEEE Trans. Automat. Contr.*, 1983, **28**, pp. 677–688
- PAPAGEORGIOU M., DIAKAKI C., DINOPOULOU V., KOTSIALOS A., WANG Y.: 'Review of road traffic control strategies', *Proc. IEEE*, 2003, pp. 2043–2067
- LIST G.F., CETIN M.: 'Modeling traffic signal control using Petri nets', *IEEE Trans. Intelligent Transport. Syst.*, 2004, **5**, pp. 177–187
- MURATA T.: 'Petri Nets: properties, analysis and applications', *Proc. IEEE*, April 1989, pp. 541–580
- PETERSON J.L.: 'Petri net theory and the modeling of systems' (Prentice Hall, 1981)
- FEBBRARO A.D., GIGLIO D., SACCO N.: 'Urban traffic control structure based on hybrid Petri nets', *IEEE Trans. Intelligent Transport. Syst.*, 2004, **5**, pp. 224–237

- [16] TOLBA C., THOMAS P., ELMOUDNI A., LEFEBVRE D.: 'Performances evaluation of the traffic control in a single crossroad by Petri nets'. Proceedings of IEEE Emerging Technologies and Factory Automation, Lisbon, Portugal, September 2003, pp. 157–160
- [17] TZES A., SEONGHO K., MCSHANE W.R.: 'Application of Petri networks to transportation network modeling', *IEEE Trans. Vehicular Technol.*, 1996, **45**, pp. 391–400
- [18] WANG J., JIN C., DENG Y.: 'Performance analysis of traffic networks based on stochastic timed Petri net models'. Proceedings of IEEE Engineering of Complex Computer Systems, Las Vegas, NE, USA, October 1999, pp. 77–85
- [19] WANG H., LIST G.F., DICESARE F.: 'Modeling and evaluation of traffic signal control using timed Petri nets'. Proceedings of IEEE Systems, Man and Cybernetics, Le Touquet, France, 1993, pp. 180–185
- [20] LIN L., NAN T., XIANGYANG M., FUBING S.: 'Implementation of traffic lights control based on Petri nets'. Proceedings of IEEE Intelligent Transportation Systems, Shanghai, China, 2003, pp. 1087–1090
- [21] DOTOLI M., FANTI M.P., LACOBELLIS G.: 'Validation of an urban traffic network model using colored timed Petri nets'. IEEE Int. Conf. on Systems, Man and Cybernetics, Waikoloa, HI, USA, 2005, pp. 1347–1352
- [22] HUANG Y.S.: 'Design of traffic light control system using statecharts', *Comp. J.*, 2006, **49**, pp. 634–649
- [23] JENSEN K.: 'Coloured Petri nets vol. 2' (Springer-Verlag, 1997)
- [24] JENSEN K.: 'Coloured Petri nets vol. 2' (Springer-Verlag, 1995)
- [25] <http://wiki.daimi.au.dk/cpntools/cpntools.wiki>, Computer tool for coloured Petri nets