Large-Scale Traffic Network Control Based on Convex Programming Coupled with B&B Strategy

Tatsuya Kato Student Member (Nagoya University,kato@okuma.nuee.nagoya-u.ac.jp)
YoungWoo Kim Non-member (Toyota Technological Institute)
Shigeru Okuma Member (Nagoya University,okuma@okuma.nuee.nagoya-u.ac.jp)
Tatsuo Narikiyo Member (Toyota Technological Institute)

Keywords: HDS, HPN, MLDS, traffic flow control

In this paper, a new method for large-scale traffic network control is proposed. First of all, we formulate, based on the Mixed Integer NonLinear Programming (MINLP) problem, the traffic network control system that is one of typical hybrid systems with nonlinear dynamics. Generally, it is difficult to find the global optimal solution to the nonlinear programming problem. However, if the problem can be recast to the convex programming problem, the global optimal solution is easily found by applying the efficient method such as Steepest Descent Method (SDM). We use in this paper general performance criteria for traffic network control and show that although the problem contains non-convex constraint functions as a whole, the generated sub-problems are always included in the class of convex programming problem.

The geometrical information on the traffic network is characterized by a Hybrid Petri Net (HPN). Then, the algebraic behavior of traffic flow is transformed into a Mixed Logical Dynamical System (MLDS) form in order to introduce an optimization technique.

In order to achieve high control performance of the traffic network with dynamically changing traffic flow, we adopt Model Predictive Control (MPC) policy. Note that MLDS formulation often encounters multiplication of two decision variables, and that without modification, it cannot be directly applied to MPC scheme. One way to avoid the multiplication is to introduce a new auxiliary variable to represent it. And then it becomes a linear system formally. However, as we described before, the introduction of discrete variables yields substantial computational amounts increased. A new method for this type of control problem is prsoposed. Although the system representation is nonlinear, MPC policy is successfully applied by means of the proposed Branch & Bound strategy. This implies we can find global optimal solution in a short time since no more auxiliary variables (such as δ_P and δ_M) are introduced.

Fig. 2 shows the HPN model of the ith intersection, where the notation for other than southward entrance lane is omitted.

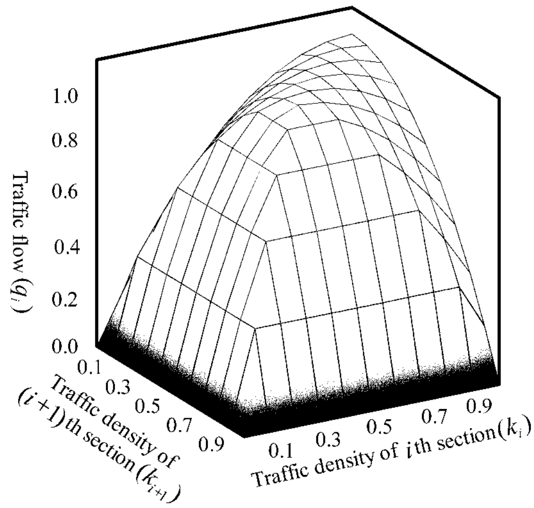

Fig. 1. Traffic flow behavior obtained from the proposed traffic flow model

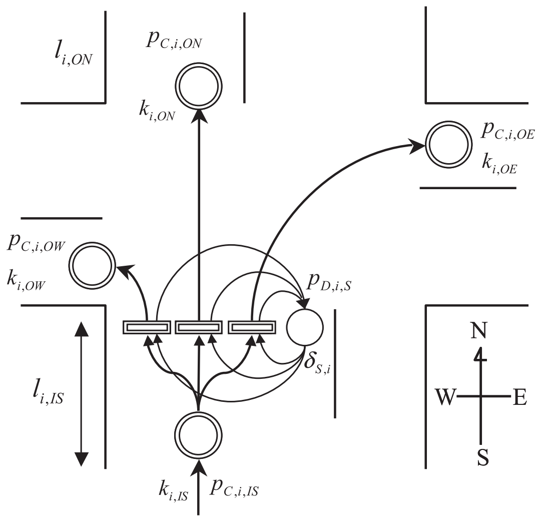

Fig. 2. HPN model of the intersection

Paper

Large-Scale Traffic Network Control Based on Convex Programming Coupled with B&B Strategy

Tatsuya Kato* Student Member
YoungWoo Kim** Non-member
Shigeru Okuma* Member
Tatsuo Narikiyo** Member

This paper presents a new framework for traffic flow control based on an integrated model description by means of a Hybrid Dynamical System (HDS). The geometrical information on the traffic network is characterized by a Hybrid Petri Net (HPN). Then, the algebraic behavior of traffic flow is transformed into a Mixed Logical Dynamical System (MLDS) form in order to introduce an optimization technique. These expressions involve both continuous evolution of traffic flow and event driven behavior of traffic light. HPN allows us to formulate the problem easily for complicated and large-scale traffic network due to its graphical understanding. MLDS enables us to optimize the control policy for traffic light by means of its algebraic manipulability and use of model predictive control framework. Since the behavior represented by HPN can be directly transformed into corresponding MLDS form, the seamless incorporation of two different modeling schemes provide a systematic design scenario for traffic flow control.

Keywords: HDS, HPN, MLDS, traffic flow control

1. Introduction

With the increasing number of automobiles and complication of traffic network, the traffic flow control becomes one of significant economic and social issues. Desirable characteristics of the traffic network control can be stated as follows: (1) precise modeling of traffic flow, (2) well-defined formulation with algebraic manipularity, and (3) applicability to large-scale traffic network. To meet these characteristics, many approaches have been proposed in the Traffic Flow Control (TFC) community.

In the microscopic approaches such as Cellular Automaton (CA) based model [(1)–(3)] and the Follow-the-Leader (FL) model [(4)], the behavior of each vehicle is represented based on the relationships with those of neighboring vehicles. The simulation results concerning high-speed driving, acceleration/deceleration driving, and so on, showed the marked similarities to the measured real data. However, this approach is not suitable for traffic light control of the inner-city districts because it requires enormous computational efforts to obtain all vehicles' behaviors in real time. Furthermore, the simulation for all combinations of the traffic lights in a given network will be impossible.

On the other hand, a fluid approximation model was proposed to represent traffic flow in the macroscopic approach. Although this model expresses well the (continuous) behavior of the traffic flow on the freeway, it is impossible that this model can be directly applied to the urban traffic network, since urban traffic network involves discontinuities of the traffic flow density, controlled by stop traffic lights at the intersections. The idea of *shock wave* [(5)–(8)], which represents the progress of the boundary of two neighboring different density areas, has been introduced in order to handle this discontinuity of the traffic density. However, it is not straightforward to manipulate the shock wave in algebraic form since the fluid traffic flow model has highly complicated nonlinear dynamics.

From these points of view, the authors proposed the piece-wise affine traffic flow model (9), where the traffic flow was represented with the traffic densities of two consecutive districts in order to consider the behavior of shock wave. The traffic flow dynamics were optimized based on Mixed Logical Dynamical System (MLDS) framework. The method used in (9) is the well-established optimization procedure. However, the method based on Mixed Integer Linear Programming (MILP) problem associated with piece-wise affine traffic flow dynamics is unfit for large-scale traffic network control, since it is computationally expensive. Consider the traffic light control of a pedestrian crossover on a one-way street. The previous method requires one binary variable (δ_S) to represent traffic light states, three binary variables (δ_P) to represent the traffic flow dynamics, and three binary variables (δ_M) to optimize the dynamics, transforming it to the linear form. This means in the worst case that MILP has 2^7 sub problems to solve.

In this paper, a new method for large-scale traffic net-

* Nagoya University
kato,okuma@nuee.nagoya-u.ac.jp
** Toyota Technological Institute
kim,n-tatsuo@toyota-ti.ac.jp

work control is proposed. First of all, we formulate, based on the Mixed Integer NonLinear Programming (MINLP) problem, the traffic network control system that is one of typical hybrid systems with nonlinear dynamics. Generally, it is difficult to find the global optimal solution to the nonlinear programming problem. However, if the problem can be recast to the convex programming problem, the global optimal solution is easily found by applying the efficient method such as Steepest Descent Method (SDM). We use in this paper general performance criteria for traffic network control and show that although the problem contains nonconvex constraint functions as a whole, the generated sub-problems are always included in the class of convex programming problem.

In order to achieve high control performance of the traffic network with dynamically changing traffic flow, we adopt Model Predictive Control (MPC) policy. Note that MLDS formulation often encounters multiplication of two decision variables, and that without modification, it cannot be directly applied to MPC scheme. One way to avoid the multiplication is to introduce a new auxiliary variable to represent it. And then it becomes a linear system formally. However, as we described before, the introduction of discrete variables yields substantial computational amounts increased. A new method for this type of control problem is proposed. Although the system representation is nonlinear, MPC policy is successfully applied by means of the proposed Branch & Bound strategy. This implies we can find global optimal solution in a short time since no more auxiliary variables (such as δ_P and δ_M [9]) are introduced.

This paper is organized as follows. In chapter 2, we present the integrated model description for the large-scale traffic network control, where the geometrical information on the traffic network is characterized by using Hybrid Petri Net (HPN) [10][11], and the algebraic description of the traffic flow is provided in consideration of the shock wave. And then they are integrated into MLDS formulation. In chapter 3, the developed model is recast to the canonical form of MINLP. And the proposed B&B algorithm coupled with convexity analysis is applied to the problem. Finally the usefulness of the proposed method is verified through some numerical experiments.

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 lights. 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 presented, which provides both graphical and algebraic descriptions for the TFCS.

2.1 HPN Model of Traffic Network The HPN is one of the useful tools to model and visualize the system behavior with both continuous and discrete variables. Fig.2 shows the HPN model for the road of Fig.1. In Fig.2, each section i of l_i-meter long constitutes

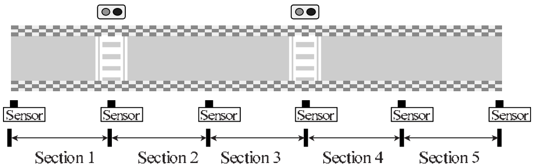

Fig. 1. Straight road

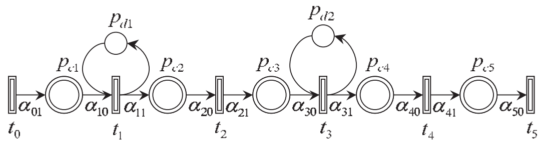

Fig. 2. HPN model of straight road

the straight road, and two traffic lights are installed at the point of crosswalks. The HPN has a structure of $N = (P, T, q, I_+, I_-, M^0)$. The set of places P is partitioned into both 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 light where green 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 light (i.e. binary value). Note that each light is supposed to have only two states 'go (green)' or 'stop (red)' for 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. 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 the HPN is represented by a simple first order differential equation for each continuous place $p_{c_i} \in P_c$ as follows :

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

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 expression supposing that $q_j(\tau)$ is constant during two successive sampling instants as follows :

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

where κ and T_s are sampling index and period, respectively.

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.*

In the proposed modeling, processes of parallel processing, formal analysis, mutual exchange of shared resources, and synchronization are expressed with the Petri Net. If the Petri Net model is not introduced, the modeling is still possible but it is very difficult to express the processes. In our previous work [9], the Petri Net based model was proposed and it was shown to be very useful.

2.2 Traffic flow dynamics 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

$$q_i(\tau) = -\frac{(k_i(\tau) + k_j(\tau))}{2}\frac{v_i(k_i(\tau)) + v_j(k_j(\tau))}{2} \cdots\cdots\cdots (3)$$

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

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

with eq.(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 exists 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 lights. In order to treat the discontinuities of the density among neighboring sections (i.e. neighboring continuous places), the idea of 'shock wave' [12] is introduced as follows. We consider the case shown in Fig.3 where the traffic density of ith section is higher than that of jth 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 jth sections as follows [12]:

$$c_i(\tau) = \frac{v_i(\tau)k_i(\tau) - v_j(\tau)k_j(\tau)}{k_i(\tau) - k_j(\tau)} \cdots\cdots\cdots (5)$$

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

(C1) $k_i(\tau) < k_j(\tau)$, and $c_i(\tau) > 0$,
(C2) $k_i(\tau) < k_j(\tau)$, and $c_i(\tau) \leq 0$,
(C3) $k_i(\tau) > k_j(\tau)$,
(C4) $k_i(\tau) = k_j(\tau)$ (no shock wave).

Firstly, in both cases of (C1) and (C2) where $k_i(\tau)$ is smaller than $k_j(\tau)$, the vehicles passing through the density boundary (dotted line) reduce their speeds. The movement of the shock wave is illustrated in Fig.3 ($c_i(\tau) > 0$). The 'measuring position' implies the position where transition t_i is assigned. Since the traffic flow $q_i(\tau)$ represents the number of vehicle passing through the measuring position per unit time. In Fig.3, n and m represent the area of the corresponding rectangular, the square measure of that implies the size of traffic flow. These considerations lead to the following models:

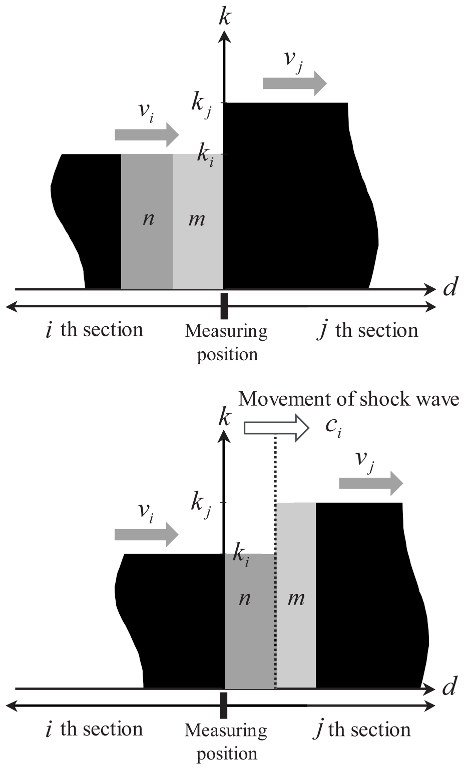

Fig. 3. Movement of shock wave in the case of $k_i(\tau) < k_j(\tau)$ and $c_i(\tau) > 0$

$$q(k_i(\tau), k_j(\tau)) = \begin{cases} -\left(\frac{k_i(\tau)+k_j(\tau)}{2}\right) v_f \left(1 - \frac{k_i(\tau)+k_j(\tau)}{2k_{jam}}\right) \\ \quad if \;\; k_i(\tau) \geq k_j(\tau) \\ -v_{f_i}\left(1 - \frac{k_i(\tau)}{k_{jam}}\right) k_i(\tau) \\ \quad if \;\; k_i(\tau) < k_j(\tau) \text{ and } c_i(\tau) > 0 \\ -v_{f_j}\left(1 - \frac{k_j(\tau)}{k_{jam}}\right) k_j(\tau) \\ \quad if \;\; k_i(\tau) < k_j(\tau) \text{ and } c_i(\tau) \leq 0 \end{cases} \cdots\cdots\cdots (6)$$

where $0 \leq k_i(\tau) \leq k_{jam}, 0 \leq k_j(\tau) \leq k_{jam}$.

2.3 Traffic Network Model at an Intersection In this subsection, we develop traffic network model at an intersection. Fig.5 shows the HPN model of the ith intersection, where the notation for other than southward entrance lane is omitted. In Fig.5, $l_{i,IS}$ and $l_{i,ON}$ are the length of the districts $p_{C,i,IS}$ and $p_{C,i,ON}$, and the numbers of the vehicles at $p_{C,i,IS}$ and $p_{C,i,ON}$ are $k_{i,IS} \cdot l_{i,IS}$ and $k_{i,ON} \cdot l_{i,ON}$, respectively. The vehicles in $p_{C,i,IS}$ are assumed to have the probability $\zeta_{i,SW}$, $\zeta_{i,SN}$,

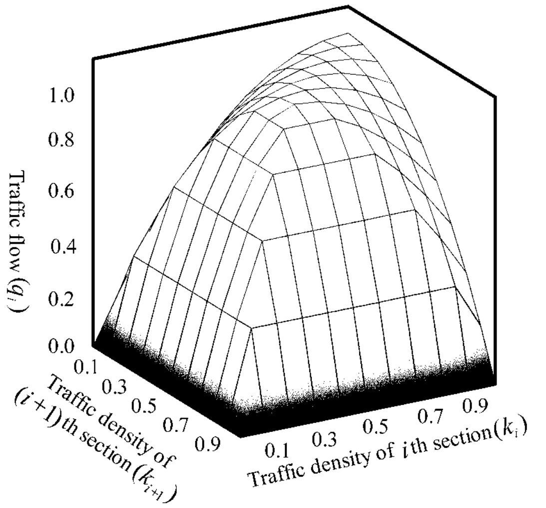

Fig. 4. Traffic flow behavior obtained from the proposed traffic flow model

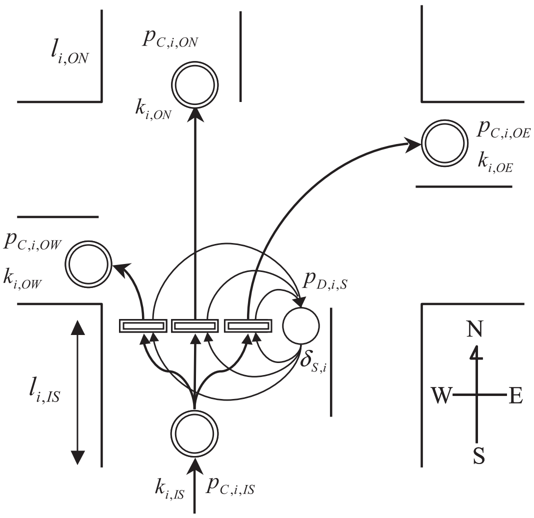

Fig. 5. HPN model of the intersection

and $\zeta_{i,SE}$ to proceed into the district corresponding to $p_{C,i,OW}$, $p_{C,i,ON}$, and $p_{C,i,OE}$ as follows,

$$k_{i,SW}(\tau) = k_{i,IS}(\tau)\zeta_{i,SW}, \quad \cdots\cdots (7)$$

$$k_{i,SN}(\tau) = k_{i,IS}(\tau)\zeta_{i,SN}, \quad \cdots\cdots (8)$$

$$k_{i,SE}(\tau) = k_{i,IS}(\tau)\zeta_{i,SE}. \quad \cdots\cdots (9)$$

Note that these probabilities are determined by the traffic network structure, and satisfy $0 \leq \zeta_{i,SW}(\tau) \leq 1$, $0 \leq \zeta_{i,SN}(\tau) \leq 1$, $0 \leq \zeta_{i,SE}(\tau) \leq 1$, and $\zeta_{i,SW}(\tau) + \zeta_{i,SN}(\tau) + \zeta_{i,SE}(\tau) = 1$. Therefore, the traffic flows of the three directions are represented by

$$q\left(k_{i,SN}(\tau), k_{i,ON}(\tau)\right), \quad \cdots\cdots (10)$$

$$q\left(k_{i,SW}(\tau), k_{i,OW}(\tau)\right), \quad \cdots\cdots (11)$$

$$q\left(k_{i,SE}(\tau), k_{i,OE}(\tau)\right). \quad \cdots\cdots (12)$$

Since the probability ζ includes the affection of yellow light, yellow light is not explicitly represented in Fig.5.

3. Model Predictive Control of Traffic Network Control System

In this section, the MLDS form is introduced to formulate the Model Predictive Control (MPC) as shown in the next subsection. The MLDS form can generally be formalized as follows [13]:

$$\boldsymbol{x}(\kappa+1) = \boldsymbol{A_\kappa x}(\kappa) + \boldsymbol{B_{1\kappa} u}(\kappa) + \boldsymbol{B_{2\kappa} \delta}(\kappa) + \boldsymbol{B_{3\kappa} z}(\kappa), \quad \cdots\cdots (13)$$

$$\boldsymbol{y}(\kappa) = \boldsymbol{C_\kappa x}(\kappa) + \boldsymbol{D_{1\kappa} u}(\kappa) + \boldsymbol{D_{2\kappa} \delta}(\kappa) + \boldsymbol{D_{3\kappa}}(\kappa), \quad \cdots\cdots (14)$$

$$\boldsymbol{E_{2\kappa} \delta}(\kappa) + \boldsymbol{E_{3\kappa} z}(\kappa) \leq \boldsymbol{E_{1\kappa} u}(\kappa) + \boldsymbol{E_{4\kappa} x}(\kappa) + \boldsymbol{E_{5\kappa}}. \quad \cdots\cdots (15)$$

where κ represents the sampling index. † The equations (13), (14) and (15) are a state equation, an output equation and a constraint inequality, respectively, where $x(\kappa)$, $y(\kappa)$ and $u(\kappa)$ are a state, a output and an 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.

Although the MLDS description allows us to represent systematically nonlinear behavior of the system dynamics by introducing auxiliary variables to approximate it to the (piece-wise) linear dynamics, methods for finding optimal solution were computationally expensive. This is because of a number of auxiliary variables introduced. This paper proposes a new method for optimizing the system with nonlinear dynamics, where nonlinear dynamics is not linearized in a piece-wise manner. The developed Mixed Integer NonLinear Programming (MINLP) problem is solved by proposed Branch & Bound strategy.

3.1 MLDS Formulation The traffic flow q is the function of k_i, k_j which are the traffic densities of the two consecutive district i and j, and contains nonlinearity because of the multiplications of the two variables as in eq.(6). Since q is also the function of traffic light, traffic flow adjoining the intersection i can be represented by introducing (continuous) auxiliary variable z as follows

$$z_{i,\tilde{i}}(\kappa) = q\Big(\frac{x_i(\kappa)}{l_i}, \frac{x_{\tilde{i}}(\kappa)}{l_{\tilde{i}}}\Big) u_j(\kappa) \quad \cdots\cdots (16)$$

where the traffic light at intersection i is $u_i(\kappa)$. Here, $BLUE(RED)$ light of the north-south orientation and $RED(BLUE)$ light of the east-west orientation are represented b 0 (1). With z of eq.(16), the state equation and the constraint inequality are formulated as follows

$$\boldsymbol{x}\ (\kappa+1) = \boldsymbol{Ax}\ (\kappa) + \boldsymbol{Bz}(\kappa) \quad \cdots\cdots (17)$$

$$\boldsymbol{E_{2\kappa} z}(\kappa) \leq \boldsymbol{E_{1\kappa} u}(\kappa) + \boldsymbol{E_{4\kappa} x}(\kappa) + \boldsymbol{E_{5\kappa}} \quad \cdots\cdots (18)$$

where $\boldsymbol{x} = [x_1, x_2, ..., x_n]^T$, $u_j \in \{0, 1\}$, $\tilde{i} \in t_j^\bullet$ and $\boldsymbol{A}$ is the matrix with the suitable dimension. Equation (18) represents the logical relationship (proposition) of eq.(6) and the maximum capacity constraints of each place, and the firing speed of each transitionm respectively.

Although the mixed logic dynamical system is represented in a compact form as eq.(17) and eq.(18), this

† Note that for simplicity, the sampling period T_s is eliminated from the description of the MLDS form.

cannot be directly applied to the model predictive control scheme, since z has the multiplication of three decision variables (z amounts to $\alpha\ u \cdot x^2$).

However, if we know the values of k_i and k_j, q is uniquely determined. We propose in this paper a new B&B algorithm which consists of refining process as well as conventional branching and bounding processes. The combination of the three processes makes it easy to handle the special type of the nonlinear programming problem.

3.2 Model Predictive Control coupled with B&B strategy The Model Predictive Control (MPC) [14] [15] is one of the well-known paradigms for optimizing the systems with constraints and uncertainties. In the MPC policy, the control input at each sampling instant is decided based on the prediction of the behavior for the next several sampling periods called the prediction horizon. In order to formulate the optimization procedure, firstly, eq.(17) is modified to evaluate the state and input variables in the prediction horizon as follows:

$$\begin{aligned}&\boldsymbol{x}(\kappa+\lambda|\kappa)\\&=\boldsymbol{A}^{\boldsymbol{\lambda}}x(\kappa)+\sum_{\eta=0}^{\lambda-1}\{\boldsymbol{A}^{\boldsymbol{\eta}}(\boldsymbol{Bz}(\kappa+\lambda-1-\eta|\kappa)\\&\qquad\cdot\boldsymbol{u}(\kappa+\lambda-1-\eta|\kappa))\}\ \cdots\cdots\cdots\cdots\cdots\cdots\ (19)\end{aligned}$$

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 eq.(19) starting from the state $x(\kappa|\kappa)=x(\kappa)$.

The MPC scheme with the MLDS formulation can be transformed to the canonical form of 0-1 MINLP problem. We propose a new Branch-and-Bound (B&B) algorithm to solve this class of programming problem, since the conventional method is not applicable for MINLP problem. The proposed method guarantees for solution optimality based on convexity analysis coupled with the proposed branch and bound strategy. The proposed algorithm is the combination of branching process, bounding process, and refining process as follows.

Branching Process: If the solution to the given subproblem is proven to satisfy 0-1 constraints, the algorithm constructs two new sub-problems, in which some variables are fixed at zero or one. Then, conventional nonlinear programming (NLP) method is applied to the problem.

Bounding Process: The sub-problem is pruned off from the enumeration tree if at least one of following conditions is met.

(1) The solution is infeasible.

(2) The solution to the sub-problem has a higher cost than best integer solution(s) discovered (which was proven to satisfy all 0-1 constraints).

Refining Process: If the constraint equations and/or inequalities have the nonlinear term(s) of known variables, these terms are reformed to have a linear form of the branching variables. This process is carried out by assigning the values of the known variables to the nonlinear terms.

By introducing the refining process, the nonlinear function, $\boldsymbol{z}(\kappa+i)$ in eq.(19) comes to have a linear form. However, if we apply the MPC scheme with the MINLP problem, the selection of the branching procedure should be carefully carried out. For example, $x(\kappa+1)$ can be easily obtained since q is known variable at the time of κ. However, enumeration of $x(\kappa+2)$ contains the multiplication of variables as follows,

$$\begin{aligned}\boldsymbol{x}(\kappa+2|\kappa)&=\boldsymbol{A}(\boldsymbol{Ax}(\kappa|\kappa)+\boldsymbol{Bz}(\kappa|\kappa)\boldsymbol{u}(\kappa|\kappa))\\&\quad+\boldsymbol{Bz}(\kappa+1|\kappa)\boldsymbol{u}(\kappa+1|\kappa)\ \cdots\cdots(20)\\&=\boldsymbol{A}^2\boldsymbol{x}(\kappa|\kappa)+\boldsymbol{ABz}(\kappa|\kappa)\boldsymbol{u}(\kappa|\kappa)\\&\quad+\boldsymbol{Bz}(\kappa+1|\kappa)\boldsymbol{u}(\kappa+1|\kappa)\ \cdots\cdots(21)\end{aligned}$$

Therefore the refining process (RP) and the branching process (BP) should be in the following order ; RP of $q(\kappa)\rightarrow$ BP of $u(\kappa)\rightarrow$ RP of $q(\kappa+1)\rightarrow$ BP of $u(\kappa+1)$ and so on.

The proposed algorithm is formulated as follows.

Modified Branch-and-Bound algorithim

Step 1(Initialization): Set List $L\equiv\{P_0\}$, $\xi^*\equiv\infty$, and $l=0$. Here, P_0 is the problem in which all 0-1 constraints are relaxed.

Step 2(Optimality Assessment): If $L=\phi$, terminate the algorithm. Here, if $\xi^*<\infty$, the solution corresponding to ξ^* is the optimal solution. Otherwise, there is no feasible solution.

Step 3(Selection of Sub-Problem): Select sub-problem P_k from the list L and substitute L with $L-\{P_k\}$.

Step 4(Bounding Process): Solve P_k. If P_k has no feasible solution, go to Step 2. If P_k has feasible solution with $\xi^k\geq\xi^*$, go to Step 2. If P_k has feasible solution with $\xi^k<\xi^*$, go to Step 5.

Step 5(Renewal of Incumbent Solution): If the solution to P_k satisfies all 0-1 constraints, substitute $\boldsymbol{x}^*$ with $\boldsymbol{x}^k$, $\xi^*\equiv\xi^k$, and go to Step 2.

Step 6(Selection of Branching Variable): If the solution to P_k violates at least one of 0-1 variables, set N whose elements are 0-1 variables, but they do not satisfy 0-1 constraints yet. Select the branching variable x_s^k whose predicted sampling index is closest to the present sampling index among N.

Step 7(Branching Process): Generate two new subproblems P_{l+1} and P_{l+2}. Impose the constraints, $x_x^k=0$ on P_{l+1} and $x_x^k=1$ on P_{l+2}, respectively. Substitute L with $L\cup\{P_{l+1},P_{l+2}\}$ and l with $l+2$.

Step 8(Refining Process): If there is any decision variable which is dependent on x_s^k, substitute the value of x_s^k to the variable, and reformulate P_{l+1} and P_{l+2} in a linear form. And go to Step 2.

By selecting the branching variable x_s^k whose predicted sampling index is closest to the present sampling index, all the dependent variables of x_s^k can be represented as the numerical form with the lower order. In our problem setting, the refining process of Step 8 can be translated as follows.

Step 8-1: Obtain $k(\Gamma)$ from the value of $x(\Gamma)$.

Step 8-2: Obtain $q(\Gamma)$ by substituting $k(\Gamma)$ to the equation (6).
Step 8-3: Obtain $z(\Gamma)$ by substituting $q(\Gamma)$ and $u(\Gamma)$ to the equation (16).
Step 8-4: Reformulate the sub-problem in a linear form.

Here, the decision variable $u(\Gamma)$ is selected as the branching variable and nonlinear term, $z(\Gamma)$ is obtained based on the known variables $q(\Gamma)$ and $u(\Gamma)$. If all the multivariate nonlinear terms can be transformed to have first order or zero order form of unknown variables by applying refining process, you can conceal nonlinear constraints from the problem setup. Consider Step 8-1 to Step 8-4, where by substituting the value of the branching variable $u(\Gamma)$, $z(\Gamma)$ comes to have zero order form of the unknown variables. This implies that we do not need to introduce the auxiliary (binary) variables (9) in order to represent the logical relations between the three modes of eq.(6) and corresponding dynamics, respectively, and to associate with optimization scheme.

Note that the bounding process is very important to reduce the problem size to a computationally manageable one. However this process is effective only when the performance criterion is a convex function. In the next section, the convexity analysis is applied to our problem setup.

4. Convexity Analysis

The problem we formulated in the previous section is recast to the convex programming problem in this subsection. The convex programming problem, where the constraint and objective functions are convex, has become quite popular recently for a number of reasons. Some of them are summarized as follows : (1) The global optimality is guaranteed for the obtained solution, (2) The attractive algorithm is easily applied, obtaining the solution with high speed due to the simple structure of the problem, and (3) The bounding process can be efficiently applied for the MINLP problem.

4.1 Performance Criteria In this subsection we firstly introduce the well-known performance criteria of traffic network control system and show they can be realized with convex functions. The following performance criteria are introduced in this paper: (1) maximization of traffic flow and (2) minimization of traffic density difference between neighboring districts. These criteria are numerically represented as follows,

$$f = \sum_{i=n+1}^{n+m} z_i, \quad \cdots\cdots (22)$$

and

$$f = \sum_{i=0}^{n-1} |x_i - x_{i+1}| \quad \cdots\cdots (23)$$

In order to verify the convexity of (22), we firstly show the traffic flow dynamics with three modes are convex functions at each mode, and show that these dynamics at each mode are continuous to the neighboring ones. By using this continuity, the overall dynamics of the traffic flow is proven to be convex.

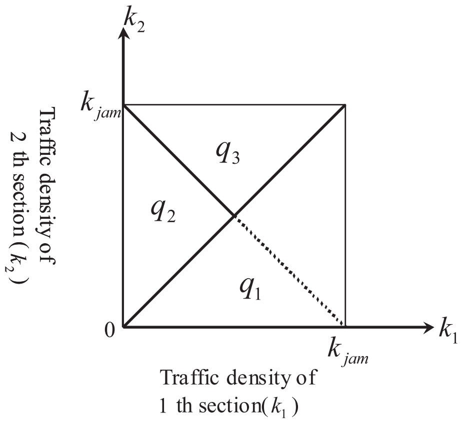

Fig. 6. Assignation of traffic flow mode

Consider Fig.6, where each mode of traffic flow is assigned. Since the Hessian matrices of $q_1(\frac{x_1}{l_1}, \frac{x_2}{l_2})$, $q_2(\frac{x_1}{l_1}, \frac{x_2}{l_2})$, and $q_3(\frac{x_1}{l_1}, \frac{x_2}{l_2})$ are nonsingular as follows,

$$\begin{aligned} \nabla^2 q_1(\boldsymbol{x}) &= \left[\frac{\partial^2 q_1(\boldsymbol{x})}{\partial x_1 \partial x_2}\right] \\ &= \begin{bmatrix} \frac{v_f}{2x_{jam}} & \frac{v_f}{2x_{jam}} \\ \frac{v_f}{2x_{jam}} & \frac{v_f}{2x_{jam}} \end{bmatrix} \geq 0 \quad \cdots\cdots (24) \end{aligned}$$

$$\begin{aligned} \nabla^2 q_2(\boldsymbol{x}) &= \left[\frac{\partial^2 q_2(\boldsymbol{x})}{\partial x_1 \partial x_2}\right] \\ &= \begin{bmatrix} \frac{v_f}{2x_{jam}} & 0 \\ 0 & 0 \end{bmatrix} \geq 0 \cdots\cdots (25) \end{aligned}$$

and

$$\begin{aligned} \nabla^2 q_3(\boldsymbol{x}) &= \left[\frac{\partial^2 q_3(\boldsymbol{x})}{\partial x_1 \partial x_2}\right] \\ &= \begin{bmatrix} 0 & 0 \\ 0 & \frac{v_f}{2x_{jam}} \end{bmatrix} \geq 0 \cdots\cdots (26) \end{aligned}$$

they are convex at each mode.

In order to show the convexity of the overall dynamics of the traffic flow, we use following lemma :

Lemma 1: The neighboring two closed convex dynamics $D_1(\Psi{=}(\psi_1,\ \psi_2,\ \cdots,\ \psi_n))$ and $D_2(\Psi)$ are convex if they are continuous at the boundary point $(\hat{\psi}_1, \hat{\psi}_2, \cdots, \hat{\psi}_n) \in \Theta$ ($\Theta = \overline{D_1(\Psi)} \bigcap \overline{D_2(\Psi) \backslash D_1(\Psi)}$) and satisfy that

if for $\forall\ i$, γ and μ

$$\nabla_\gamma D_1(\Psi)\Big|_{\psi_i=\hat{\psi}_i} \leq (\geq) \nabla_\gamma D_2(\Psi)\Big|_{\psi_i=\hat{\psi}_i} \cdots (27)$$

then

$$\nabla^2_{\mu,\mu} D_1(\Psi)\Big|_{\psi_i=\hat{\psi}_i} \leq (\geq) \nabla^2_{\mu,\mu} D_2(\Psi)\Big|_{\psi_i=\hat{\psi}_i} \cdots\cdots (28)$$

where overline denote the closure of the set, $1 \leq i, \gamma, \mu \leq n$, $\nabla_\gamma D$ is the γth element of ∇D, and $\nabla^2_{\mu,\mu} D$ is the $(\mu,\ \mu)$th element of the matrix $\nabla^2 D$.

The continuity at the boundary is easily confirmed by letting $k_1(\tau) = k_2(\tau) = k(\tau)$ as follows,

$$q_1(k_1(\tau), k_2(\tau)) = q_2(k_1(\tau), k_2(\tau)) \cdots\cdots\cdots\cdots (29)$$

$$= q_3(k_1(\tau), k_2(\tau)) \cdots\cdots\cdots\cdots (30)$$

$$= k(\tau) v_f \Big(1 - \frac{k(\tau)}{k_{jam}}\Big). \cdots\cdots\cdots (31)$$

Lastly, with following eqs. (32) to (35),

$$\nabla q_1(\boldsymbol{x})\Big|_{x_1=\hat{x_1}} = \left[\frac{v_f}{k_j}k - \frac{v_f}{2}, \frac{v_f}{k_j}k - \frac{v_f}{2}\right] \cdots (32)$$

$$\nabla^2 q_1(\boldsymbol{x})\Big|_{x_1=\hat{x_1}} = \begin{bmatrix} \frac{v_f}{2x_{jam}} & \frac{v_f}{2x_{jam}} \\ \frac{v_f}{2x_{jam}} & \frac{v_f}{2x_{jam}} \end{bmatrix} \cdots\cdots\cdots (33)$$

$$\nabla q_2(\boldsymbol{x})\Big|_{x_1=\hat{x_1}} = \left[2v_f \frac{k}{k_{jam}} - v_f, 0\right] \cdots\cdots\cdots (34)$$

$$\nabla^2 q_2(\boldsymbol{x})\Big|_{x_1=\hat{x_1}} = \begin{bmatrix} \frac{v_f}{2x_{jam}} & 0 \\ 0 & 0 \end{bmatrix}, \cdots\cdots\cdots\cdots (35)$$

the convexity condition of lemma 1 was satisfied, since

$$\nabla_1 q_1(x) \leq \nabla_1 q_2(x) \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots (36)$$

in pair with

$$\nabla_1^2 q_1(x) \leq \nabla_1^2 q_2(x). \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots (37)$$

In the same way,

$$\nabla_1 q_2(x) \leq \nabla_1 q_3(x), \;\; \nabla_1 q_1(x) \leq \nabla_1 q_3(x) \cdots (38)$$

are satisfied , paired together with

$$\nabla_1^2 q_2(x) \leq \nabla_1^2 q_3(x), \;\; \nabla_1^2 q_1(x) \leq \nabla_1^2 q_3(x). \cdots (39)$$

Therefore, the convexity of overall dynamics are confirmed.

Note that although z is the multiplication of q and u, the performance criteria (22) is a convex function. This is because u is the vector whose elements $u_i \in \{0, 1\}$ are binary variables, if $u_i = 1$, z_i remains as it stands now, otherwise the term z_i is dropped off from the performance criterion. And (22) is also a convex function, since $|x_1 - x_2|$ can be transformed to $(e_x^+ + e_x^-)$, minimizing $e_x^+ + e_x^-$ with the conditions of

$$e_x^+ \geq 0 \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots (40)$$

$$e_x^- \geq 0 \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots (41)$$

$$e_x^+ - e_x^- = x_1 - x_2 \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots (42)$$

where e_x^+ and e_x^- are equivalently

$$e_x^+ = \frac{(x_1 - x_2) + |x_1 - x_2|}{2} \cdots\cdots\cdots\cdots\cdots (43)$$

$$e_x^- = \frac{-(x_1 - x_2) + |x_1 - x_2|}{2}. \cdots\cdots\cdots\cdots\cdots (44)$$

Since all the constraints are described in the form of eq.(18), the problems (22) and (23) are included in the class of the convex programming problem.

The efficient method such as Penalty Method (PM) can be easily applied to the convex programming problem with performance scheme as follows,

$$\text{minimize} F(\boldsymbol{x}, r) = f(\boldsymbol{x}) + r\boldsymbol{P}(\boldsymbol{x}) \cdots\cdots\cdots\cdots (45)$$

$$\boldsymbol{P}(\boldsymbol{x}) \begin{cases} = 0, & \boldsymbol{x} \in X \\ > 0, & \boldsymbol{x} \notin X \end{cases} \cdots\cdots\cdots\cdots\cdots\cdots\cdots (46)$$

where $f(\boldsymbol{x})$ is the convex performance criterion of the original problem, $r(>0)$ is the cost coefficient which increases as iteration l increases, X is the convex set, and P is the continuous penalty function satisfying eq.(46).

If we can select the feasible initial solution, the optimal solution would be found in a short time. In this paper, the existence of solution is verified as follows.

Lemma 2: The range of $x_i(\kappa)$ where $1 \leq i \leq m$ is $0 \leq x_i(\kappa) \leq l_i k_{jam}$. If $x_i(\kappa+1)$ always exists within the range for all i in the case of $0 \leq x_i(\kappa) \leq l_i k_{jam}$ for all i, the feasible solution $\boldsymbol{x}(\kappa+1)$ can be found.

Proof : Consider the following equation:

$$l_i k_i(\kappa+1) - l_i k_i(\tau)$$
$$= -\boldsymbol{q}(k_{i-1}(\kappa), k_i(\kappa))T_s + \boldsymbol{q}(k_i(\kappa), k_{i+1}(\kappa))T_s \cdots\cdots\cdots\cdots\cdots\cdots (47)$$

It is obvious that x_i is within the range if and only if

$$l_i k_i(\tau) \geq -\boldsymbol{q}(k_i(\tau), k_{i+1}(\tau))T_s \cdots\cdots\cdots\cdots (48)$$

$$l_i k_{jam} - l_i k_i(\tau) \leq \boldsymbol{q}(k_{i-1}(\tau), k_i(\tau))T_s \cdots\cdots (49)$$

By substituting q of (48) to (6), following inequality is obtained from the both $k_i(\kappa) \geq k_{i+1}(\kappa)$ and $k_i(\kappa) < k_{i+1}(\kappa)$.

$$1 \geq \frac{v_f}{l_i}\left(1 - \frac{k_i}{k_{jam}}\right)T_s. \cdots\cdots\cdots\cdots\cdots\cdots (50)$$

Since $\frac{v_f T_s}{l_i} \ll 1$, (48) can be easily confirmed. In the similar way, the condition (49) can be easily confirmed.

5. Numerical Experiments

5.1 Numerical Environments In order to show the usefulness of our proposed method, we show, in this section, some results of the numerical experiments. We considered the traffic network of Fig.7, where the square network with 1000×1000 [m^2] consists of 16 intersections and 112 districts, all with 2 lanes bi-directionally. Four controllers are applied to find optimal traffic light for the overall network. It is known that the cellular automaton model (CA model) can simulate the real traffic flow with great granularity [(1)], although it takes too much computation time. Therefore, the CA model was constructed based on [(1)] to show feasibility of the proposed model. In our previous work [(9)], the model proposed in the paper was compared with the CA model to confirm the feasibility of the model. Each controller is assigned to the network with 500×500 [m^2]. We assume that from the outside of the network traffic flows of vehicles move into the network with random speeds, whereas the traffic flows inside the network, move from the network with the speed of infinity (no congestion arises and affects the traffic flow inside the network). The variables used in this paper are as follows; $\boldsymbol{x} \in \Re^{56}$, $\boldsymbol{q} \in \Re^{80}$, $\boldsymbol{\delta} \in \{0, 1\}^4$.

5.2 Traffic Flow Control System for Traffic Network We shows the results obtained by applying our proposed methods in Table 1, where H denotes

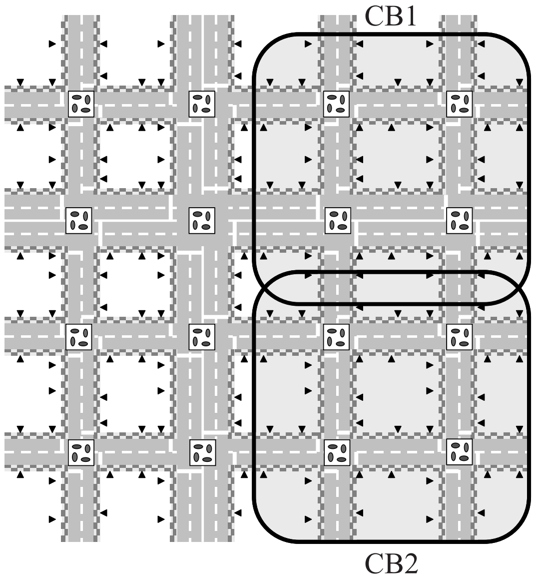

Fig. 7. Traffic Network

Table 1. Numerical Experimental Result WRT H

	No Control	$H=1$	$H=2$
A	2724	2884	2913
B	-	3.1	370.4
C	-	1.2	14.6

the length of the prediction horizon, 'No Control' implies that the traffic light is changed at every 30 seconds, and

A: Number of cars passing through the boundary of every two consecutive districts,

B: Average computation time,

C: Average number of the sub-problem generated.

From the results in Table 1, we find that although the MPC with longer prediction horizon enables more vehicles to pass through the traffic network, the difference between the cases of $H = 1$ and $H = 2$ is not so remarkable. This implies that the proposed method can be applied to find semi-optimal solution for the real traffic control system with a proper selection of prediction horizon length.

5.3 Comparison of computational amount In order to evaluate the computational amount of the proposed method, we compare in Table 2 computational times obtained by applying our proposed method and conventional method [9]. We used Althlon XP 2400+ Windows 2000 for this experiments. In Table 2, A implies the introduced number of δ, B implies the computation time, and C implies the number of cars passed during the corresponding sampling interval. Note that our proposed method finds better solution with a shorter time. This is because the proposed method does not approximate nonlinear dynamics [9] and solves non-linear programming problem, reformulating it to the convex programming problem. Furthermore, the proposed refining process enables to eliminate the introduction of auxiliary variables that without them, the logical relation between k_i, k_j, c_i, and q in (6) is well formulated and optimized.

Table 2. Comparison of the computational efforts

Length of H	Proposed Method			Method of [9]		
	A	B	C	A	B	C
1	4	0.02	616	244	14.98	616
2	8	1.34	724	488	265.18	718
3	12	129.20	869	732	2688.6	870

Table 3. Experimental Result in case of no arterial road

	No Control	$H=1$	$H=2$
A	5249	5660	5717
B	-	3.1	370.4
C	-	1.2	14.6

Table 4. Experimental Result in case of 2 arterial roads

	No Control	$H=1$	$H=2$
A	6060	6980	7185
B	-	3.6	250.4
C	-	1.3	10.4

5.4 TFCS for large-scale traffic network In this subsection, the effectiveness of our proposed method for large-scale traffic network control with the arterial roads is shown. If the traffic light controller is applied to the large-scale traffic network in a centralized manner, the computational amount would be fairly enormous. The proposed method, as in Fig.7, designates the control block which groups some traffic lights in order that the feasible solutions may be obtained during the sampling interval. Fig.7 illustrates that four control blocks (CB) constitute the entire traffic network where the sensory information at each boundary of CBs is shared for the control of both blocks. Note that two arterial roads are running north-south (second road from the left) and east-west (second road from the top), respectively. Table 3 and Table 4 shows the obtained solutions by applying the proposed method both in the case that there is no arterial roads and in the case that there are 2 arterial roads. In both numerical experiments, traffic densities at each road were set to exactly same. The results in both cases show that the proposed method has good solution in both cases. Note that our method has always better solutions than the cases of No Control.

6. Conclusions

In this paper, a new method for traffic light control based on hybrid dynamical system theory has been proposed. First of all, the synthetic modeling method for the traffic flow control system has been proposed where the information on geometrical traffic network was modeled by using Hybrid Petri Net, 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 light by means of its algebraic manipulability. Secondly, the shock wave model has been introduced in order to treat the discontinuity of the traffic flow. The developed non-linear dynamics was formulated based on the Mixed Integer NonLinear Programming problem, and yields global optimal solution coupled with convexity analysis. Lastly, the proposed Branch and Bound algorithm, which introduced the refining process, enables to minimize the introduced number of the auxiliary variables, whereas the conventional MINLP problems are known as computationally expensive. Some numerical experiments have been carried out, and have shown the usefulness of the proposed design framework. Our future works include the analytical consideration of stochastically changing traffic network dynamics.

Acknowledgment

This work was supported by High-Tech Research Center,Project for Private Universities: matching fund subsidy from MEXT(Ministry of Education, Culture, Sports, Science and Technology), 2002-2006.

(Manuscript received July 20, 2005,

revised Feb. 20, 2006)

References

(1) Y. Kato: "Traffic Flow Simulation by Cellular Automaton Method", *JSIE*, Vol.15, No.2, pp.242–250 (2000)

(2) K. Nagel and M. Schreckenberg: "A Cellular Automaton Model for Freeway Traffic", J. de Physique I France, pp.2–2221 (1992)

(3) P.P. Chaudahuri and others: "Additive Cellular Automata-Theory and Applications", IEEE Comput. Soc. Press (1997)

(4) H. Hisao: "Considering the traffic flow as the pulverulent body", *Parity*, Vol.13, No.5, pp.13–22 (1998) (in Japanese)

(5) M.J. Lighthill and G.B. Whitham: "On kinematic waves II. A theory of traffic flow on long crowded roads", Proc. R. Soc. London Ser. A, Vol.229, p.281 (1955)

(6) H.J. Payne: "Models of freeway traffic and control in Mathematical Models of Public Systems", Simulation Council Proceedings Series, *LaJolla*, California, Vol.1, No.1, pp.51–61 (1971)

(7) I. Prigogine and R. Herman: "Kinetic Theory of Vehicular Traffic", Elsevier, New York (1971)

(8) M. Cremer and J. Ludwig: "A fast simulation model for traffic flow on the basis of Boolean operations", Mathematics and Computers in Simulation, Vol.28, pp.297–303 (1986)

(9) T. Kato, Y.W. Kim, and S. Okuma: "Model Predictive Control of Traffic Flow Based on Hybrid System Modeling", ICCAS, pp.368–373 (2005)

(10) F. Balduzzi, A. Giua, and G. Menga: "First-order hybrid Petri nets: a model for optimization and control.", *IEEE Trans. on Robotics and Automation*, Vol.16, No.4, pp.382–399 (2000)

(11) M. Aoyama, N. Uchihira, and K. Hiraishi: "Petri Net theory and its applicatoins", ISCIE (1995) (in Japanese)

(12) R. Haberman: Mathematical Models, Prentice-Hall (1977)

(13) A. Bemporad and M. Morari: "Control of systems integrating logic, dynamics, and constraints", *Tech. Report AUT98-04, ETH, Automatica*, Special issue on hybrid systems, Vol.35, No.3, pp.407–427 (1999)

(14) M. Morari and E. Zafirou: Robust Process Control, Prentice Hall (1989)

(15) E.F. Camacho and C. Bordons: Model predictive control in the process industry, Springer-Verlag (1995)

Tatsuya Kato (Student Member) Tatsuya Kato was born in Aichi, JAPAN, in 1979. He received the B.E degrees in Electrical Engineering from Aichi Tnstitute of Technology, JAPAN in 2002 and the M.E degree in Electrical Engineering from Nagoya University, Japan in 2004. Now, he is candidate for the Ph.D. degree in Department of Mechanical Science and Engineering of Nagoya University.

YoungWoo Kim (Non-member) YoungWoo Kim was born in TaeGu city, Korea, in 1973. He received the B.E. degree in Electrical Engineering from YeungNam University, Korea in 1997, and the M.S and Ph.D. degrees in Electrical Engineering from Nagoya Univerity, Japan in 2000 and 2004, respectively. Since 2004, he has been working for Space Robotic Research Center of Toyota Technological Institute, Nagoya, Japan. He was the recipient of the ICCAS'2005 Best Paper Award. His recent research interests are in the areas of hybrid and non-linear dynamical systems.

Shigeru Okuma (Member) Shigeru Okuma was born in Gifu, JAPAN, in 1948. He received the B.E., M.E. and Ph.D. degrees in Electrical Engineering from Nagoya University, JAPAN in 1970, 1972 and 1978, respectively. He also received the M.E. degree in systems engineering from Case Western Reserve University, Cleveland, OH, in 1974. Since 1990, he has been a Professor in the Department of Electrical Engineering and Computer Science, Nagoya University. His research interests are power electronics, robotics and evolutionary soft computing. Dr. Okuma was the recipient of the IEEE IECON'92 Best Paper Award, in addition to paper awards from the Japan Society for Precision Engineering and Institute of Electrical Engineering of JAPAN.

Tatsuo Narikiyo (Member) Tatsuo Narikiyo was born in 1952 in Fukuoka. He received a B.E. degree in applied physics from Nagoya University in 1978 and a Dr.Eng. degree in control engineering from the same university in 1984. He had been a Research scientist at the Government Industrial Research Institute, Kyushu from 1983 to 1990. From April 1990 he was an Associate Professor in the department of Information and Control Engineering at Toyota Technological Institute. Since April 1998 he has been a Professor at the same Institute. His main research interests are in the control system design for linear and nonlinear mechanical systems.