Model Predictive Control of Traffic Flow Based on Hybrid System Modeling

Tatsuya Kato*,YoungWoo Kim**and Shigeru Okuma*

*Dept. of Information Electronics, Nagoya University, Nagoya, Aichi, Japan
(Tel: +81-52-789-2777; e-mail: kato@okuma.nuee.nagoya-u.ac.jp
Tel: +81-52-789-2775; e-mail: okuma@okuma.nuee.nagoya-u.ac.jp)
**, Space Robotics Research Center, Toyota Technological Institute , Nagoya, Aichi, Japan
(Tel: +81-52-809-1818; e-mail: kim@toyota-ti.ac.jp)

Abstract: This paper presents a new framework for the traffic flow control based on integrated model descriptions as the Hybrid Dynamical System (HDS).

Keywords: Traffic Control,Mixed Logical Dynamical Systems ,Hybrid Petri Net

1. Introduction

With the increasing number of automobile and complication of traffic network, the traffic flow control becomes one of the significant economic and social issues in urban life. Many researchers have been involved in related researches in order to alleviate the traffic congestion.

Although this model expresses well the behavior of the flow on the freeway, it is unlikely that this model is also applicable to the urban traffic network which involves many discontinuities of the density coming from the existence of the intersection controlled by the traffic signals.

This paper presents a new method for the real-time traffic signal control based on integrated model descriptions as the Hybrid Dynamical System (HDS). The geometrical information on the traffic network is characterized by using Hybrid Petri Net (HPN) by both graphical and algebraic descriptions. Then, the algebraic behavior of traffic flow is transformed into the Mixed Logical Dynamical Systems (MLDS) form in order to introduce the optimization technique.

2. Modeling of Traffic Flow Control System (TFCS) based on HPN

The Traffic Flow Control System (TFCS) is the collective entity of the traffic network, traffic flow and traffic signals. Although some of them have been fully considered by the previous studies, most of the previous studies did not simultaneously consider all of them. In this section, the HPN model is developed, which provides both graphical and algebraic descriptions for the TFCS.

2.1. Representation of TFCS as HPN

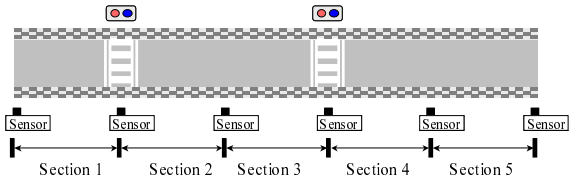

Fig. 1. Straight road

HPN is one of the useful tools to model and visualize the system behavior with both continuous and discrete variables. Figure 1 shows the HPN model for the road of Fig.2. In Fig.1, each section i of l_i-meter long constitutes the straight road, and two traffic lights are installed at the point of crosswalk. HPN has a structure of $N = (P, T, q, I_+, I_-, M^0)$. The set of places P is partitioned into a subset of discrete places P_d and a subset of continuous places P_c. $p_c \in P_c$ represents each section of the road, and has maximum capacity (maximum number of cars). Also, P_d represents the traffic signal where green signal is indicated by a token in the corresponding discrete place $p_d \in P_d$. The marking $M = [\boldsymbol{m_C}|\boldsymbol{m_D}]$ has both continuous (m dimension) and discrete (n dimension) parts where $\boldsymbol{m_C}$ represents the number of vehicles in the correponding continuous places, and $\boldsymbol{m_D}$ denotes the state of the corresponding traffic signal (i.e. takes binary value). Note that each signal is supposed to have only two states 'go (green)' or 'stop (red)' for the simplicity. T is the set of continuous transitions which represent the boundary of two successive sections. The function $q_j(\tau)$ specifies the firing speeds assigned to transition $t_j \in T$ at time τ. $q_j(\tau)$ represents the number of cars passing through the boundary of two successive sections (measuring position) at time τ.

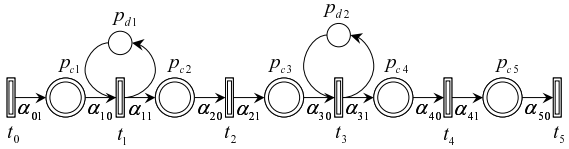

Fig. 2. HPN model of straight road

The functions $I_\pm(p,t)$ are forward and backward incidence relationships between transition t and place p which connects the transition. The element of $I(p,t)$ is always 0 or 1. Finally, M^0 is specified as the initial marking of the place $p \in P$.

The net dynamics of HPN is represented by a simple first order differential equation for each continuous place $p_{c_i} \in P_c$ as follows :

$$\frac{dm_{C,i}(\tau)}{dt} = \sum_{t_j \in {p_{c_i}}^\bullet \cup {p_{c_i}}^\bullet} I(p_{c_i}, t_j) \cdot q_j(\tau) \cdot m_{D,j}(\tau) \qquad (1)$$

where $m_{C,i}(\tau)$ is the marking for the place $p_{c_i} (\in P_c)$ at time τ, $m_{D,j}(\tau)$ is the marking for the place $p_{d_j} (\in P_d)$, and $I(p,t) = I_+(p,t) - I_-(p,t)$. The equation (1) is transformed to its discrete-time version supposing that $q_j(\tau)$ is constant

during two successive sampling instants as follows :

$$m_{C,i}((\kappa+1)T_s) = m_{C,i}(\kappa T_s) + \sum_{t_j \in {p_{c_i}}^\bullet \cup p_{c_i}^\bullet} I(p_{c_i}, t_j) \cdot q_j(\kappa T_s) \cdot m_{D,j}(\kappa T_s) \cdot T_s \quad (2)$$

where κ and T_s are sampling index and period.
Note that the transition t is *enabled* at the sampling instant κT_s if the marking of its preceding discrete place $p_{d_j} \in P_d$ satisfies $m_{D,j}(\kappa) \geq I_+(p_{d_j}, t)$. Also if t does not have any input (discrete) place, t is always *enabled*.

2.2. Definition of flow q_i

In order to derive the flow behavior, the relationship among q_i, k_i and v_i must be specified. One of the simple ideas is to use the well-known model

$$q_i(\tau) = \frac{(k_i(\tau) + k_{i+1}(\tau))}{2} \frac{v_i(k_i(\tau)) + v_{i+1}(k_{i+1}(\tau))}{2} \quad (3)$$

supposing that the density k_* and average velocity v_* of the flow in i and $(i+1)$th sections are almost identical. Then, by incorporating the velocity model

$$v_i(\tau) = \quad v_{f_i} \cdot \left(1 - \frac{k_i(\tau)}{k_{jam}}\right) \quad (4)$$

with (3), the flow dynamics can be uniquely defined. Here, k_{jam} is the density in which the vehicles on the roadway are spaced at minimum intervals (traffic-jammed), and v_{f_i} is the free velocity, that is, the velocity of the vehicle when no other car exists in the same section.
If there exist no abrupt change in the density on the road, this model is expected to work well. However, in the urban traffic network, this is not the case due to the existence of the intersections controlled by the traffic signals. In order to treat the discontinuities of the density among neighboring sections (i.e. neighboring continuous places), the idea of 'shock wave'[4] is introduced as follows. We consider the case as shown in Fig.3 where the traffic density of ith section is higher than that of $(i+1)$th section in which the boundary of density difference designated by the dotted line is moving forward. Here, the movement of this boundary is called shock wave and the moving velocity of the shock wave c_i depends on the densities and average velocities of ith and $(i+1)$th sections as follows [4]:

$$c_i(\tau) \quad = \quad \frac{v_i(\tau)k_i(\tau) - v_{i+1}(\tau)k_{i+1}(\tau)}{k_i(\tau) - k_{i+1}(\tau)} \quad (5)$$

The traffic situation can be categorized into the following four types taking into account the density and shock wave.

(C1) $k_i(\tau) < k_{i+1}(\tau)$, and $c_i(\tau) > 0$,
(C2) $k_i(\tau) < k_{i+1}(\tau)$, and $c_i(\tau) \leq 0$,
(C3) $k_i(\tau) > k_{i+1}(\tau)$,
(C4) $k_i(\tau) = k_{i+1}(\tau)$ (no shock wave).

Firstly, in both cases of (C1) and (C2) where $k_i(\tau)$ is smaller than $k_{i+1}(\tau)$, the vehicles passing through the density boundary (dotted line) reduce their speeds. The movement of the shock wave is illustrated in Fig.3 ($c_i(\tau) > 0$) and Fig.4 ($c_i(\tau) \leq 0$).
In Figs.3 and 4 , the 'measuring position' implies the position where transition t_i is assigned. Since the traffic flow

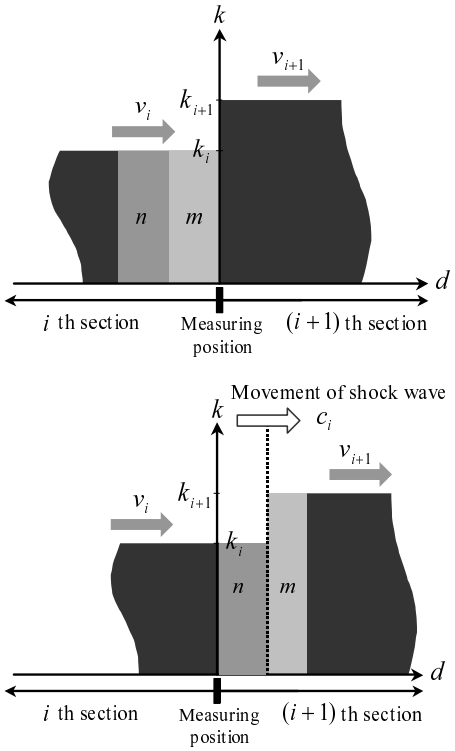

Fig. 3. Movement of shock wave in the case of $k_i(\tau) < k_{i+1}(\tau)$ and $c_i(\tau) > 0$

$q_i(\tau)$ represents the number of vehicle passing through the measuring position per unit time, in the case of (C1), it can be represented by $n+m$ in Fig.3, where n and m represent the area of the corresponding rectangular, i.e. the product of the v_i and k_i. Similarly, in the case of (C2), $q_i(\tau)$ can be represented by m in Fig.4. These considerations lead to the following models:

in the case of (C1)

$$q_i(\tau) \quad = \quad v_i(\tau)k_i(\tau) \quad (6)$$
$$= \quad v_{f_i}\left(1 - \frac{k_i(\tau)}{k_{jam}}\right)k_i(\tau) \quad (7)$$

in the case of (C2)

$$q_i(\tau) \quad = \quad v_{i+1}(\tau)k_{i+1}(\tau) \quad (8)$$
$$= \quad v_{f_{i+1}}\left(1 - \frac{k_{i+1}(\tau)}{k_{jam}}\right)k_{i+1}(\tau) \quad (9)$$

In the cases of (C3) and (C4) where $k_i(\tau)$ is greater than $k_{i+1}(\tau)$, the vehicles passing through the density boundary come to accelerate. In this case, the flow can be well approximated by taking into account the average density of neighboring two sections. This is intuitively because the difference of the traffic density is going down. Then in the cases of (C3) and (C4), the traffic flow can be formulated as follows:

in the cases of (C3) and (C4),

$$q_i(\tau) = \left(\frac{k_i(\tau)+k_{i+1}(\tau)}{2}\right) v_f(\tau) \left(1 - \frac{k_i(\tau)+k_{i+1}(\tau)}{2k_{jam}}\right) \quad (10)$$

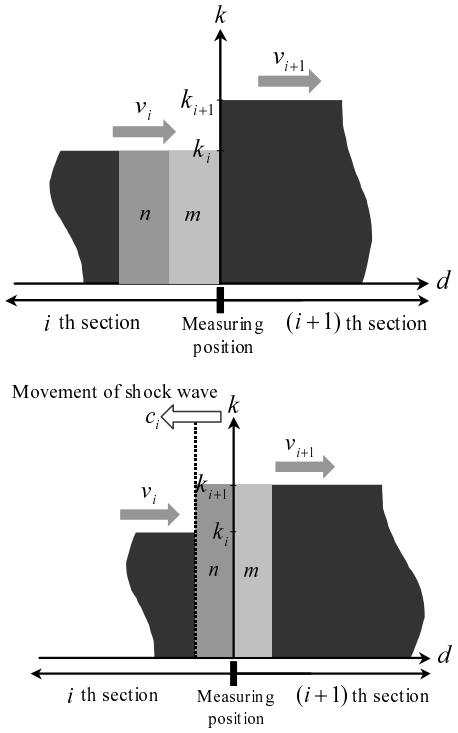

Fig. 4. Movement of shock wave in the case of $k_i(\tau) < k_{i+1}(\tau)$ and $c_i(\tau) \leq 0$

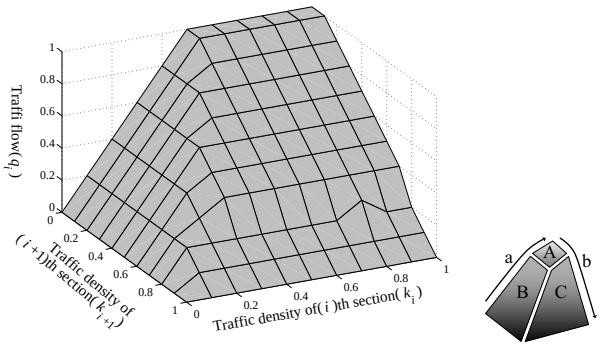

Fig. 5. Traffic flow behavior obtained from CA model

As the results, the flow model (6) $\sim$ (10) taking into account the discontinuity of the density can be summarized as follows:

$$q_i(\tau) = \begin{cases} \left(\frac{k_i(\tau)+k_{i+1}(\tau)}{2}\right) v_f \left(1 - \frac{k_i(\tau)+k_{i+1}(\tau)}{2k_{jam}}\right) \\ \quad if\ k_i(\tau) \geq k_{i+1}(\tau) \\ v_{f_i}\left(1 - \frac{k_i(\tau)}{k_{jam}}\right) k_i(\tau) \\ \quad if\ k_i(\tau) < k_{i+1}(\tau) \text{ and } c(\tau) > 0 \\ v_{f_{i+1}}\left(1 - \frac{k_{i+1}(\tau)}{k_{jam}}\right) k_{i+1}(\tau) \\ \quad if\ k_i(\tau) < k_{i+1}(\tau) \text{ and } c(\tau) \leq 0 \end{cases} \quad (11)$$

Figure 6 shows the traffic volume by proposed traffic flow model. We can see that the Fig.6 shows the similar characteristics to in Fig.5 that computed by Cellular Automaton (CA) model.

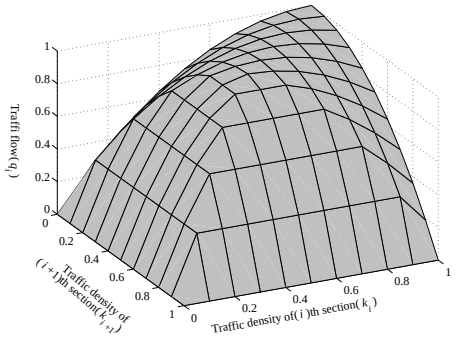

Fig. 6. Traffic flow behavior obtained from the proposed traffic flow model

3. Transformation to MLDS form

Although the HPN can represent the hybrid dynamical behavior of the TFCS including both continuous traffic flow and discrete traffic signal control, it is still not well formulated when some optimization problem is addressed. In this section, the MLDS form is introduced to formulate the Model Predictive Control (MPC) stated in the next section.
The MLDS form can generally be formalized as follows [6]:

$$\begin{aligned} \boldsymbol{x}(\kappa+1) &= \boldsymbol{A_\kappa x}(\kappa) + \boldsymbol{B_{1\kappa} u}(\kappa) \\ &\quad + \boldsymbol{B_{2\kappa}\delta}(\kappa) + \boldsymbol{B_{3\kappa} z}(\kappa) \quad (12) \\ \boldsymbol{y}(\kappa) &= \boldsymbol{C_\kappa x}(\kappa) + \boldsymbol{D_{1\kappa} u}(\kappa) \\ &\quad + \boldsymbol{D_{2\kappa}\delta}(t) + \boldsymbol{D_{3\kappa}}(\kappa) \quad (13) \\ \boldsymbol{E_{2\kappa}\delta}(\kappa) + \boldsymbol{E_{3\kappa} z}(\kappa) &\leq \\ \boldsymbol{E_{1\kappa} u}(\kappa) + \boldsymbol{E_{4\kappa} x}(\kappa) + \boldsymbol{E_{5\kappa}} & \quad (14) \end{aligned}$$

In the MLDS form,κ represents the sampling index. Note that sampling period T_s is eliminated the following. Equations (12), (13) and (14) are state equation, output equation and constraint inequality, respectively, where x, y and u are the state, output and input variable, whose components are constituted by continuous and/or 0-1 binary variables, $\delta(\kappa) \in \{0, 1\}$ and $z(\kappa) \in \Re$ represent auxiliary logical (binary) and continuous variables. The MLDS is known to be able to represent other forms of HDS such as Piece-Wise Affine (PWA), Hybrid Automaton (HA) and so on.
In the TFCS represented by HPN, equation (2) is directly transformed to the state equation in MLDS form by regarding the continuous marking as the state. Also, the TFSC only has binary input variable which denotes the state of the traffic signal (i.e. green or red). The output variable is not specified in our problem setting since all states are supposed to be measurable in this work.
The constraint inequality of (14) often plays an essential role to represent some nonlinearity which exists in the original system. In the TFCS, the nonlinearity appears in (11). In the following, this nonlinear constraint is transformed to the set of linear inequality constraints. The flow model developed in the previous section (shown in Fig.5) can be approximated by the Piece-Wise Affine (PWA) model shown in the right figure of Fig.5, which consists of three planes as follows:

Plane A: The traffic flow q_i is saturated ($k_i(\kappa) \geq a$ and $k_{i+1}(\kappa) \leq (k_{jam} - a)$)

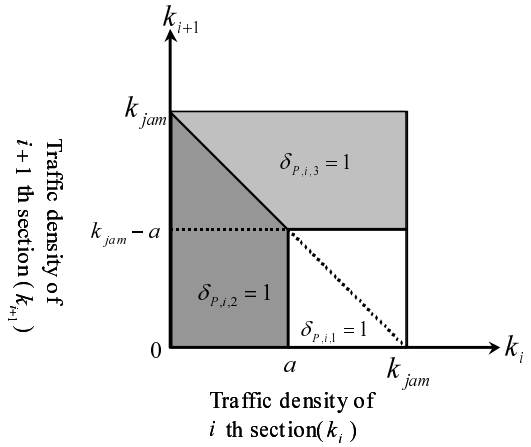

Fig. 7. Division of flow model by introducing auxiliary variables

Plane B: The traffic flow q_i is mainly affected by the quantity of traffic density $k_i(\kappa)$ ($k_i(\kappa) < a$ and $k_i(\kappa) + k_{i+1} \leq k_{jam}$)

Plane C: The traffic flow q_i is mainly affected by the quantity of traffic density $k_{i+1}(\kappa)$ ($k_{i+1}(\kappa) > k_{jam} - a$ and $k_i(\kappa) + k_{i+1} > k_{jam}$)

Here, a is the threshold value to specify the region of saturation characteristic of the traffic flow, that is, if $k_i(\kappa) \geq a$ and $k_{i+1}(\kappa) < k_{jam} - a$, the $q_i(\kappa)$ takes almost its maximum value q_{max}.

Figure 7 shows this partitions on $k_{i+1} - k_i$ plane. In order to derive the linear inequalities expression of the flow model, three auxiliary variables $\delta_{P,i,1}(\kappa)$, $\delta_{P,i,2}(\kappa)$ and $\delta_{P,i,3}(\kappa)$ are introduced, and are defined as follows:

$$[\delta_{P,i,1}(\kappa) = 1] \leftrightarrow \begin{cases} k_i(\kappa) & \geq & a \\ k_{i+1}(\kappa) & \leq & k_{jam} - a \end{cases} \tag{15}$$

$$[\delta_{P,i,2}(\kappa) = 1] \leftrightarrow \begin{cases} k_i(\kappa) & \leq & a - \varepsilon \\ k_i(\kappa) + k_{i+1}(\kappa) & \leq & k_{jam} \end{cases} \tag{16}$$

$$[\delta_{P,i,3}(\kappa) = 1] \leftrightarrow \begin{cases} k_{i+1}(\kappa) & \geq & k_{jam} - a + \varepsilon \\ k_i(\kappa) + k_{i+1}(\kappa) & \geq & k_{jam} + \varepsilon \end{cases} \tag{17}$$

$$\delta_{P,i,1}(\kappa) + \delta_{P,i,2}(\kappa) + \delta_{P,i,3}(\kappa) = 1 \tag{18}$$

where ε is a small tolerance.

By using these binary variables, the flow model $q_i(\kappa)$ given by (11) can be rewritten in a compact linear form as follows:

$$\begin{aligned} q_i(\kappa) &= q_{max}\delta_{P,i,1}(\kappa) + \frac{q_{max}k_i(\kappa)}{a}\delta_{P,i,2}(\kappa) \\ &\quad + \frac{q_{max}(1 - k_{i+1}(\kappa))}{a}\delta_{P,i,3}(\kappa) \end{aligned} \tag{19}$$

$$\sum_{i=1}^{3} \delta_{P,i,j}(\kappa) = 1$$

where $0 \leq k_i(\kappa) \leq k_{jam}$, $0 \leq k_{i+1} \leq k_{jam}$ $(= 1)$, and q_{max} is the maximum value of the traffic flow.

The equations (15) to (17) can be generalized as follows:

$$[\delta_{P,i,j}(\kappa) = 1] \quad \leftrightarrow \quad \left[\begin{bmatrix} k_i \\ k_{i+1} \end{bmatrix} \in \Re_j \right] \tag{20}$$

$$\Re_j \quad = \quad \left\{ \begin{bmatrix} k_i \\ k_{i+1} \end{bmatrix} : \boldsymbol{S_j k_i}(\kappa) \leq \boldsymbol{T_j} \right\} \tag{21}$$

where $\boldsymbol{k_i}(\kappa) = [k_i(\kappa)\ k_{i+1}(\kappa)]^T$ and $\boldsymbol{S_j}$ and $\boldsymbol{T_j}$ are the matrices with suitable dimensions. Also, this logical conditions can be transformed to inequalities

$$\boldsymbol{S_j k_i}(\kappa) \quad - \quad \boldsymbol{T_j} \leq \boldsymbol{M_j}^*[1 - \delta_{P,i,j}(\kappa)], \tag{22}$$

$$\boldsymbol{M_j^*} \quad \stackrel{\triangle}{=} \quad \max_{\boldsymbol{k_i} \in \Re_j} \boldsymbol{S_j k_i(\kappa)} - \boldsymbol{T_j}. \tag{23}$$

The flow $q_i(\kappa)$ of (19) can be represented by the vector form by using
$\boldsymbol{\delta_{P,i}}(\kappa) = [\delta_{P,i,1}(\kappa)\ \delta_{P,i,2}(\kappa)\ \delta_{P,i,3}(\kappa)]$ as follows:

$$q_i(\kappa) \quad = \quad f(\boldsymbol{\delta_{P,i}}(\kappa),\ \boldsymbol{k_{i+1}}(\kappa)) \tag{24}$$

$$\quad = \quad \sum_{j=1}^{3} (\boldsymbol{f_i^j}(\kappa)\boldsymbol{k_i}(\kappa) + h_i^j)\delta_{P,i,j}(\kappa) \tag{25}$$

where $\boldsymbol{f_i^j}$ and h_i^j are given as follows (see Fig. 7):

$$\boldsymbol{f_i^1} = [\ 0 \quad 0\ ] \quad , \quad h_i^1 = q_{max} \tag{26}$$

$$\boldsymbol{f_i^2} = [\ \tfrac{q_{max}}{a} \quad 0\ ] \quad , \quad h_i^2 = 0 \tag{27}$$

$$\boldsymbol{f_i^3} = [\ 0 \quad -\tfrac{q_{max}}{a}\ ] \quad , \quad h_i^3 = \frac{q_{max}}{a} \tag{28}$$

Next, we introduce an auxiliary bariable 'controlled traffic flow' $\boldsymbol{z_i}(\kappa) = [z_{i,1}(\kappa)\ z_{i,2}(\kappa)\ z_{i,3}(\kappa)]$ which implies the flow under the traffic signal control. $z_{i,j}(\kappa)$ is defined by

$$z_{i,j} = (\boldsymbol{f_i^j}(\kappa)\boldsymbol{k_i}(\kappa) + h_i^j)u_i(\kappa)\delta_{P,i,j}(\kappa). \tag{29}$$

That is,

$$q_i u_i = z_{i,1} + z_{i,2} + z_{i,3} \tag{30}$$

where $u_i(\kappa) \in \{0, 1\}$ denotes the binary control input which represents the state of the traffic signal. Then the equivalent inequalities to (29) is given as follows:

$$z_{i,j}(\kappa) \quad \leq \quad M_i u_i(\kappa)\delta_{P,i,j}(\kappa) \tag{31}$$

$$z_{i,j}(\kappa) \quad \geq \quad m_i u_i(\kappa)\delta_{P,i,j}(\kappa) \tag{32}$$

$$z_{i,j}(\kappa) \quad \leq \quad \boldsymbol{f_i^j k_i}(\kappa) + h_i^j - m_i(1 - u_i(\kappa)\delta_{P,i,j}(\kappa)) \tag{33}$$

$$z_{i,j}(\kappa) \quad \geq \quad \boldsymbol{f_i^j k_i}(\kappa) + h_i^j - M_i(1 - u_i(\kappa)\delta_{P,i,j}(\kappa)) \tag{34}$$

where M_i and m_i are

$$M_i \quad = \quad \max_{\boldsymbol{k_i}(\kappa) \in \Re_j} \left\{ \boldsymbol{f_i^j k_i}(\kappa) + h_i^j \right\} \tag{35}$$

$$m_i \quad = \quad \min_{\boldsymbol{k_i}(\kappa) \in \Re_j} \left\{ \boldsymbol{f_i^j k_i}(\kappa) + h_i^j \right\} \tag{36}$$

The product term $u_i(\kappa)\ \delta_{P,i,j}(\kappa)$ can also be replaced by another auxiliary logical variable[6].

As the results, the MLDS form for the TFCS can be formalized as follows:

$$\begin{aligned} \boldsymbol{x}(\kappa+1) &= \boldsymbol{A}\boldsymbol{x}(\kappa)+\boldsymbol{B}\boldsymbol{z}(\kappa) && (37)\\ \boldsymbol{z}(\kappa) &= diag(\boldsymbol{C}\boldsymbol{u}(\kappa))\boldsymbol{\delta}(\kappa) && (38)\\ \boldsymbol{E_2}\boldsymbol{\delta}(\kappa) &+ \boldsymbol{E_3}\boldsymbol{z}(\kappa) \\ &\leq \boldsymbol{E_1}\boldsymbol{u}(\kappa)+\boldsymbol{E_4}\boldsymbol{x}(\kappa)+\boldsymbol{E_5} && (39)\end{aligned}$$

where the element $x_i(\kappa)$ of $\boldsymbol{x}(\kappa) \in \Re^{|P|}$, is marking of the place p_{c_i} at the sampling index κ, the element $u_i(\kappa)(\in \{0,1\})$ of $\boldsymbol{u}(\kappa) \in Z^{|T|}$, is the state of the traffic signal installed at ith section and $\boldsymbol{\delta}(\kappa)$=$[\boldsymbol{\delta_P}(\kappa),\ \boldsymbol{\delta_M}(\kappa)]'$. Note that if there is no traffic signal installed at ith section, $u_i(\kappa)$ is always set to be one. $\boldsymbol{E_1}$, $\boldsymbol{E_2}$, $\boldsymbol{E_3}$, $\boldsymbol{E_4}$ and $\boldsymbol{E_5}$ of Fig.2 are described in the appendix.

4. Model Predictive Control for TFCS

The Model Predictive Control (MPC) [7]is one of the well-known paradigms for optimizing the systems with constraints and uncertainties. The Receding Horizon Control (RHC) policy is the key idea to realize the MPC. In the RHC, finite-horizon optimization is carried out based on the measured state at each sampling instant, and only the first control input is applied to the controlled plant. In this section, firstly, the RHC policy is briefly reviewed, then the optimization problem for the TFCS is formulated as the Mixed Integer Linear Programming (MILP). Finally, some idea to reduce the computational amount is described.

4.1. RHC for TFCS

In RHC policy, the control input at each sampling instant is decided based on the prediction of the behavior for next several sampling periods called the prediction horizon.

In order to formulate the optimization procedure, firstly, equation (37) is modified to evaluate the state and input variables in the prediction horizon as follows:

$$\begin{aligned} \boldsymbol{x}(\kappa+\lambda|\kappa) &= \boldsymbol{A^{\kappa}}x(\kappa) \\ &+\sum_{\eta=0}^{\lambda-1}\{\boldsymbol{A^{\eta}}(\boldsymbol{B}(diag(\boldsymbol{C}\boldsymbol{u}(\kappa+\lambda-1-\eta|\kappa))) \\ &\cdot\boldsymbol{\delta}(\kappa+\lambda-1-\eta|\kappa))\} \qquad (40)\end{aligned}$$

where $x(\kappa+\lambda|\kappa)$ denotes the predicted state vector at sampling index $\kappa+\lambda$, which is obtained by applying the input sequence, $u(\kappa),\cdots,u(\kappa+\lambda)$ to (37) starting from the state $x(\kappa|\kappa)=x(\kappa)$.

Now we consider the following control requirements that usually appear in the TFCS.

- *(R1)* Maximize the traffic flow over entire traffic network
- *(R2)* Avoid the frequent change of traffic signal
- *(R3)* Avoid the concentration of traffic mass in a certain section

These requirements can be realized by minimizing the following objective function.

$$\begin{aligned} &J(\boldsymbol{u}(\kappa|\kappa),\cdots,\boldsymbol{u}(\kappa+N|\kappa) \\ &\ ,\boldsymbol{x}(\kappa|\kappa),\cdots,\boldsymbol{x}(\kappa+N|\kappa),\boldsymbol{\delta}(\kappa|\kappa),\cdots,\boldsymbol{\delta}(\kappa+N|\kappa)) \\ &=\sum_{\lambda=1}^{N}\Bigg\{-\sum_i w_{1,i}\Bigg\{\Bigg(\boldsymbol{\Theta_i}\begin{bmatrix}\frac{x_i(\kappa+\lambda|\kappa)}{l_i}\\ \frac{x_{i+1}(\kappa+\lambda|\kappa)}{l_{i+1}}\end{bmatrix} \\ &+\boldsymbol{\Phi}\Bigg)'\boldsymbol{\delta_{M,i}}(\kappa+\lambda|\kappa)\Bigg\} \\ &-\sum_i w_{2,i}\Big\{1-\big|u_i(\kappa+\lambda|\kappa)-u_i(\kappa+\lambda+1|\kappa)\big|\Big\} \\ &+\sum_i w_{3,i}\left\{\left|\frac{x_i(\kappa+\lambda|\kappa)}{l_i}-\frac{x_{i+1}(\kappa+\lambda|\kappa)}{l_{i+1}}\right|\right\}\Bigg\} \qquad (41)\end{aligned}$$

where

$$\boldsymbol{\Theta_i}=\begin{bmatrix}0 & 0\\ \frac{q_{max}}{a} & 0\\ 0 & -\frac{q_{max}}{a}\end{bmatrix}\boldsymbol{\Phi_i}=\begin{bmatrix}q_{max}\\ 0\\ \frac{q_{max}k_i(\kappa)}{a}\end{bmatrix} \qquad (42)$$

N denotes the prediction horizon. Also, $w_{1,i}$, $w_{2,i}$ and $w_{3,i}$ are positive weighting parameters for ith section which satisfy $w_{1,i}+w_{2,i}+w_{3,i}=1$, and $0 \leq w_{1,i} \leq 1$, $0 \leq w_{2,i} \leq 1$ and $0 \leq w_{3,i} \leq 1$. The three terms in the left side of (41) correspond to the requirement (R1), (R2) and (R3), respectively.

As the results, the optimization problem can be formulated as follows:

find
$\boldsymbol{\delta(\kappa+\lambda|\kappa)}=[\ \boldsymbol{\delta_P(\kappa+\lambda|\kappa)},\ \boldsymbol{\delta_M(\kappa+\lambda|\kappa)}\ ]'$
$(\lambda=1,\cdots N)$
which minimizes (41)

subject to (22), (23),(26) $\sim$ (28),*and* (30) $\sim$ (39)

The MLDS formulation coupled with RHC scheme can be transformed to the canonical form of 0-1 Mixed Integer Linear Programming (MILP) problem with the objective function of (41). As the solver for the MILP, we have adopted the Branch-and-Bound (B&B) algorithm.

5. Numerical experiments

5.1. Signal control in intersections

In this section, we consider the signal control for the traffic network as shown in Fig. 8. This traffic network consists of four intersections, however, only the single-way of the traffic flow is allowed on each road. We assume that each car comes in from left side of this network every 30 seconds, and comes in from top side every 100 seconds. This implies that the horizontal traffic flow has higher amount than the vertical traffic flow. We have examined following four control conditions.

1. change the signal every 30 seconds (no control)
2. MPC with Prediction Horizon $N=1$
3. MPC with Prediction Horizon $N=4$ without consideration of uniformity of traffic density (w_3 was set to be zero)
4. MPC with Prediction Horizon $N=4$ with consideration of uniformity of traffic density

CA was used to simulate the movement of each car.

Table 1 shows the total number of cars which pass through this traffic network in both horizontal and vertical directions.

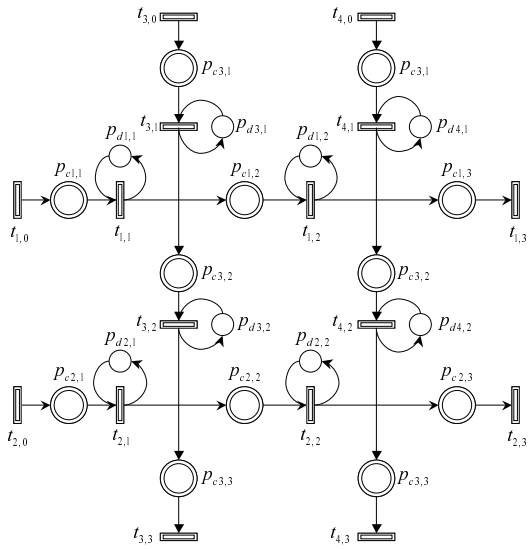

Fig. 8. HPN of traffic network with four single-way intersections

Table 1. Results of the intersection control

Control Conditions	1	2	3	4
Number of passing cars	18	41	56	58
Rate of Green signal in left to right road(%)	50	65.2	88.5	87.5

From these results, we can see that the MPC with longer prediction horizon enables more cars to get through this traffic network. Also, evaluations of the computational efforts are shown in Fig. 9.

1. Branch-and-Bound method
2. full search method

Here, "full search method" implies the algorithm where 0-1 combinations are directly allocated to the binary variables, and quadratic programming is applied in order to optimize the continuous variables. The optimal solution is found based on the performance criterion. From Fig.9, we can see that the difference of the computational efforts between two schemes becomes larger as the increase of the horizon.

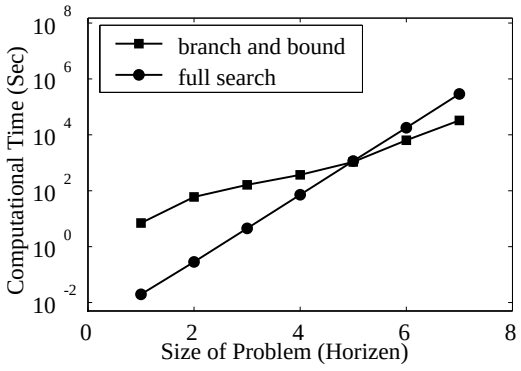

Fig. 9. Computational efforts

6. Conclusions

In this paper, we have proposed a new method for traffic signal control based on hybrid dynamical system theory. First of all, the synthetic modeling method for the Traffic Flow Control System (TCCS) has been proposed where the information on geometrical traffic network was modeled by using Hybrid Petri Net (HPN), whereas the information on the behavior of traffic flow was modeled by means of Mixed Logical Dynamical Systems (MLDS) form. The former allows us to easily apply our method to complicated and wide range of traffic network due to its graphical understanding. The latter enables us to optimize the control policy for the traffic signal by means of its algebraic manipulability and use of model predictive control framework. Secondly, the shock wave model has been introduced in order to treat the discontinuity of the traffic flow. By approximating the derived flow model with piece-wise linear function, the flow model has been naturally coupled with the MLDS form. Finally, the model predictive control problem for the TFCS has been formulated. This formulation has been recasted to the 0-1 Mixed Integer Linear Programming (MILP) problem. Some numerical experiments have been carried out, and have shown the usefulness of the proposed design framework.

References

[1] S.Darabha and K.R.Rahagopal "Intelligent cruise control systems and traffic flow stabilityhTransportation Research Part C 7(1999) 329-352

[2] K. Nagel and M. Schreckenberg. A Cellular Automaton Model for Freeway Traffic. Journal de Physique I France, 2: 2221, 1992.

[3] First-order hybrid Petri nets: a model for optimization and control. , IEEE Trans. on Robotics and Automation, Vol. 16, No. 4, pages 382-399. 2000.

[4] Richard Haberman Mathematical Models

[5] Chaudahuri, P.P, et al. : Additive Cellular Automata - Theory and Applications, IEEE Computer Society Press(1997)

[6] A. Bemporad, M. Morari, " Control of systems integrating logic, dynamics, and constraints", Automatica, Vol. 35, n. 3, p. 407-427, 1999.

[7] M. Morari ; E. Zafirou. Robust Process Control. Prentice Hall, 1989.