STOCHASTIC CONGESTION CONTROL MODEL FOR VIDEO TELECONFERENCE SERVICE TRFFIC (VTST) SYSTEMS

[1]K.SENBAGAM, [2]Dr.C.V.SESHAIAH
Assistant Professor, Department of Mathematics,Sri Ramakrishna Engineering College, Coimbatore
Professor and HOD of Science and Humanities,Sri Ramakrishna Engineering College, Coimbatore
visnusen@gmail.com

ABSTRACT:-A congestion control model of a network of signalized intersections is proposed based on a discrete – time, steady state Markov decision process. The model incorporates probabilistic forecasts of individual video actuations at downstream inductance loop detectors that are derived from a macroscopic link transfer function. The model is tested both on a typical isolated Roberston's traffic platoon intersection and a simple network comprised of five four – legged signalized intersections, and compared to full-actuated control. Analyses of simulation results using this approach show significant improvement over traditional full-actuated control, especially for the case of high volume, but not saturated, traffic demand.

Keywords: Congestion control, Markov control process, traffic flow, Video, simulation.

1. Background

The existence of real-time congestion traffic control systems that react to actual traffic conditions on-line, the most notable among these being the well-known OPAC algorithm(Gartner,1983), and RHODES, a real-time traffic –congestion control system that uses a traffic flow arrivals algorithm(HEAD,1995) based on detector information to predict future traffic volume.

In general, two issues must be addressed to achieve real-time congestion traffic control: (1) development of a mathematical model for the control of the stochastic, highly nonlinear VTST system, and (2) design of appropriate control law such that the behavior of the system meets certain performance indices (e.g., minimum queue length, minimum delay time, etc.). Mathematical models used for the representation of traffic phenomena on signalized surface street network can be classified into the following three generalized categories: (1) store-and forward models (Hakimi, 2009; Singh and Tamura, 2006; D's Ans and Gazis, 2008), (2) dispersion-and-store model Cremer and Schoof, 2007; Chang etal., 1994), and (3) kinematic wave models(Stephanedes and Chang, 1993 , Lo, 2001).

There are two fundamental approaches for on-line optimization: binary choice logic and the sequential approach. In the binary choice logic approach, time is divided into successive small intervals, and a binary decision is made either to extend the current signal phase by one interval, or to terminate it. Examples of this approach include Miller's algorithm, traffic optimization logic(TOL), modernized optimized service actuation strategy (OSAS), stepwise adjustment of VTST timing (SAVTSTT), etc. (Lin,2009; Lin and Vijayakumar, 2010). The drawback of this approach is that it only considers a very short future time interval (usually 3-6 s) for the decision, and thus cannot guarantee the overall optimization of the VTST operation. In the sequential; approach, the length of a decision-making stage is relatively longer (from 50 to100 s) to more closely approach the long-term optimal control. In OPAC, developed explicitly for real-time traffic control, the alternative disadvantages of the binary and sequential approaches are mitigated by incorporating a rolling-horizon approach; however, its application formally is limited to isolated intersections. Artificial neural networks (ANN) also have been applied to finding the solution for traffic control problems (Nakatsuji and Kaku, 2007) through an assumed mapping between the control variables (e.g., the split) and the objective function (e.g., the queue length); the neural network is trained off-line, using the nonlinear mapping ability of ANN, to realize this relationship. Then the signal optimization is performed on-line, using the self-organization property of an ANN. The training algorithm is a stepwise method(combination of a Cauchy machine and the "back-propagation" algorithm). However, this approach is valid only when the traffic system is in steady state.

2. Introduction

At a signalized intersection, VTST operates in one of three different control modes: In pre-times control, semi-actuated control and full-actuated control (Wilshire etal., 2006). In pre-timed control, all of the control parameters are fixed and preset off-line. Off-line techniques (e.g., the various versions of the TRANSYT (Robertson 2002) family of software packages) are useful in generating the parameters for fixed timing plans for conventional pre-timed urban traffic control systems based on the deterministic traffic conditions during different time periods of the day (e.g., peak hours, off-peak hours). In actuated (both semi- and full-) control, the control service is adjusted in accordance with a "closed-loop, on-line" control strategy based on real-time traffic demand measures obtaind from detectors; while the controllers themselves respond to the fluctuations of the traffic flows in the network, the base parameters do not. Alternatively, a class of control algorithms that includes SCOOT (Split, Cycle And Optimization Technique) (Hunt(i)
etal., 2008; Robertson and Bretherton, 2002) and CCTS (Coordinated Congestion Traffic System)(ii)
(Lowrie, 2006) are generally considered to be "On-line" algorithms, in which the control strategy is to "match" the current traffic conditions obtained from detectors to the "best" pre-calculated off-line timing plan. (iii)

Although most existing congestion signal control strategies incorporate an implicit recognition that traffic conditions are time variant(iv)
due to Stochastic processes, they generally adopt explicitly deterministic control models. Additionally, most employ heuristic control strategies without an embedded traffic flow model. Alternatively, the random nature of the traffic signal lends itself more directly to a stochastic control approach. In the work reported here, a stochastic traffic signal control scheme, based on Markovian decision control, is introduced. The objective is to develop a real-time congestion control strategy that explicitly incorporates the random nature of the traffic system in the control. A Markov control model is first developed; then the signal control problem is formulated as a decision-making problem for the Markov model. This approach is tested both on a typical isolated traffic intersection and a simple network comprised of five four-legged signalized intersection improvement over traditional full-actuated control, especially for the case of high volume traffic demand.

3. Markov Service Traffic Control Model

A stochastic process x(t) is called Markov (Papoulis, 2006) if its future probabilities are determined by its most recent values; i.e., if for every n and $t_1 < t_2 < ..... < t_n$

$$P(x(t_n) \leq x_n / x(t) \leq x_{n-1},\ \forall\ t) = P(x(t_n) \leq x(t_{n-1})).$$

The congestion control algorithm proposed is based on a discrete, stationary, Markov control model defined on (**X,** A **,P,R**), where

X, a Brownian space is the state space and every element in the space $x \in$ **X** is called a state;
A, also a Brownian space, is defined as the set of all possible controls(or alternatives). Each state $x \in$ **X** is associated with a non-empty measurable subset A (x) of A whose elements are the admissible alternatives when the system is in state x;
P , a probability measure space in which an element p_{ij}^{k} denotes the transition probability from state i to state j under alternative action k; and
R, a measurable function, also called a one-step reward.

Selection of a particular alternative results to an immediate reward and a transition probability to the next state . The total expected discounted reward over an infinite period of time is defined as

$$D \overset{\Delta}{=} E\left[\sum_{t=0}^{\infty}\beta^{t} r(x_t, h_t)\right] \qquad (1)$$

Where $r(\cdot)$ is the one-step transition reward, $\beta\ t$ ($0 \le \beta\ t \le 1$) is the discount factor, and h is the policy. The optimal reward d^* , or the supermum (least upper bound) of D , is defined as

$$d^*(x,h^*) = \sup_{h\in A}\{D(x,h)\} \qquad (2)$$

It can be obtained by solving a functional equation (also called the dynamic programming equation, or DPE):

$$d^* = Td^* \qquad (3)$$

Where T is a contraction operator and

$$Td(x) = \max_{h\in A}\left[q(x,h) + \beta\sum_{j=1}^{N} d(k)p_{ij}^{h}\right] \qquad (4)$$

The expected one-step transition reward $q(x,h)$ is defined as

$$q(x,h) = \sum_{j=1}^{N} r_{ij}^{h} p_{ij}^{h} \qquad (5)$$

The unique solution of the above DPE can be calculated iteratively by the successive approximation method (Hernandez-Lerma,2008)

$$d_n(x) = \max_{h\in A}\left[q(x,h) + \beta\sum_{j=1}^{N} d_{n-1}(k)p_{ij}^{h}\right] \qquad (6)$$

Therefore, for a specific control problem, once the transition matrix and the reward matrix are defined, then by maximizing the total expected reward, a policy for choosing an alternative for each state can be obtained. This represents the optimal strategy that should be followed.

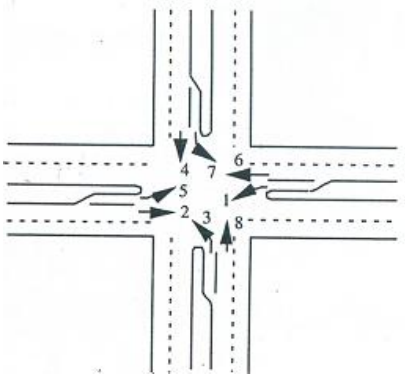
4. Traffic Flow

Consider the typical four-legged isolated traffic intersection shown in Fig.1, where the various possible traffic movements.

The state equation for the continuous traffic flow process associated with any movement j that is sampled every Δt seconds, where time is indexed with the integer k, can be expressed by the current queue $q^j(k)$:

$$q^j(k) = q^j(k-1) + \Delta q^j(k), \text{ j = 1, 2, …. , 8,} \qquad (7)$$

Fig.1. A typical video traffic intersection.



where $\Delta q^j(k) = q_{in}^j(k) - q_{out}^j(k)$ is the difference between the input $q_{in}^j(k)$ and the output $q_{out}^j(k)$ during time interval $(k-1,k)$, and $q^j(k-1)$ is the queue at previous time instant $k-1$. For a typical four-legged traffic intersection with eight movements, the current queue $q(k)$ can be further defined by the vector

$$\vec{q}(k) = (q^j(k))^T = [q^1(k), q^2(k), \ldots\ldots, q^8(k)]^T \qquad (8)$$

where prime (T) is used to denote transpose. The input $\vec{q}_{in}(k)$ and output $\vec{q}_{out}(k)$ of the intersection (i.e., number of video conferences entering/leaving the intersection) can also be similarly defined as vectors of like dimension:

$$\vec{q}_{in}(k) = (q_{in}^j(k)), \ \vec{q}_{out}(k) = (q_{out}^j(k)) \qquad (9)$$

The output $\vec{q}_{out}(k)$ can further be expressed as a function of the current control of the intersection, $c(k)$, and the current queue, $\vec{q}(k)$:

$$\vec{q}_{out}(k) = \vec{f}_{out}(c(k), \vec{q}(k)) \qquad (10)$$

Where $\vec{q}_{out}(k)$ is also a vector of the same dimension, i.e.,

$$\vec{q}_{out}(k) = (f_{out}^{j}(k)) \qquad (11)$$

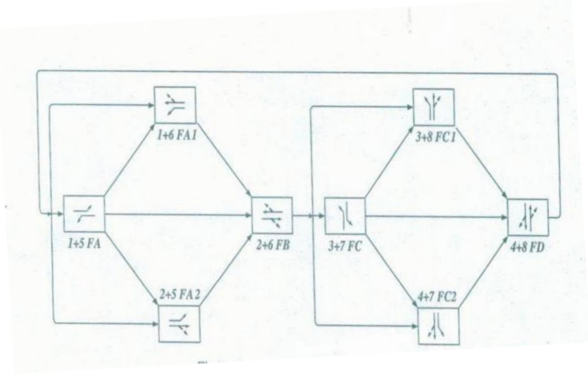
and where the elements $f_{out}^{j}(k)$ are determined by

$$f_{out}^{j}(k) = \begin{cases} \min\left[q^{j}(k);\dfrac{\Delta t}{h_{\min}}\right], & c^{j}(k)=0 \\ 0, & c^{j}(k)=1 \end{cases} \qquad (12)$$

In which $h_{\min}$ is the minimum head way and $c^{j}(k)$ is a dichotomous variable indicating the control signal for the j^{th} movement: $c^{j}(k)=0$ denotes that the j^{th} movement has the green signal and $\mathrm{c}^{\mathrm{j}}(k)=1$ indicates a red signal.

Under standard eight-phase dual-ring control(Fig.2), the barrier divides the eight phases into two interlocked groups(rings):east/west and north/south: in each ring, four movements(two through movements and their corresponding left-turn movements) must be served if there is demand. Although there are 2.4!=48 different phase sequences available, depending on the traffic demand, the ring and barrier rules restrict the maximum number of phase transition in a single cycle to six- a maximum of three distinct phase combinations on each side of the barrier. Using this information, the phase sequencing constraints on choice of the current control depends, at most, on three previous control signals:

Fig.2. Eight-phase dual-ring service control



$$c(k) = f_u(q(k),\tau,c(k-\tau_1),c(k-\tau_2),c(k-\tau_3)) \qquad (13)$$

Where τ_1 is the time duration of the most recent previous phase, τ_2 is the time duration of the next-to-last phase, and so on. In addition to the sequencing constraints, the duration of the current signal, τ must be bounded between some minimum (e.g. minimum green, minimum green extension) and (maximum green) time period:

$$\tau_{\min} \leq \tau \leq \tau_{\max} \qquad (14)$$

This schema easily can be generalized to traffic network with multiple intersections. In a traffic network with n intersection, the order of the dynamic equation is increased to n x 8(assuming that there are eight traffic movements in each intersection). However, any complicated traffic network can be decomposed into a group of small "elementary networks" as shown in Fig.3, consisting of five intersections. In this manner the study of the entire traffic network can be reduced to the analysis of these elementary networks and the inter-connections between them.

The complete traffic dynamic model for the network shown in Fig.3

$$\vec{c}(k) = \left[\vec{c}_{1}(k),\vec{c}_{2}(k),\ldots,\vec{c}_{5}(k)\right]^{T}$$

$$\vec{f}_{c}(k) = \left[\vec{f}_{c_1}(k),\vec{f}_{c_2}(k),\ldots,\vec{f}_{c_5}(k)\right]^{T}$$

$$\vec{q}_{out}(k) = \left[\vec{q}_{out\,1}(k), \vec{q}_{out\,2}(k), \ldots, \vec{q}_{out\,5}(k)\right]^T$$

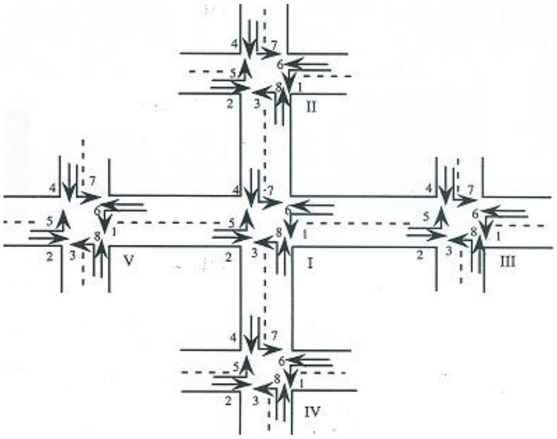
$$\vec{q}_{in}(k) = \left[\vec{q}_{in\,1}(k), \vec{q}_{in\,2}(k), \ldots, \vec{q}_{in\,5}(k)\right]^T \qquad (15)$$

$$\vec{q}\,(k) = \left[\vec{q}_1(k), \vec{q}_2(k), \ldots, \vec{q}_5(k)\right]^T$$

$$\vec{f}_{out}(k) = \left[\vec{f}_{out_1}(k), \vec{f}_{out_2}(k), \ldots, \vec{f}_{out_5}(k)\right]^T$$

$$\vec{q}(k) = \vec{q}(k-1) + \Delta\vec{q}(k)$$

Fig.3. A typical elementary traffic network with five intersections.



where

$$q_{outi}^{j}(k) = f_{out}\left(c_i^j(k), q_i^j(k)\right) \qquad (16)$$

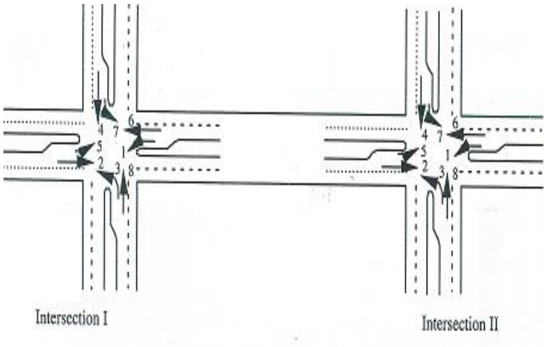
$$f_{out}^{j}(k) = \begin{cases} \min\left[q_i^j(k); \dfrac{\Delta t}{h_{\min}}, c_i^j(k) = 0,\right] \\ 0, \qquad\qquad c_i^j(k) = 1 \end{cases}$$

i = 1, 2, …,5; j = 1, 2, …,8 (17)

In Eq. (15), the subscripts to the various vector quantities refer to the particular intersection, and the vector quantities themselves are as previously defined.

Unlike the case of an isolated intersection, the interactions between intersections must be included in the traffic model for this case. For example, consider the simple case of the two adjacent intersections shown in Fig. 4.

Fig.4. A traffic network with two intersections



The eight traffic movements associated with each intersection can be classified into two different types:

1. External movement. The arrival videos come from/go to a “dummy node” outside the network (these video can be considered as the “input/output” of this network); and
2. Internal movement. The arrival videos comes/goes to a neighboring node inside the network(these video can be considered as the “interconnection” of this network).

For example, movements 1 and 6 are internal movements of intersection I, which receive the outputs from intersection II, movements 3 and 6. All of the other movements of intersection one are external movements. Similarly, all of the movements of intersection II are external movements, with the exception of movements 2 and 5, which receive the output from the movements 2 and 7 of intersection I. then, for intersection, the internal movements are defined by

$$q_{in1}^{1}(k) = f_{in}\left(\beta_{II,I}^{3,1}(k-T_{II,I}^{3}), q_{outII}^{3}(k-T_{II,I}^{3}), \beta_{II,I}^{6,1}(k-T_{II,I}^{6}), q_{outII}^{6}(k-T_{II,I}^{6})\right)$$

$$q_{in1}^{6}(k) = f_{in}\left(\beta_{II,I}^{3,6}(k-T_{II,I}^{3}), q_{outII}^{3}(k-T_{II,I}^{3}), \beta_{II,I}^{6,6}(k-T_{II,I}^{6}), q_{outII}^{6}(k-T_{II,I}^{6})\right) \qquad (18)$$

where $\beta_{i_1,i_2}^{j_1,j_2}$ is defined as the video moving fraction from intersection i_1, movement j_1 to intersection i_2, movement j_2 and where $T_{i_1,i_2}^{j_1}$ represents the time for the first video conference in the platoon of video in movement j_1 of intersection i_1 to reach intersection i_2.

The time- dependent moving factors can be represented by the moving fraction matrix, $\vec{\beta}(k)$, whose elements indicate the percentage of video moving from a certain movement at the upstream intersection to a specific movement at the downstream intersection. For the case of two intersections shown in Fig. 4, $\vec{\beta}(k)$ can be written as a 16x16 matrix:

$$\vec{\mathrm{B}}(k) = \begin{bmatrix} \vec{0} & \left[\beta_{I,II}^{j_1,j_2}(k)\right] \\ \left[\beta_{II,I}^{j_1,j_2}(k)\right] & \vec{0} \end{bmatrix}$$

$$= \begin{bmatrix} \vec{0} & \vec{\mathrm{B}}_{I,II}(k) \\ \vec{\mathrm{B}}_{II,I}(k) & \vec{0} \end{bmatrix} \qquad (19)$$

where

$$\vec{\mathrm{B}}_{I,II}(k) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \beta_{I,II}^{2,2}(k) & 0 & 0 & \beta_{I,II}^{2,5}(k) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \cdots & & & \cdots & & & & \cdots \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \beta_{I,II}^{7,2}(k) & 0 & 0 & \beta_{I,II}^{7,5}(k) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\vec{\mathrm{B}}_{II,I}(k) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \beta_{I,II}^{3,1}(k) & 0 & 0 & 0 & 0 & \beta_{II,I}^{3,6}(k) & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \beta_{II,I}^{6,1}(k) & 0 & 0 & 0 & 0 & \beta_{II,I}^{6,6}(k) & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Using this general expression for $\vec{\beta}(k)$

$$q_{in\ i}^{j}(k) = f_{in}\left(\beta_{m,i}^{r,j}(k-T_{m,i}^{r}), q_{out\ m}^{r}(k-T_{m,i}^{r})\right), \forall r \in M_m^i, m \in I_i$$

(20)

where I_i is the set of all neighboring intersections with direct approaches to intersection I, and M_m^i is the set of all movements of intersection m that contribute to the internal movements of intersection i.

Practical application of Eq. (20) relies on the ability to predict both the time-dependent moving fractions, $\vec{\beta}(k)$, and the platoon travel times from neighboring intersections to the target intersection, $T_{i_1,i_2}^{j_1}$. The estimation of moving fractions from count data has been the subject of numerous investigations; see, e.g., the review provided by Maher(1984) for a proposed of models that require counts for only one cycle but needed prior turning proportion estimation. Davis and Lan (1995) gave a method that estimates intersection turning movement proportions from less-than-complete sets of traffic counts. Chang and Tao (1997) propose a time-dependent turning estimation that incorporates service timing parameters on the distribution of intersection flow. Mirchandani etal. (2001) gave four closed-form estimation methods:(i) maximum entropy(ME), (ii) generalized least squared(GLS), (iii) least-squared error(LS), and (iv) least-squared error/ generalized least squared error(LS/GLS).

In its basic model representation:

$$Q_1(t_0+T) = F \cdot Q_2(t_0) + (1-F) \cdot Q_1(t_0+T-1) \qquad (21)$$

where

$$F = \frac{1}{1+\alpha\beta T_{avg}}$$

and where Q_1 , Q_2 are traffic volumes at the downstream and upstream intersections(measured in videos/h), respectively; α and β are called platoon dispersion parameters; t_0 is the initial time when the platoon leaves upstream intersection; T_{avg} is the average travel time, and T is the minimum travel time between the two intersections.

$$T = \beta T_{avg} \qquad (22)$$

Substituting Robertson’s platoon dispersion formula into Eq.(20) leads to

$$q_{in\ i}^{j}(k) = \sum_{\substack{r \in M_m^i \\ m \in I_i}} F \cdot \beta_{m,i}^{r,j}(k-T_{m,i}^{r}) \cdot q_{out\ m}^{r}(k-T_{m,i}^{r}) + (1-F) \cdot q_{in\ i}^{j}(k-1) \qquad (23)$$

with the current control vector defined by

$$c_i(k) = f_c\left(q_i(k), T_i, c_i(k-\tau_1), c_i(k-\tau_2), c_i(k-\tau_3)\right) \qquad (24)$$

where $\tau_{\mathrm{min}i} \leq \tau_i \leq \tau_{\mathrm{max}i}$, $\beta_{m,i}^{r,j}$ ❨can be derived from counts from upstream stopline detectors according to existing procedures discussed previously, and where the $T_{m,i}^{r}$ are determined by Eq. (22) from parameter specification and average travel speed.

5. Markov congestion control model for VTST system

The state variable in the traffic dynamics equation developed above is queue length. Although the state of the Markov control model can be defined as the number of video conference in the intersection, this approach results in an excessively large number of states, even for a single intersection. To address this problem, the state of the Markov control model is instead defined by introduction of a binary threshold value (number of video conferences) indicating whether of not the current queue for a particular movement is sufficiently large to be "congested", i.e., if the queue length of a specific movement is greater than its threshold value, then the movement is in the "congested mode"; otherwise it is in the "non-congested mode". These binary modes (congestion/non-congestion) are defined as the two states in the state space **X**.

Since the state space is discrete, the probability measures P is a discrete transition law, and the probability matrix $\vec{P}$ is time-varying traffic flow. At time step k, $\vec{P}$ is a function of $\vec{q}\;k$, $\overset{\wedge}{\Delta q}\;k+1$ and $\vec{c}\;k$:

$$\vec{P}\;k = \vec{f}_p\left[q\;k\;, \vec{q}_{in}\;k+1\;, c\;k\;\right] \qquad (25)$$

where $\vec{q}\;k$ is the current queue, $\overset{\wedge}{\Delta q}\;k+1$ is the estimated number of arrivals in the next time interval, and $\vec{c}\;k$ is the control signal. Assuming that at time step, the current queue length of a specific movement I is denoted by q_0; and q_g video can pass through the intersection if the traffic signal for this movement is green; then the transition probability from any current state (either congested or non-congested) to the non-congested state under control signal c can be written as

$$P_{S_i \to N_i}^{c_i} = p\left(\hat{q}{}_{in}^{i} + q_0^i - \delta\;\;c_i\;\;\cdot q_g^i \leq q_{threshold}^i\right) \quad (26)$$

and, to the congested state, as

$$p_{S_i \to C_i}^{c_i} = 1 - p_{S_i \to N_i}^{c_i} \qquad (27)$$

Where $\delta\;\;c_i\;\; = \begin{cases} 1, & \text{when } \mathrm{c_i} = G_i \\ 0, & \text{otherwise.} \end{cases}$ (28)

In the above, $q_{threshold}^i$ is the threshold which defines the congested/non-congested state; S_i is the current state (N_i for non-congested state and C_i for congested state); c_i is the control signal (G_i for green signal and R_i for red signal). Two special cases are noted in that:

$$p_{C_i \to C_i}^{R_i} \equiv 1 \text{ and } p_{C_i \to N_i}^{R_i} \equiv 0 \qquad (29)$$

As mentioned previously, for a typical traffic intersection with eight independent movements, the total number of states is 2^8 = 256. The transition probability for each movement is also independent; therefore, the overall transition probability for an intersection is

$$p_{State_j \to State_r}^{\vec{c}} = \prod_{i=1}^{8} p_{S_i \to S_i}^{c_i} \qquad (30)$$

where

j,r=1,2,…,256; and $\vec{c}\;\;k\;\;$ =[u$_1$,u$_2$,…,u$_8$]^T. The reward matrix $\vec{R}$ has the same dimension and a definition similar to that of the probability matrix. The control objective is to maintain the non-congested condition or, if already congested, to transit to a non-congested state. The latter yields a greater reward than the former and the transition from a non-congested state to congest carries a greater penalty than remaining in a congested state. Since the congested/non-congested state is defined in terms of queue length, the reward matrix is a function of the current queue, the threshold, and the control signal:

$$\vec{R}\;k\;=\vec{f}_r\left[\vec{q}_0\;\;k\;\;,\vec{q}_{threshold}\;\;k\;\;,\vec{c}\;\;k\;\;\right] \quad (31)$$

For example, if the objective is to minimize the queue length, then the reward for each possible case can be chosen as the following:

$$r^{G_i}_{N_i\to N_i}=q^i_0+M_1$$

$$r^{R_i}_{N_i\to N_i}=q^i_0+M_2$$

$$r^{G_i}_{N_i\to C_i}=q^i_0+M_3$$

$$r^{R_i}_{N_i\to C_i}=M_4$$

$$r^{G_i}_{C_i\to N_i}=q^i_0+M_5$$

$$r^{R_i}_{C_i\to N_i}=N.A.$$

$$r^{G_i}_{C_i\to C_i}=q^i_0+M_6$$

$$r^{R_i}_{N_i\to N_i}=M_7$$

Where M_i , i= 1,2,…,7, are constants which can be specified for a specific control problem.

Similar to the probability matrix, the overall reward for an intersection with eight independent movements is

$$r^{\vec{c}}_{State_j\to State_r}=\prod_{i=1}^{8} r^{c_i}_{S_i\to S_i}\,,$$

where j,r =1,2,…,256. (32)

The signal phase are the different alternative for each state; for a typical isolated traffic intersection with eight independent movements under eight-phase dual-ring signal, the signal control problem takes the form of a 256-state Markov process with eight alternative for each state. The optimal policy is then obtained by selecting the alternative for each state that maximizes the total expected reward. As has been demonstrated above, this optimal solution is unique and can be calculated iteratively by the successive approximation method.

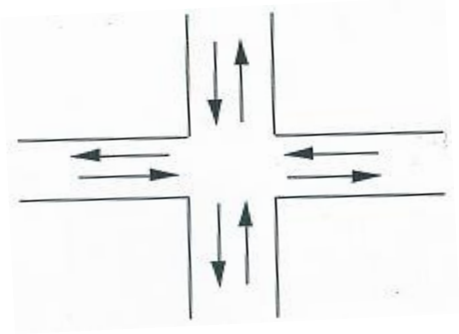
The proposed Markov control model can be illustrated by the simplified example of the two-phase isolated intersection shown in Fig. 5, in which traffic flows along two directions, i.e., north/south (denoted by 1) and east/west (denoted by 2). Thus, there are four possible state, i.e., $N_1N_2, N_1C_2, C_1N_2, C_1C_2$. Fig. 6 shows the schematics of this Markov Chain. To simplify the example, amber displays and all red signals (R_1R_2) are ignored; G_1G_2 is prohibited for obvious reasons. Under these conditions, there are two alternatives (signal phases) in each state, i.e., G_1R_2 and R_1G_2 . With the usual assumption of Poisson arrivals, the various transition probabilities can be calculated directly. For example, the transition probabilities from the non-congested states are

$$p^G_{N\to C}=\sum_{n=1}^{q_{threshold}-q+q_g}\frac{\lambda\Delta t\;\;^n\,e^{-\lambda\Delta t}}{n!}$$

$$p^G_{N\to N}=\sum_{n=1}^{q_{threshold}-q}\frac{\lambda\Delta t\;\;^n\,e^{-\lambda\Delta t}}{n!} \quad (33)$$

$$p^G_{N\to C}=1-p^G_{N\to N}\;,\;p^R_{N\to C}=1-p^R_{N\to N}\,, \quad (34)$$

Fig.5. An isolated intersection with through movements only



where n is a positive integer (n = 1,2,…); λ is the average video arrival rate and Δt is the time interval (i.e., duration of each counting period).

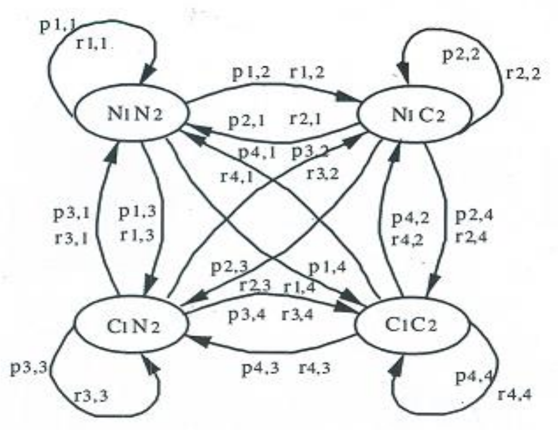
The corresponding state probabilities are

$$p^{G_1R_2}_{N_1N_2\to N_1N_2}=p^{G_1}_{N_1\to N_1}\cdot p^{R_2}_{N_2\to N_2}\,,$$

$$p^{G_1R_2}_{N_1N_2\to N_1C_2}=p^{G_1}_{N_1\to N_1}\cdot p^{R_2}_{N_2\to C_2}\,,$$

$$\vdots \quad (35)$$

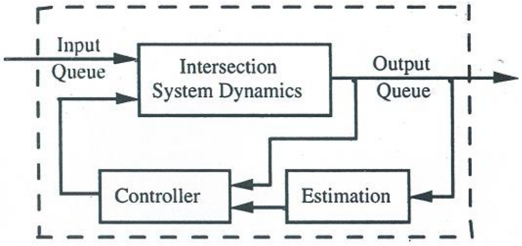
Fig.6. The Markov Chain for the example



Since, for this example, there are four states with two alternatives for each state, the elements above form an 8x4 transition probability matrix, as shown in Table 1. Elements of the reward matrix can be calculated in similar fashion.

A general block diagram of control using this scheme at an isolated signalized intersection is illustrated in Fig.7. Based on the current and the estimated traffic flow, the controller generates a traffic control signal to control the traffic system for the next time interval.

Fig. 7. Traffic control at signalized system



In the application of this procedure to real-time congestion control for a traffic system, the time-varying probability matrix $\vec{P}$ and the reward matrix $\vec{R}$ are calculated and updated every Δt seconds; a decision is then made regarding the choice of the control signal for the next time interval based on the current measurement from the detector, as well as the estimation. Once the optimal policy is found, it is implemented for one time step(i.e., Δt seconds). At the next time interval, both the probability matrix and reward matrix are updated and the whole decision making process is repeated.

Fig.8. Traffic control for two intersections

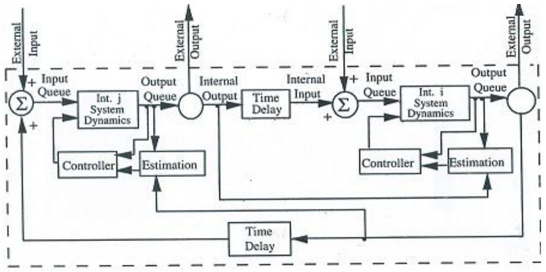


Table 1. The state probability matrix for the example

State		N_1N_2	N_1C_2	C_1N_2	C_1C_2
N_1N_2	G_1R_2	$p^{G_1R_2}_{N_1N_2\to N_1N_2}$	$p^{G_1R_2}_{N_1N_2\to N_1C_2}$		$p^{G_1R_2}_{N_1N_2\to C_1C_2}$
	R_1G_2	$p^{R_1G_2}_{N_1N_2\to N_1N_2}$			
N_1C_2	G_1R_2		$\ddots$		
	R_1G_2		$\ddots$		
C_1N_2	G_1R_2			$\ddots$	
	R_1G_2			$\ddots$	
C_1C_2	G_1R_2				$p^{G_1R_2}_{C_1C_2\to C_1C_2}$
	R_1G_2				$p^{R_1G_2}_{C_1C_2\to C_1C_2}$

To enforce the phase constraints, a step-by-step decision-making procedure (also termed a “decision tree”) is employed. For example, a decision is made first to determine which ring will be served by the Markovian decision algorithm. After this is determined, the second decision is to choose one of the four alternatives from the first decision, again using the Markovian decision algorithm. The next phase is either fixed or can be chosen from the two phases left, depending upon the second decision. At the last decision step for this ring, there is either no phase or just one fixed phase left. This procedure not only guarantees the phase constraints but also dramatically reduces computation time.

Application of the decision control to the signal control of a network of multiple intersections proceeds along a similar manner; a block diagram for the control system of two traffic intersections is shown in Fig.8. In such cases, the control signal of the two neighboring intersections do not interact until some minimum travel time, at which time the control is modeled through the probability estimation of internal movement arrivals at the downstream intersections. That is, assuming that the minimum travel time between two intersections is longer than the minimum green extension time, the control signals of the two intersections do not interact due to the random travel time delay between them. After the minimum travel time, the control at one intersection does affect intersections downstream; this effect is modeled in the probability estimation at the downstream intersections. As a result, adjacent intersections can be “isolated” and the respective control actions can be separately.

6. Results on application of Markov congestion model

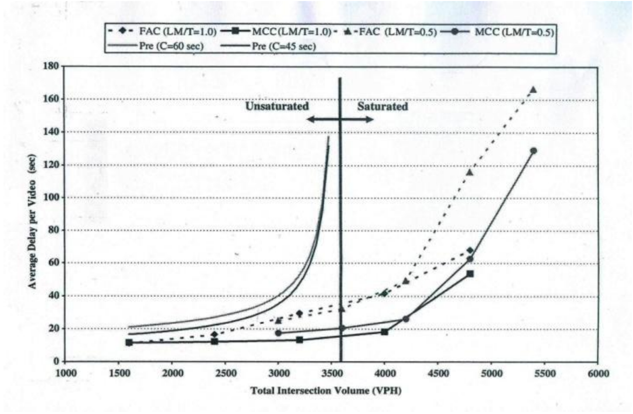
In this section, the control model is tested by simulation on both an isolated traffic intersection and a typical traffic network with five interconnected intersections to evaluate its performance with respect to conventional full-actuated control. Specifically, a series of computer simulations are performed, under various different video arrival rates, and the means and variances of the respective performance measures of the conventional and proposed congestion control algorithm are analyzed. The simulations assume that queues on all approaches are empty as an initial condition and that video arrivals on external approaches follow a Poisson distribution; for demonstration purposes, a value of $q_{threshold} = 1$ (i.e., the presence of any queue) was assumed. The reward matrix was based on the objective being to minimize the queue length, and the reward calculated according to Eq.(32). In the case of the network simulation, the distance between any two adjacent intersections is chosen to be 1000 feet. The parameters used in the simulation (for all the movements) are summarized as follows:

Using the same set of input (arrival) data, the Markovian control algorithm and the conventional full-actuated control were applied to a four-legged isolated traffic intersection, such as that shown in Fig.1, with eight movements(four through movements and their corresponding left-turn movements) to evaluate their performances. The algorithm used to simulate full-actuated control was designed to mimic the logic of a common type 170 dual ring controller with parameters as specified in the previous table-eight-phase operation was assumed. To minimize initial condition effects, the two algorithms are applied for a simulated time of 65 min., and the average delay (per video) during the last dive minutes of the simulation is used for comparison. Two different general cases were considered: (1) uniform demand among all conflicting movements, and (2) the through traffic demand dominates the left-turn demand by a ratio of 2:1. The two algorithms were applied for different arrival rates, representing a range of both unsaturated and saturated conditions.(under the assumption of 2-second minimum headways, the intersection has a total capacity of 3600 videos per hour of green). In order to provide statistical significance for the simulation results, the two algorithms were tested on different sets of random data for each arrival rate (a total of forty on the cases in which left-moving traffic was assumed equal to through traffic, and fifteen in the cases in which left-moving traffic was equal to half of the through traffic).

The means of the average delay per video for the final- minute period of each set of forty simulations corresponding to the two cases of left-move to through traffic of 1.0(LM/T=1.0) and 0.5(LM/T=0.5) are plotted in Fig. 9, where “MCC” stands for the Markov congestion control algorithm and “FAC” stands for the full-actuated control. As a further “benchmark” comparison, delay calculations based on Webster’s delay equation for Poisson arrivals under fixed-time (pre-timed) control are also provided (labeled Pre (C=60s) and Pre(C=45s) for cycle lengths of 60 and 45 s, respectively.

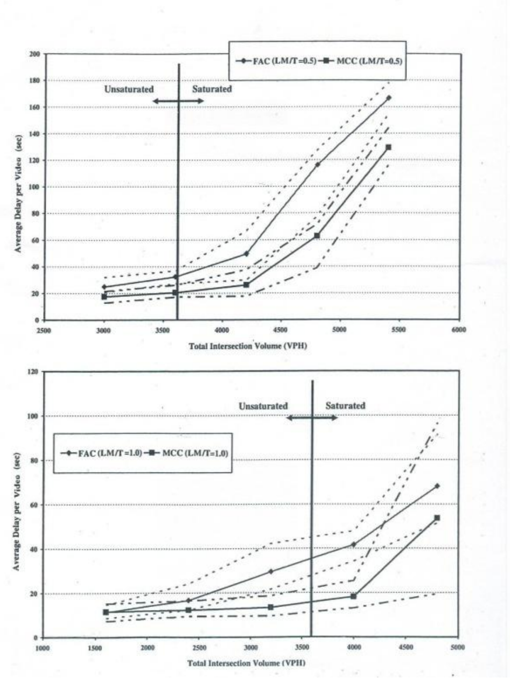
Significance tests based on t-statistics resulting from hypothesis tests on the difference of sample means indicate that the difference in means of the simulation results is significantly different (at 0.05 level or above) for all cases except for the LM/T = 1.0 case in which the total intersection volume 1500vph. The hypothesis tests on the difference of means assume that the two populations are independent and have a normal distribution. Alternatively, order statistics (distribution-free statistics) estimate the limits within which a certain percentage of the probability of the random variable with a certain degree of conference without having prior knowledge of the probability distribution. For the case involving 40 samples taken from a population, the upper/lower bound within which 90% of the probability of the random variable lies can be obtained with 95% confidence. Fig.10 displays these bounds on the steady state delay resulting both from full-actuated and from Markovian control algorithms.

Fig.9. Algorithm performance comparison for isolated intersection



From the above figures, except for the case in which the left-move traffic volume is equal to the through volume (LM/T=1.0) and the volume is relatively light(e.g., arrival rate is 200 video /hour/movement), the performance of the Markov algorithm is significantly better than the fully actuated controller(as well as the pre-timed controller). For example, for LM/T=1.0, when λ = 300, the Markov algorithm shows about a 25% improvement on the average steady state delay; for λ = 400 and λ = 500, the average steady state delay of the Markov controller is only about one half of that of the full-actuated controller. As expected, under saturated conditions both algorithm exhibit increasingly worse delay, although the Markov control (on average) still outperforms full-actuated control. The simulation results indicate that by applying the Markov congestion control algorithm, the average delay at an isolated intersection may be reduced dramatically (22-51%).

Fig.10.Upper and lower bounds on simulation results



The Markov congestion control algorithm was also tested on a typical traffic network of five intersections, such as that depicted in Fig.3. For this case, Poisson arrivals were assumed at the external inputs; the arrivals at all interval approaches are an outcome of the control strategy employed at associated upstream intersections. The tests were conducted for LM/T = 1.0 using five different arrival rates: λ = 200,300,400,500 and 600 video per hour per movement. The internal approaches linking the five intersections were assumed to be 1000 ft in length , and the average travel speed assumed to be 30mph(resulting in a value of T_{avg}=23s). The parameters in Robertson's platoon dispersion model were assumed to be α = 0.35, β = 0.8 – the common default values for Indian studies. The mean values(of the 40 sets of data) of the steady state delay are plotted in Fig. 11. The dotted lines in Fig. 11 display the upper/lower bounds within which 90% of the probability of the steady state delay resulting both from full-actuated and Markovian control algorithms lay.

The results indicate that the Markov algorithm substantially outperforms traditional full-actuated control, particularly when the intersection is at, or near saturation. For example, when λ ≤500(total intersection volume of 4000vph), the average steady state delay of the Markov controller is only about one half of that of the fully actuated controller. Under heavy over-saturated conditions (λ =600), delay with both algorithms tend to converge at a high value.

Fig.11. Algorithm performance comparison for simple network case

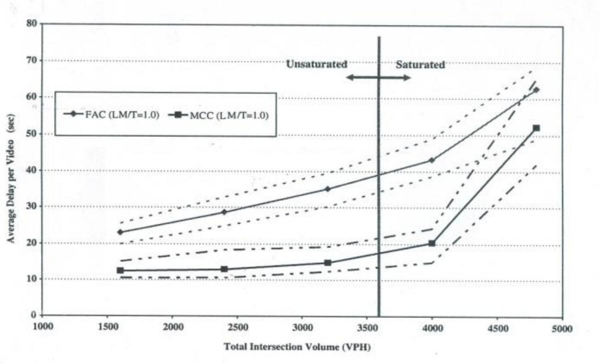
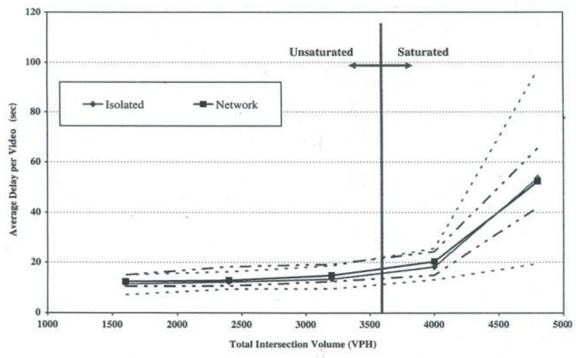
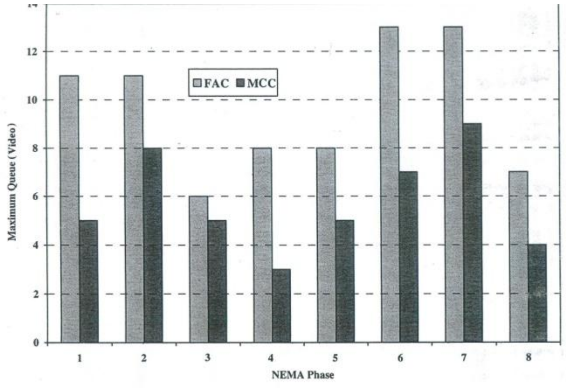


Fig.12. MCC performance comparison between network and isolated intersection examples



We note that, under simple five-node network conditions with identical arrival rates, the performance of the Markov control algorithm closely mirrors that obtained in the case of the isolated intersection example (Fig.12). Although preliminary, the result suggest that application of the algorithm in a network setting tends to decrease variability in performance; this is expected, since the variability expressed in the Poisson arrival patterns at the external nodes becomes as increasingly minor factor as the number of internal approaches increases. This latter factor may help to explain the large variance seen in the isolated intersection case under heavy oversaturation.

Fig. 13. Maximum queue comparison (3200 vph)



As stated previously, the specific objective used in these examples of application of the Markov congestion control algorithm was not specifically to minimize delay, but rather to minimize the queues on the intersection approaches; the delay performance characteristics presented above were an ancillary outcome of the specific objective. Relative to performance related to that specific objective, Fig.13 presents representative values of the maximum queues for each movements obtained for the network case in which the total intersection volume is 3200vph, or about 90% of intersection capacity.

The results indicate that the Markov congestion control algorithm significantly out performs full-actuated control in this aspect, although it must be noted that full-actuated control is not explicitly designed to minimize queue length, but rather implicitly works towards this end via its extension settings.

6. Summary and conclusions

Traffic signal control is a major ATMS component and its enhancement arguably is the most efficient way to reduce surface street congestion. The objective of the research presented here has been to present a more effective systematic approach to achieve real-time congested signal control for traffic networks.

In this research, the problem of finding optimal VTST timing plans has been solved as a decision-making problem for a controlled Markov process. Controlled Markov processes have been used extensively to analyze and control complicated stochastic dynamical systems; its probabilistic, decision making features match almost perfectly with the design features of a traffic signal control system. The Markovian decision model developed herein as the system model for signal control incorporates Robertson's platoon dispersion traffic model between intersections and employs the value iteration algorithm to find the optimal decision for the controlled Markov process. Analysis of computer simulation results indicates that this systematic approach is more efficient that the full-actuated control, especially under the conditions of high traffic demand.

There are ,of course, significant limitation to the present approach. Most notable is that as the size of the traffic network increases, i.e., the number of nodes/intersections and /or links increases, the dimensions of the Markovian control model increases dramatically, requiring more memory space and computation time. This dimensionality issue is very important to real-time implementation, where processing speed is crucial. In the current formulation, one potential solution to this problem is alluded to by decomposing the network into sets of inter-linked network kernels of five intersections that could be handled by distributed/parallel processing protocols; however, no attempt has been made to thoroughly investigate the issues of such decomposition algorithms. Further, before any attempt to implement the results, a comprehensive sensitivity analysis needs to be conducted to study the effects of the various parameters employed in the simulation testing on both the performance of the model as well as on the objective function. Finally, for field testing, the original C language code must first be rewritten into assembly language; then firmware can be loaded, or " burned in" to the PROM (Programmable Read Only Memory) chip of the controller. The work can be extended to multistage traffic flow conferencing.

References

[1] Aboudolas,K, Papagerorgiou.M, Kosmatopoulos.E, "Control and Optimization Methods for Traffic Control in Large-Scale Congested Urban Road Networks", American Control Conference, IEEE Transactions on 2007(3132-3138).

[2] B.Hoh, M.Gruteser, R.Herring, J.Ban, D.Work, J-C. Herrera, A.Bayen, M.Annavaram and Q.Jacobson, "Virtual trip lines for distributed privacy-preserving traffic monitoring", in 6th International Conference on Mobile system, Applications and Services (Breck-enridge,CO), pp. 15-28, June 17-18 2008.

[3] Camponogara.E, de Oliveira.L.B, "Distributed Optimization for Model Predictive Control of Linear-Dynamic Networks", Systems, Man and Cybernetics, Systems and Humans, IEEE Transactions 0n 2009(1331-1338).

[4] Chang.G and X.Tao. Estimation of Time-Dependent Turning Fractions at Signalized Intersections. Transportation Research Record 1997, 1644(1), 142-149.

[5] Chen.A, Chootinan.P, Recker.W, 2005. Estimating the quality of synthetic origin-destination trip table estimated by path flow estimator, Journal of Transportation Engineering, American Society of civil Engineers 131 (7), 506-513.

[6] Davis.G.A, Lan.D.C,1995, "Estimation intersection turning movement propotions from less-than-complete

sets of traffic counts", Transportation Research Record 15010, 53-59.

[7] D.Work, O.P. Tossavainen, Q.Jacobson and A.Bayen, "Lagrangian sensing: Distributed traffic estimation with mobile devices". Accepted for publication, ACC American Control Conference, 2009, Saint Louis, MO.

[8] D.Work, O.P. Tossavainen, S. Blandin,A, Bayen, T.Llwuchukwu and K.Tracton, "An ensemble Kalman filtering approach to highway traffic estimation using GPS enabled mobile devices", in Proc. Of the 47th IEEE Conference on Decision and Control, (Cancun, Mexico).

[9] Dal Yujie, Dongbin Zhao, "A Traffic Signal Control Algorithm for Isolated Intersections based on Adaptive Dynamic Programming" , ICNSC, International IEEE Conference on 2010.

[10] H.Gault and I.Taylor, "The use of output from vehicle detectors to access delay in computer- controlled area traffic control systems", Tech.Rep. Research Report No.31, Transportation Operation Research Group, University of Newcastle Upon Tyne, 1977.

[11] Henclewood.D, Hunter.M and R.Fujimoto. 2008. Proposed Methodology for a Data-Driven

Simulation for Estimating Performance Measures Along Signalized Arterials in Real-time, 2008.

[12] Jiuh-Biing Sheu, "A stochastic optimal control approach to real-time, incident-responsive traffic signal control at isolated intersections", Transport Science 2002.

[13] Kalige Wen, Shiru Qu, Yumel Zhang, "A stochastic adaptive control model for isolated

intersections", Robotics and Biomimetics, International IEEE Conference on 2007.

[14] L.Mihaylova, R.Boel and A.Hegyi, "Freeway traffic estimation within recursive Bayesian framework", Automatica, vol.43, no. 2, pp. 290.

[15] Liu.H.X, W.X.Ma and H.H.Wu, "Development of a Real-Time Arterial performance Monitoring System Using Traffic Data Available from Existing Signal System", Queue Minneapolis on 2009.

[16] Liu.H.X and W.X.Ma 2007, "Time-Dependent Travel Time Estimation Model for Signalized Arterial Network", TRB Annual Meeting 2007 (vol.1, p.21). Washington DC: National Academies.

[17] Liu.H.X and W.X.Ma 2007, "Time-Dependent Travel Time Estimation Model for Signalized Arterial Network", TRB Annual Meeting 2007 (vol.1, p.21). Washington DC: National Academies.

[18] Lo.H.K, 2001. "A cell-based traffic control formulation: strategies and benefits of dynamic timing plans", Transportation Sciences 35 (2), 148-164.

[19] Lowrie.P, 1982,"The Sydney coordinated adaptive control system-principles, methodology, algorithms". In: IEEE Conference Publication, 207.

[20] Maher.K.J, 1984. "Estimating the turning flows at a junction: a comparison of three models", Traffic Engineering and control 25, 19-22.

[21] P B. Mirchandani, S A.Nobe, W. Wu, 2001, " The use of on-line turning proportion estimation in real-time traffic-adaptive signal control", Transportation Research Record1728, 80-86.

[22] P.Cheng, Z.Qiu and B.Ran, "Particle filter based traffic state estimation using cell phone network data", in Proc. IEEE Intelligent Transprotation Systems Conference ITSC'06, pp.1047-1052, 2006.

[23] P.Hao, X.Ban, R.Herring and A.Bayen, "Delay pattern estimation for signalized intersections using sampled travel times", in 2009 Transportation Research Board Annual Meeting, (Washington,D.C.), 2009.

[24] Papoulis. A, 1984. "Brownian movement and markoff process. Probability, Random variables and Stochastic Processes", second edition, McGraw-Hill, New York, pp. 515-553.

[25] Robertson.D.I, Bretherton.R.D, 2002, "Optimizing networks of traffic signals in real-time: the SCOOT method", IEEE Transaction on Vehicular Technology 40,1.

[26] S.Vijayakumar, "Adaptive signal control at isolated intersections", Journal of Transportation

Engineering.

[27] Singh.M.G, Tamura.H, 1974. Modelling and Hirearchical optimization for oversaturated urban road traffic network. Internation Journal of Control 20(6), 913-934.

[28] Stephanedes.Y.J, Chang.K.K, 1993. Optimal control of freeway corridors. Journal of Transportation Engineering 119(4),504-514.

[29] T.Park and S.Lee, "A Bayesian approach for estimating link travel time in urban arterial traffic network, "Lecture notes in computer science, p.1017, 2004.

[30] V.Sisiopiku and N.Rouphail, "Travel time estimation from loop detector data for advanced traveler information system applications", Tech. Rep., Illinois University Transportation Research Consortium, 1994.

[31] Wei Cheng, Xiaolan Liu, Wenfeng Zhang, " An optimal adaptive traffic signal control algorithm for intersection", Intelligent Control and Automation 2006.

[32] Wilshire.R, Black.R, etal, 1985. " Traffic control systems Handbook", FHWA-IP-85-12.

[33] Yu, Nie, Zhang.H.M, Recker.W.W, 2005, "Inferring Origin-destination trip matrices with a

decoupled GLS path flow estimator", Traffic Research Part B: Methodological 39 (6),

497-518.