Discrete-Time Point Processes in Urban Traffic Queue Estimation

JOHN S. BARAS, MEMBER, IEEE, WILLIAM S. LEVINE, MEMBER, IEEE, AND TAHSIN L. LIN

Abstract—This research was motivated by the belief that it is possible to develop improved algorithms for the computer control of urban traffic. Previous research suggested that the computer software, and especially the filtering and prediction algorithms, is the limiting factor in computerized traffic control. Since the modern approach to filtering and prediction begins with the development of models for the generation of the data and since these models are also useful in the control problem, this paper deals with the modeling of traffic queues and filtering and prediction.

It is shown that the data received from vehicle detectors is a discrete-time point process. The formation and dispersion of queues at a traffic signal is then modeled by a discrete-time time-varying Markov chain which is related to the observation point process. Three such models of increasing complexity are given. Recent results in the theory of point-process filtering and prediction are then used to derive the nonlinear minimum error variance filters/predictors corresponding to these models. It is then shown that these optimal estimators are computationally feasible in a microprocessor. All three algorithms were tested against the UTCS-1 traffic simulator and, in one case, against an algorithm in current use called ASCOT. Some results of these tests are shown. They indicate good performance in every case and better performance than ASCOT in the comparable case.

Manuscript received April 27, 1978; revised September 11, 1978. Paper recommended by H. J. Payne, Past Chairman of the Transportation Systems Committee. This work was supported in part by the U. S. Department of Transportation under Contract DOT-OS-60134, and in part by the University of Maryland Computation Center.

The authors are with the Department of Electrical Engineering, University of Maryland, College Park, MD 20742.

I. INTRODUCTION

THERE IS considerable current interest in the development of computer-based systems for the control of urban traffic. Over 25 such systems have been installed within the United States and approximately 125 others are in various stages of implementation [1]. Such systems have the potential to reduce traffic delay, fuel consumption, air

0018-9286/79/0200-0012$00.75 ©1979 IEEE

pollution, and accidents. It has been estimated that improved traffic signal systems could save 800 million gallons of fuel annually in the United States [2].

Generally, these computer-based systems consist of a collection of conventional looking traffic lights, a collection of vehicle detectors (usually inductive loops), and a computer that adjusts the traffic lights based to some extent on the signals from the detectors. However, the systems that are in use today are, from the control engineer's viewpoint, rather crude. The Federal Highway Administration recently built a system (the Urban Traffic Control System or UTCS) in Washington, DC to serve as a test of more advanced control procedures [3]. The UTCS was built in three versions. The first generation was a conventional system in which the computer was used to detect traffic patterns based on detector data. The computer then chose one of six previously determined timing patterns and set the traffic lights accordingly. The timing pattern in use could be changed, at most, every 15 minutes. It should be apparent that this is basically an open-loop system. Consequently, it should not be surprising that the first generation system produced only slight improvement over a completely open-loop system in which signal timing patterns were chosen according to time of day. The second and third generations of UTCS were designed to be successively more traffic responsive and to rely more heavily on the detector data and on-line signal optimization. The second and third generation systems performed *worse* than the first generation [4], the costs of data transmission escalated, and UTCS was shut down in the fall of 1976. Tarnoff [4] argues that this degradation in performance was due to errors in surveillance and prediction of traffic.

As control engineers, we could find several other reasons for the poor performance of the second and third generation systems but we also felt that the filtering and prediction could be substantially improved. Furthermore, since the models developed to solve the filtering and prediction problem could then be used in the control problem, we felt that filtering and prediction was a good starting point.

We have since developed four filters/predictors for various aspects of urban traffic. Three of these, all used for estimation of queues at traffic signals, are described in this paper. A companion paper describes the fourth which is used for estimating the size of platoons of vehicles in the network. In order to keep the length of these papers reasonable most of the details of our work have been omitted. The interested reader can find a much more detailed treatment in [5].

II. Discrete Time Point Processes and Traffic Queuing Models

The modeling, and with it the filtering and prediction, of signals that are indirectly observed via a point process has progressed considerably in the past few years thanks to the efforts of Bremaud [6], Boel, Varaiya, and Wong [7], [8], Snyder [9], Segall, Davis, and Kailath [10], Davis [11], and many others. In applications, modeling is the fundamental problem since some models lead directly to finite dimensional realizations of the optimal filter/predictor while other models do not. We present here a brief summary of results on the modeling of discrete-time point processes following Segall [12].

Consider a sequence of observations $\{n(t)\}_{t=1}^{\infty}$ with $n(t)=0$ or $n(t)=1$ being the only possibilities for each t. Suppose the probability that $n(t)=1$ is influenced by previous occurrences as well as by some other related sequence $\{x(t)\}_{t=1}^{\infty}$ ($x(t)$ may be vector-valued). The factors that may affect the occurrence probability at time t are the past observations denoted

$$n^{t-1}=\{n(1),n(2),\cdots,n(t-1)\} \tag{2.1}$$

and the past and present of the related sequence

$$\boldsymbol{x}^{t}=\{\boldsymbol{x}(1),\boldsymbol{x}(2),\cdots,\boldsymbol{x}(t)\}. \tag{2.2}$$

The information carried by these signals is as usual denoted by the σ-algebra generated by them,

$$\mathcal{B}_{t-1}=\sigma\{n^{t-1},\boldsymbol{x}^{t}\}. \tag{2.3}$$

We then define $a(\cdot)$ by

$$\Pr\{n(t)=1|\mathcal{B}_{t-1}\}=1-\Pr\{n(t)=0|\mathcal{B}_{t-1}\} \triangleq a(t,n^{t-1},\boldsymbol{x}^{t}). \tag{2.4}$$

Then

$$E^{\mathcal{B}_{t-1}}\{n(t)\} \triangleq E\{n(t)|\mathcal{B}_{t-1}\}=a(t,n^{t-1},\boldsymbol{x}^{t}), \tag{2.5}$$

where $E^{\mathcal{B}_{t-1}}\{z\}=E\{z|\mathcal{B}_{t-1}\}$ is the conditional expectation of z given $\mathcal{B}_{t-1}$. If we define

$$\omega(t) \triangleq n(t)-a(t,n^{t-1},\boldsymbol{x}^{t}) \tag{2.6}$$

then

$$E^{\mathcal{B}_{t-1}}\{\omega(t)\}=0, \tag{2.7}$$

which says, roughly, that $\omega(t)$ is unpredictable given the information represented by $\mathcal{B}_{t-1}$.

Similarly, if we write

$$\boldsymbol{x}(t+1)=E^{\mathcal{B}_{t-1}}\{\boldsymbol{x}(t+1)\} + \left[\boldsymbol{x}(t+1)-E^{\mathcal{B}_{t-1}}\{\boldsymbol{x}(t+1)\}\right] \tag{2.8}$$

and define

$$\boldsymbol{f}(t,n^{t-1},\boldsymbol{x}^{t})=E^{\mathcal{B}_{t-1}}\{\boldsymbol{x}(t+1)\} \tag{2.9}$$

$$\boldsymbol{u}(t)=\boldsymbol{x}(t+1)-E^{\mathcal{B}_{t-1}}\{\boldsymbol{x}(t+1)\} \tag{2.10}$$

we obtain

$$E^{\mathcal{B}_{t-1}}\{\boldsymbol{u}(t)\}=0.$$

Assembling all of the above gives

$$\left.\begin{aligned}\boldsymbol{x}(t+1)&=\boldsymbol{f}(t,n^{t-1},\boldsymbol{x}^{t})+\boldsymbol{u}(t)\\ n(t)&=a(t,n^{t-1},\boldsymbol{x}^{t})+\omega(t).\end{aligned}\right\} \tag{2.11}$$

We emphasize that this is *not* a "signal plus noise" model and that this model applies to *any* discrete-time point process that is related to another time-varying quantity. The equations simply reflect the fact that *any* observation sequence can be divided into the sum of a *predictable* part and an *unpredictable* part. Thus, the modeling problem reduces to finding the functional form of $a(t,n^{t-1},x^t)$ and $f(t,n^{t-1},x^t)$.

We now apply the above modeling procedure to produce models for traffic queues.

A. A Simple Queuing Model (Model A)

The simplest practical traffic flow estimation problem occurs in the case of the single, isolated, intersection of two one-way, single-lane streets. In order to adjust the traffic light to optimize, in some sense, (or even improve) the flow of traffic it is necessary to obtain fairly good estimates of the traffic queues upstream from the intersection. In practical systems the estimate needs to be based on a minimal amount of historical data and on the signals from one or more detectors positioned as shown in Fig. 1. Assume, for simplicity, that the light operates on a simple, known, red-green cycle (no amber), that there is only one detector, and that the detector is located N car lengths from the stop line.

The observed signal from the detector will be denoted by $n^a(t)$,

$$n^a(t)=\begin{cases}0, & \text{if no vehicle is over the detector}\\ 1, & \text{if a vehicle is over the detector.}\end{cases} \tag{2.12}$$

In practice, time is discretized with a small enough discretization interval (1/32 s in UTCS) for each vehicle to be over the detector for several samples. For simplicity, it is assumed here that the data are sampled so that each vehicle produces exactly one pulse (one 1).

There are many factors which affect the rate process associated with $n^a(t)$. We believe that two of the most important are the upstream traffic signal and the number of vehicles in the queue. Thus, in this simplest model we let

$\lambda(k,t)$ = rate at which vehicles arrive at the detector given that the queue length is k.

$z(t)$ = queue length at time t.

Equivalently,

$$\lambda(k,t)=\Pr\left[n(t)=1|z(t)=k,t\right]$$
$$\lambda(N,t)=\Pr\left[n(t)=1|z(t)=N,t\right]=0. \tag{2.13}$$

Also

$$\lambda(k,t)=\begin{cases}\lambda_{kr}, & \text{when upstream traffic light is red}\\ \lambda_{kg}, & \text{when upstream traffic light is green.}\end{cases} \tag{2.14}$$

Although we do not do so, one can account for the delay between the time traffic departs from the upstream signal and the time it arrives at the detector by appropriately adjusting the "phase" of $\lambda(t)$ with respect to the upstream traffic signal.

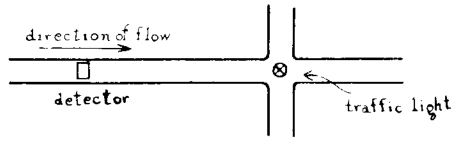


Fig. 1. Detector location.

Similarly, there is an unobserved point process $n^d(t)$ at the stop line

$$n^d(t)=\begin{cases}1, & \text{if a vehicle departs at time } t\\ 0, & \text{otherwise.}\end{cases} \tag{2.15}$$

We assume that the rate process corresponding to $n^d(t)$ is dominated by the downstream traffic signal and by the number of vehicles in the queue. Then,

$\mu(k,t)$ = rate at which vehicles depart from the queue given that the queue length is k.

So,

$$\mu(k,t)=\Pr\left[1 \text{ departure}|z(t)=k,t\right] \tag{2.16}$$
$$\mu(0,t)=0, \qquad \text{for all } t$$

and

$$\Pr\left[\text{more than 1 departure}\right]=0$$

$$\mu(k,t)=\begin{cases}\mu_{kr}, & \text{when downstream traffic light is red}\\ \mu_{kg}, & \text{when downstream traffic light is green.}\end{cases} \tag{2.17}$$

Furthermore, assume that arrivals and departures, conditioned on knowledge of the queue, are independent.

Examination of real traffic data shows that the assumption of conditionally inhomogeneous Poisson arrivals and departures is not strictly correct. It is also obvious that the coarse time discretization is throwing away useful information about the velocity with which vehicles cross the detector. There are three very good reasons for making these assumptions despite the inaccuracies they introduce. First, it will be seen that the filter/predictor based on these assumptions tends to ignore the extra randomness inherent in the conditionally Poisson assumption. Second, examination of real traffic data shows that the time dependence of vehicle arrivals caused by upstream traffic signals is a dominant effect and this is accurately modeled. Finally, the filter/predictor based on these assumptions is easily implemented in a microprocessor, and can easily be made adaptive.

It should also be noted that the assumption of a single-lane street is inessential.

This actually completes the construction of a model for the point process $n^a(t)$. To see this, and to put the model in the form of (2.11), define

$$Q_{ij}^T(t)=\Pr[\text{queue at time } t+1 \text{ contains } i \text{ vehicles}|\text{queue at time } t \text{ contains } j \text{ vehicles}]. \tag{2.18}$$

At this point we make the approximation that a vehicle joins the queue the instant that it crosses the detector. Thus, strictly speaking, our "queue" is the number of vehicles between the detector and the stopline. With this approximation, it is elementary that

$$\left.\begin{array}{l} Q_{ii}^T(t)=(1-\lambda(i))(1-\mu(i))+\lambda(i)\mu(i), \\ Q_{i,i-1}^T(t)=\lambda(i-1)(1-\mu(i-1)), \\ Q_{i,i+1}^T(t)=\mu(i+1)(1-\lambda(i+1)), \\ Q_{ij}^T=0, \quad \text{elsewhere,} \end{array}\right\} \quad i=0,1,\cdots,N \tag{2.19}$$

where the argument t has been suppressed in both λ and μ. Introduce the row vector

$$\boldsymbol{\lambda}^T(t)=[\lambda(0,t),\lambda(1,t),\cdots,\lambda(N-1,t),0] \tag{2.20}$$

and following Segall [12], define

$$x_k(t)=\begin{cases} 1, & \text{if there are } k \text{ vehicles in the "queue" } k=0,1,\cdots,N \\ 0, & \text{otherwise} \end{cases} \tag{2.21}$$

$$\boldsymbol{x}^T(t)=[x_0(t),x_1(t),\cdots,x_N(t)].$$

It is now straightforward to establish that

$$\left.\begin{array}{l} \boldsymbol{x}(t+1)=\boldsymbol{Q}^T(t)\boldsymbol{x}(t)+\boldsymbol{u}(t) \\ n(t)=\boldsymbol{\lambda}^T(t)\boldsymbol{x}(t)+\omega(t) \end{array}\right\} \tag{2.22}$$

where $\boldsymbol{u}(t)$ and $\omega(t)$ are "noise" processes. More precisely, $\boldsymbol{u}(t)$ and $\omega(t)$ are martingale difference sequences with respect to the σ-algebra generated by the sequences $\{n(0),n(1),\cdots,n(t-1)\}$ and $\{\boldsymbol{x}(0),\boldsymbol{x}(1),\cdots,\boldsymbol{x}(t)\}$.

B. More Detailed Single Detector Queue Model (Model B)

There are two changes that, on theoretical grounds, ought to improve the above model of traffic queuing at a signal. The first arises because many detectors give velocity (or a signal related to velocity) in addition to occurrence time for each vehicle that crosses the detector. The second involves defining the queue more accurately as the number of vehicles that are actually stopped at the traffic signal. The model described here incorporates both of these improvements.

It is convenient to think of the queue and the detector as characterized by three point processes (two of which are not observed).

The first is the observed arrival process at the detector, denoted by

$$y^a(t)=\begin{cases} 0, & \text{if no vehicle crosses the detector at time } t \\ v_j, & \text{if a vehicle crosses the detector at time } t \text{ with velocity } v_j. \end{cases} \tag{2.23}$$

Note that 1) we discretize velocity with a fairly coarse discretization so that $j=1,2,\cdots,J$ (J small, around 5); and 2) this model applies to the detector model incorporated in the UTCS-1 simulation. It needs to be modified slightly for a real detector. It is convenient to represent $y^a(t)$ as the sum of J point processes denoted by

$$n_j^a(t)=\begin{cases} 0, & \text{if no vehicle crosses the detector at time } t \\ 1, & \text{if a vehicle crosses the detector at time } t \text{ with velocity } v_j, \\ & j=1,2,\cdots,J. \end{cases} \tag{2.24}$$

Obviously,

$$\boldsymbol{n}^a(t)=\begin{bmatrix} n^a(t) \\ \cdot \\ \cdot \\ n_J^a(t) \end{bmatrix} \quad \text{contains the exact same information as } y^a(t).$$

The second is the unobserved arrival process at the tail of the queue, denoted by

$$n^t(t)=\begin{cases} 0, & \text{if no vehicle joins the queue at time } t \\ 1, & \text{if a vehicle joins the queue at time } t. \end{cases} \tag{2.25}$$

The third is the unobserved departure process at the head of the queue, denoted by

$$n^h(t)=\begin{cases} 0, & \text{if no vehicle leaves the queue at time } t \\ 1, & \text{if a vehicle leaves the queue at time } t. \end{cases} \tag{2.26}$$

We are most interested in the number of vehicles in the queue at each instant of time. To keep track of this number we again define

$$x_k(t)=\begin{cases} 1, & \text{if there are } k \text{ vehicles in the queue at time } t,\ k=0,1,\cdots,N \\ 0, & \text{otherwise.} \end{cases} \tag{2.27}$$

If the street segment of interest fills we activate a similar scheme for the street segment upstream from the detector.

As explained earlier, we now have to characterize each of these point processes by writing explicit expressions for the dependence of their "rates" on the observed sample paths of $\boldsymbol{n}^a(t)$, $n^t(t)$, $n^h(t)$, and $\boldsymbol{x}^t$. We begin with the rate processes associated with each of the components of $\boldsymbol{n}^a(t)$. Thus,

$$\lambda_j^a(\boldsymbol{x}^t,\boldsymbol{n}^{a,t-1},n^{h,t-1},n^{t,t-1},t) \triangleq \Pr\{n_j^a(t)=1|\mathcal{B}_{t-1}\}.$$

As in model A, we make the simplifying assumption that

$$\begin{aligned} \lambda_j^a(\boldsymbol{x}^t,\boldsymbol{n}^{a,t-1},n^{h,t-1},n^{t,t-1},t) &= \lambda_j^a(\boldsymbol{x}(t),t) \\ &= \sum_{i=0}^{N} \lambda_j^a(i,t)x_i(t); \qquad j=1,2,\cdots,J \end{aligned} \tag{2.28}$$

where

$\lambda_j^a(i,t) = \Pr\{$a vehicle crosses the detector with velocity v_j given that there are i vehicles in the queue at time $t\}$.

Next, we assume

$$\lambda_j^a(i,t) = v_{ji}\lambda^a(i,t) \tag{2.29}$$

where

$v_{ji} = \Pr\{$vehicle crosses detector with velocity $v_j|$a vehicle crosses the detector and $x_i(t)=1\}$

$\lambda^a(i,t)$ is identical to the $\lambda(i,t)$ used in model A, (2.13), (2.14).

We are aware of, and use elsewhere [5], [13], the fact that vehicles tend to arrive in platoons so that λ_j^a depends on $\boldsymbol{n}^{a,t-1}$ as well as on $\boldsymbol{x}(t)$ and t. However, this assumption greatly simplifies the model. Since the platoon arrivals are highly correlated with the upstream traffic signal, the approximation introduces only a small error.

We still have to specify the matrix $\boldsymbol{V}$ whose jith component is v_{ji} above. A relation between the velocity over the detector and the queue length is known to exist, has been experimentally measured, and has been used before to help estimate the queue [14], [15]. In the report [5] we give a derivation for the matrix $\boldsymbol{V}$ based on some (widely accepted [15]) assumptions about the way drivers decelerate to join a queue. The estimator based on this $\boldsymbol{V}$ performs fairly well (see Section IV) suggesting that crude estimates of $\boldsymbol{V}$ are sufficient.

Next, we construct a model for the arrival process $n^t(t)$ at the tail of the queue. As before, this really means modeling the rate

$$\lambda^t(\boldsymbol{x}(t),\boldsymbol{n}^{a,t-1},n^{h,t-1},n^{t,t-1},t) \triangleq \Pr\{n^t(t)=1|\mathfrak{B}_{t-1}\}.$$

Since n^h and n^t are not observed, the fundamental problem is to model the dependence of λ^t on $\boldsymbol{x}(t)$, t, and $\boldsymbol{n}^{a,t-1}$. It is obvious that there is a delay between the appearance of a vehicle with velocity v_j at the detector and the time that vehicle joins the queue. This delay clearly depends on the number of vehicles in the queue ($\boldsymbol{x}(t)$) and the velocity with which the vehicle crossed the detector. In the report [5], we give a detailed derivation for this delay based on reasonable assumptions about the way vehicles decelerate to join a queue. In any case, for each vehicle that crosses the detector (say vehicle k) we define a deterministic vector that corresponds to the expected time at which that vehicle joins the queue. We denote this vector by

$$\boldsymbol{\tau}_k = [\tau_{k0}, \tau_{ki}, \cdots, \tau_{kN}]^T \tag{2.30}$$

where τ_{ki} = expected time that kth vehicle joins the queue given that the queue contained i vehicles when the kth vehicle passed the detector. Note that $\boldsymbol{\tau}_k$ is only defined after the kth vehicle crosses the detector. The only values of $\boldsymbol{\tau}_k$ that are of interest correspond to vehicles that have crossed the detector but have not yet joined the queue. To keep track of these values of $\boldsymbol{\tau}_k$, we define

$\sigma_{1i} \triangleq$ the smallest known value of τ_{ki} satisfying the inequality $t-\tau_{ki} \leqslant 1$ s;

$\sigma_{2i} \triangleq$ the smallest known value of τ_{ji} satisfying the inequalities $\tau_{ji} > \sigma_{1i}$ and $t-\tau_{ji} \leqslant 1$ s;

$\sigma_{3i} \triangleq$ the smallest known value of τ_{mi} satisfying the inequalities $\tau_{mi} > \sigma_{2i}$ and $t-\tau_{mi} \leqslant 1$;

$$\boldsymbol{\sigma} \triangleq [\boldsymbol{\sigma}_1 \vdots \boldsymbol{\sigma}_2 \vdots \boldsymbol{\sigma}_3]. \tag{2.31}$$

Finally, we assume that

$$\lambda^t(\boldsymbol{x}^t,\boldsymbol{n}^{a,t-1},n^{h,t-1},n^{t,t-1},t) = \lambda^t(\boldsymbol{x}(t),t,\boldsymbol{\sigma}). \tag{2.32}$$

But

$$\begin{aligned}\lambda^t(\boldsymbol{x}(t),t,\boldsymbol{\sigma}) &= \Pr\{\text{a vehicle joins the queue at time } t \text{ given } \boldsymbol{\sigma} \text{ and } \boldsymbol{x}(t)\}\\ &= \Pr[n^t(t)=1|x_i(t)=1,\boldsymbol{\sigma}].\end{aligned}$$

Thus, to complete the model, one has to explicitly specify the above probability for each value of i and $\boldsymbol{\sigma}$. We make the approximation that

$$\Pr[n^t(t)=1|x_i(t)=1,\boldsymbol{\sigma}_k] = \begin{cases}0, & i=N\\ 0.4, & i<N,\ t=\sigma_{ki}\\ 0.29, & i<N,\ |t-\sigma_{ki}|=1\\ 0.02, & \text{otherwise}\end{cases} \tag{2.33}$$

where $k=1,2,3$. We also assume that the above arrival probabilities are conditionally independent for different values of k. Thus, we have that

$$\Pr[n^t(t)=1|x_i(t)=1,\boldsymbol{\sigma}] = \sum_{k=1}^{3} \Pr[n^t(t)=1|x_i(t)=1,\boldsymbol{\sigma}_k]. \tag{2.34}$$

The choice of distribution is, obviously, arbitrary. The parameter given as 0.02 is intended to reflect the probability that a vehicle joins the queue without crossing the detector. If there was an entrance to a parking garage between the detector and the stop line, that number would need to be larger. The assumption that the vehicle joins the queue at its estimated arrival time ± 1 s is also ad hoc but is reasonable. In principle, the probability in (2.33) could be measured. Such a measurement would require a great deal of work, but since the results should not depend too heavily on the specific street, a few such measurements would probably suffice for the entire country. Furthermore, our results suggest that it may be adequate to use a guessed distribution. This completes the model of the arrival process at the tail of the queue.

We next model the departure process at the head of the queue by

$$\lambda^h(x(t),t,\text{etc.})=\begin{cases}\lambda^h(t), & \text{if } x_0(t)=0\\ 0, & \text{if } x_0(t)=1\end{cases} \quad (2.35)$$

where

$$\lambda^h(t)=\begin{cases}0, & \text{signal is red}\\ \lambda^h, & \text{signal is green.}\end{cases} \quad (2.36)$$

This is a relatively crude approximation. We know the departure rate changes slightly at the time the signal changes and we know the departure rate is larger for moving traffic than it is for stopped traffic. However, this model is simple and, we hope, reasonably accurate.

Now, it is a straightforward matter to place this model in the form of (2.11). Specifically,

$$\left.\begin{aligned}Q_{ii}^T(t,\sigma)&=(1-\lambda^t(i,t,\sigma))(1-\lambda^h(i,t))\\&\quad+\lambda^t(i,t,\sigma)\lambda^h(i,t)\\Q_{i,i-1}^T(t,\sigma)&=\lambda^t(i,t-1,\sigma)(1-\lambda^h(i,t-1))\\Q_{i,i+1}^T(t,\sigma)&=(1-\lambda^t(i,t+1,\sigma))\lambda^h(i,t+1)\\Q_{ij}^T(t,\sigma)&=0,\qquad i\neq j,\ i+1\neq j,\ i-1\neq j.\end{aligned}\right\} \quad (2.37)$$

Thus

$$\begin{aligned}x(t+1)&=Q^T(t,\sigma)x(t)+u(t)\\n_j^a(t)&=\sum_{i=0}^{N}v_{ji}\lambda^a(t,i)x_i(t)+\omega_j(t),\qquad j=1,2,\cdots,J\end{aligned} \quad (2.38)$$

completes the model with $u(t)$ and $\omega(t)$ having similar properties as for model A. It is slightly more convenient to write the second part of (2.38) as

$$n^a(t)=Vx(t)x^T(t)\lambda^a(t)+\omega(t) \quad (2.39)$$

where

$$\lambda^a(t)=\begin{bmatrix}\lambda^a(0,t)\\ \vdots\\ \lambda^a(N,t)\end{bmatrix}.$$

C. Two Detector Queue Model (Model C)

In this subsection we develop a queue model utilizing an additional detector located at the stop line.

Fortunately, we have done all the hard work in developing Model B. The only problem involved in augmenting Model B to utilize the information available from the additional detector is to characterize the point process at the new detector. We assume the new detector provides *only* occurrence times (no velocity data), since there is relatively little information in the velocity at the stop line. Proceeding, let

$$n^d(t)=\begin{cases}1, & \text{if a vehicle crosses the stop line detector at time } t\\ 0, & \text{otherwise.}\end{cases}$$

We assume the associated rate

$$\begin{aligned}\lambda^d(t,\mathcal{B}_{t-1})&=\lambda^h(x(t),t)\\&=\begin{cases}0, & \text{if signal is red}\\ \lambda^d, & \text{if signal is green and } x_0(t)=0\\ 0, & \text{if } x_0(t)=1.\end{cases}\end{aligned} \quad (2.40)$$

This is obviously an approximation to the reality but we believe it is an adequate approximation. The model now becomes

$$x(t+1)=Q^T(t,\sigma)x(t)+u(t) \quad (2.41)$$

$$\begin{bmatrix}n^a(t)\\ n^d(t)\end{bmatrix}=\begin{bmatrix}Vx(t)x^T(t)\lambda^a(t)\\ x(t)^T\lambda^d(t)\end{bmatrix}+\begin{bmatrix}\omega(t)\\ \omega_{J+1}(t)\end{bmatrix} \quad (2.42)$$

where $Q(t,\sigma)$ is as in (2.37) (the corresponding equation in Model B) and

$$\lambda^d(t)=\begin{bmatrix}0\\ \lambda^d\\ \vdots\\ \lambda^d\end{bmatrix},\qquad \text{if signal is green} \quad (2.43)$$

$$\lambda^d(t)=\mathbf{0},\qquad \text{if signal is red.}$$

We note that we have also obtained simpler two detector models. The simplest is described by replacing $\lambda^t(i,t,\sigma)$ in (2.37) with $\lambda(i,t)$ as in model A and using the second of (2.22) in place of the first of (2.42). An intermediate complexity model is obtained by replacing $\lambda^t(i,t,\sigma)$ in (2.37) with $\lambda^a(i,t)$ from (2.29) and using the same (2.42).

In summary, it should be clear that many other similar, queuing models could be constructed. In fact, it is hoped that the ones constructed here demonstrate the technique so that the reader can, if he wishes, construct a model of his own.

III. Filter/Predictors Based on Queuing Models

All of the filter/predictors developed in this section are based on the following result from Segall [12]: given an observed point process $n(t)$ that is related to a "signal" process $x(t)$ via (see also (2.11))

$$x(t+1)=f(t,n^{t-1},x^t)+u(t);\ x(1)=x_1 \quad (3.1)$$

$$n(t)=a(t,n^{t-1},x^t)+\omega(t) \quad (3.2)$$

where $\{u(t)\}$ and $\{\omega(t)\}$ are martingale difference sequences with respect to $\{\mathcal{B}_{t-1}\}$. Then $\hat{x}(t+1|t)$, the minimum square error estimate of $x(t+1)$ given the observations

$$\mathcal{F}_t = \sigma\{n(1), n(2), \cdots, n(t)\}$$

satisfies the recursive formula

$$\hat{\boldsymbol{x}}(t+1|t) = \hat{\boldsymbol{f}}(t|t-1) + (\hat{a}(t|t-1) - \hat{a}^2(t|t-1))^{-1} \cdot E^{\mathcal{F}_{t-1}}\{\tilde{\boldsymbol{f}}(t|t-1)\tilde{a}(t|t-1) + \boldsymbol{u}(t)\omega(t)\}\nu(t) \quad (3.3)$$

with

$$\hat{\boldsymbol{x}}(1|0) = E\{\boldsymbol{x}(1)\}, \quad (3.4)$$

where

$$\hat{\boldsymbol{f}}(t|t-1) = E^{\mathcal{F}_{t-1}}\{\boldsymbol{f}(t)\} = E\{\boldsymbol{f}(t)|n(t-1), n(t-2)\cdots\} \quad (3.5)$$

$$\tilde{\boldsymbol{f}}(t|t-1) = \boldsymbol{f}(t) - \hat{\boldsymbol{f}}(t|t-1) \quad (3.6)$$

$\hat{a}(t|t-1)$ and $\tilde{a}(t|t-1)$ are defined similarly

and

$$\nu(t) = n(t) - \hat{n}(t|t-1) \quad (3.7)$$

is the innovations process. All that remains is to compute explicit formulas for the expectations contained in (3.3)–(3.7).

A. Filter/Predictor Using Occurrence Times Only (Model A)

It is a straightforward calculation, which can be found in Segall [12], to show that the optimal one-step predictor based on Model A is

$$\hat{\boldsymbol{x}}(t+1|t) = \boldsymbol{Q}^T(t)\hat{\boldsymbol{x}}(t|t-1) + \frac{[\boldsymbol{S}^T(t)\hat{\boldsymbol{x}}(t|t-1) - \boldsymbol{Q}^T(t)\boldsymbol{\Sigma}(t)\boldsymbol{\lambda}(t)]}{[\boldsymbol{\lambda}^T(t)\hat{\boldsymbol{x}}(t|t-1) - (\boldsymbol{\lambda}^T(t)\hat{\boldsymbol{x}}(t|t-1))^2]} \cdot (n(t) - \boldsymbol{\lambda}^T(t)\hat{\boldsymbol{x}}(t|t-1)) \quad (3.8)$$

$$\hat{\boldsymbol{x}}(1|0) = \boldsymbol{\pi}(0) = \text{a priori probability distribution of queue length at } t=0 \quad (3.9)$$

where

$$\boldsymbol{\Sigma}(t) = \hat{\boldsymbol{x}}(t|t-1)\hat{\boldsymbol{x}}^T(t|t-1) \quad (3.10)$$

and $\boldsymbol{S}(t)$ is defined by

$$S_{ij}(t) = \Pr[x_j(t+1) = 1, n(t) = 1 | x_i(t) = 1]. \quad (3.11)$$

It is easily shown that

$$\left.\begin{aligned} S_{ii}(t) &= \lambda(i,t)\mu(i,t) \\ S_{i,i+1}(t) &= \lambda(i,t)(1-\mu(i,t)) \\ S_{ij}(t) &= 0, \quad j \neq i, j \neq i+1. \end{aligned}\right\} \quad (3.12)$$

On the other hand, the filter (i.e., the minimum square error estimate of $\boldsymbol{x}(t)$ given the observations $\mathcal{F}_t$) is given [12] by

$$\hat{\boldsymbol{x}}(t|t) = \hat{\boldsymbol{x}}(t|t-1) + \frac{[\mathrm{diag}\{(\hat{\boldsymbol{x}}(t|t-1))\} - \boldsymbol{\Sigma}(t)]\boldsymbol{\lambda}(t)}{[\boldsymbol{\lambda}^T(t)\hat{\boldsymbol{x}}(t|t-1) - (\boldsymbol{\lambda}^T(t)\hat{\boldsymbol{x}}(t|t-1))^2]} \cdot (n(t) - \boldsymbol{\lambda}^T(t)\hat{\boldsymbol{x}}(t|t-1)) \quad (3.13)$$

where $\mathrm{diag}\{\hat{\boldsymbol{x}}(t|t-1)\} = \mathrm{diag}\{\hat{x}_0(t|t-1), \cdots, \hat{x}_N(t|t-1)\}$.

We can now combine (3.8)–(3.13) to provide a realization of the optimal filter/predictor.

A straightforward calculation shows that (3.13) reduces to

$$\hat{x}_i(t|t) = \begin{cases} \dfrac{(1-\lambda(i,t))\hat{x}_i(t|t-1)}{\sum_{i=0}^{N}(1-\lambda(i,t))\hat{x}_i(t|t-1)}, & \text{if } n(t)=0 \\[2ex] \dfrac{\lambda(i,t)\hat{x}_i(t|t-1)}{\sum_{i=0}^{N}\lambda(i,t)\hat{x}_i(t|t-1)}, & \text{if } n(t)=1 \end{cases} \quad (3.14)$$

where use has been made of the easily demonstrated fact that

$$\sum_{i=0}^{N}\hat{x}_i(t|t-1) = 1.$$

Similarly, some algebraic manipulations (see [5]) give

$$\hat{\boldsymbol{x}}(t+1|t) = \begin{cases} M^T(t)\hat{x}(t|t), & \text{if } n(t)=1 \\[1ex] M^T(t)\hat{x}(t|t) + \dfrac{[Q^T(t) - M^T(t)]\hat{x}(t|t-1)}{\sum_{i=0}^{N}(1-\lambda(i,t))\hat{x}_i(t|t-1)}, & \text{if } n(t)=0 \end{cases} \quad (3.15)$$

where

$$\boldsymbol{M}^T(t) = \begin{bmatrix} 0 & 0 & & & & 0 \\ 1 & \mu(1) & 0 & 0 & & 0 \\ 0 & (1-\mu(1)) & \mu(2) & 0 & & 0 \\ 0 & 0 & (1-\mu(2)) & \mu(3) & & \\ & & 0 & \ddots & & \vdots \\ & & \ddots & \ddots & \mu(N-1) & 0 \\ 0 & 0 & \cdots & \cdots & 1-\mu(N-1) & \mu(N) \end{bmatrix}. \quad (3.16)$$

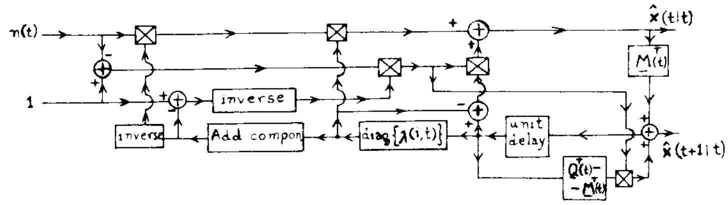


Fig. 2. Block diagram of optimal filter/predictor for Model A.

It should be apparent that (3.14) and (3.15), coupled with (3.9) for initialization, represent an algorithm for the minimum error variance filter/predictor which can be easily realized in a microprocessor, see Fig. 2. This is especially so since $\lambda(t)$, $Q^T(t)$ and $M^T(t)$ are all piecewise constant and periodic.

B. Single Detector Filter/Predictor Using Velocity and Occurrence Time

This filter/predictor is based on Model B. Thus, the basic problem is to derive the optimal filter/predictor for a system that is modeled by

$$x(t+1)=Q^T(t,\sigma)x(t)+u(t) \tag{3.17}$$

$$n^a(t)=V[\operatorname{diag}\{\lambda^a(t)\}]x(t)+\omega(t) \tag{3.18}$$

where it is easy to see that (3.18) is identical to (2.39). This is again in the form of (3.1)–(3.2). Thus, the minimum square error estimate of $x(t+1)$ given the observations

$$\mathcal{F}_t=\sigma\{n^a(1),\cdots,n^a(t)\}$$

satisfies the recursive formula given by (3.3)–(3.7). Thus, the derivation of the filter/predictor reduces to finding explicit expressions for the several conditional expectations in (3.3)–(3.7). The first problem that occurs, if one attempts to simply parallel Segall's calculations [12], is in the calculation of $\hat{f}(t|t-1)$:

$$\hat{f}(t|t-1)=E\{Q^T(t,\sigma)x(t)|n^a(t-1),n^a(t-2),\cdots\}.$$

If one refers back to the derivation of this model (and specifically σ) it is seen that σ and hence $Q^T(t,\sigma)$ is a *deterministic* function of past observations. This is, of course, only an approximation to reality. However, once the model incorporates this approximation, it is rigorously correct that

$$\begin{aligned}\hat{f}(t|t-1)&=E^{\mathcal{F}_{t-1}}\{Q^T(t,\sigma)x(t)\}\\&=Q^T(t,\sigma)E^{\mathcal{F}_{t-1}}\{x(t)\}\\&=Q^T(t,\sigma)\hat{x}(t|t-1).\end{aligned} \tag{3.19}$$

Once (3.19) is established it is easy to show that the one-step predictor is given by

$$\begin{aligned}\hat{x}(t+1|t)=Q^T(t,\sigma)\hat{x}(t|t-1)+Q^T(t,\sigma)\big([\operatorname{diag}\{\hat{x}(t|t-1)\}]\\-\hat{x}(t|t-1)\hat{x}^T(t|t-1)\big)\\\cdot[\operatorname{diag}\{\lambda^a(t)\}]V^TA^{-1}(\hat{x}(t|t-1))\big(n^a(t)\\-V[\operatorname{diag}\{\lambda^a(t)\}]\hat{x}(t|t-1)\big).\end{aligned} \tag{3.20}$$

A similar derivation gives the filter as

$$\begin{aligned}\hat{x}(t|t)=\hat{x}(t|t-1)+\big([\operatorname{diag}\{\hat{x}(t|t-1)\}]\\-\hat{x}(t|t-1)\hat{x}^T(t|t-1)\big)[\operatorname{diag}\{\lambda^a(t)\}]V^Tx\\\cdot A^{-1}(\hat{x}(t|t-1))\big(n^a(t)-V[\operatorname{diag}\{\lambda^a(t)\}]\hat{x}(t|t-1)\big)\end{aligned} \tag{3.21}$$

where we have defined

$$A(\hat{x}(t|t-1))=[\operatorname{diag}\{\hat{a}(t|t-1)\}]-\hat{a}(t|t-1)\hat{a}^T(t|t-1)$$

and

$$\hat{a}(t|t-1)=V[\operatorname{diag}\{\lambda^a(t)\}]\hat{x}(t|t-1).$$

We remark that (3.20) and (3.21) are only valid under the simplifying assumption that

$$E^{\mathcal{F}_{t-1}}\{u(t)\omega^T(t)\}=0.$$

We make this assumption even though, strictly speaking, it is only accurate when the end of the queue is not near the detector (unsaturated section). This inaccuracy for long queues is unimportant since the model breaks down for long queues anyway.

Finally, (3.20) and (3.21) appear to be too complicated to implement in a small computer. However, one can derive a formula for A^{-1} that requires the inversion of only six scalars [5]. Then, the calculations can be reduced still further to

$$\hat{x}(t+1|t)=Q^T(t,\sigma)\hat{x}(t|t) \tag{3.22}$$

and

$$\left.\begin{aligned}&\hat{x}_i(t|t)=\frac{(1-\lambda^a(i,t))\hat{x}_i(t|t-1)}{\sum_{j=0}^{N}(1-\lambda^a(j,t))\hat{x}_j(t|t-1)}, &&\text{if } n_l^a(t)=0,\ l=1,\cdots 5\\&\hat{x}_i(t|t)=\frac{v_{li}\lambda^a(i,t)\hat{x}_i(t|t-1)}{\sum_{j=0}^{N}v_{lj}\lambda^a(j,t)\hat{x}_j(t|t-1)}, &&\text{if } n_l^a(t)=1 \text{ for some } l\in[1,\cdots 5].\end{aligned}\right\} \tag{3.23}$$

By comparing these equations to (3.14) and (3.15) which describe the simpler filter/predictor it is seen that the numerical complexity of the second filter/predictor is similar to that of the first filter predictor. Thus, a microprocessor realization is feasible again.

C. Two Detector Filter/Predictor

This filter/predictor is based on Model C. Since Model C is so similar to Model B, it is quite straightforward to derive the new filter/predictor.

The only complication is that it is possible to have $n^d(t)=1$ and one of the components of $\boldsymbol{n}^a(t)$ also equal to one. However, it is reasonable to assume $n^d(t)$ and $\boldsymbol{n}^a(t)$ are uncorrelated whenever the queue is not empty. When the queue is empty there is some relation between the two measurements, which is very difficult to describe and model. Thus, we make the simplifying assumption that $\boldsymbol{n}^a(t)$ and $n^d(t)$ are uncorrelated.

Once this assumption is made, the derivation of the new filter/predictor goes through easily [5]. The most convenient way to express the result is in terms of a correction to (3.22) and (3.23). That is, (see (3.23)),

$$\left.\begin{aligned} \hat{x}_i(t|t) &= \frac{(1-\lambda^a(i,t))\hat{x}_i(t|t-1)}{\sum_{j=0}^{N}(1-\lambda^a(j,t))\hat{x}_j(t|t-1)} + fc_i, && \text{if } n_l^a(t)=0,\ l=1,\cdots 5 \\ \hat{x}_i(t|t) &= \frac{v_{li}\lambda^a(i,t)\hat{x}_i(t|t-1)}{\sum_{j=0}^{N} v_{lj}\lambda^a(j,t)\hat{x}_j(t|t-1)} + fc_i, && \text{if } n_l^a(t)=1 \text{ for some } l\in[1,\cdots 5] \end{aligned}\right\} \tag{3.24}$$

where

$$fc_i = \begin{cases} -\hat{x}_i(t|t-1) + \dfrac{(1-\lambda^d(i,t))\hat{x}_i(t|t-1)}{\sum_{j=0}^{N}(1-\lambda^d(j,t))\hat{x}_j(t|t-1)}, & n^d(t)=0 \\ -\hat{x}_i(t|t-1) + \dfrac{\lambda^d(i,t)\hat{x}_i(t|t-1)}{\sum_{i=0}^{N}\lambda^d(j,t)\hat{x}_j(t|t-1)}, & n^d(t)=1. \end{cases} \tag{3.25}$$

Similarly, (see (3.22))

$$\hat{\boldsymbol{x}}(t+1|t) = \boldsymbol{Q}^T(t,\boldsymbol{\sigma})\boldsymbol{x}(t|t) + \boldsymbol{pc} \tag{3.26}$$

where

$$\boldsymbol{pc} \begin{cases} = (\boldsymbol{\Omega}^T(t,\boldsymbol{\sigma}) - \boldsymbol{Q}^T(t,\boldsymbol{\sigma}))(\boldsymbol{fc} + \hat{\boldsymbol{x}}(t|t-1)), & \text{if } n_d(t)=1 \\ = (\boldsymbol{\Omega}^T(t,\boldsymbol{\sigma}) - \boldsymbol{Q}^T(t,\boldsymbol{\sigma}))(\boldsymbol{fc} + \hat{\boldsymbol{x}}(t|t-1)) \\ \quad + \dfrac{[\boldsymbol{Q}^T(t,\boldsymbol{\sigma}) - \boldsymbol{\Omega}^T(t,\boldsymbol{\sigma})]\hat{\boldsymbol{x}}(t|t-1)}{\sum_{j=0}^{N}(1-\lambda^d(j,t))\hat{x}_j(t|t-1)}, & \text{if } n_d(t)=0 \end{cases} \tag{3.27}$$

where $\boldsymbol{fc}$ is given by (3.25), pc denotes predictor-correction and

$$\boldsymbol{\Omega}^T(t,\boldsymbol{\sigma}) = \begin{bmatrix} \lambda_d(0) & \cdots & & 0 \\ & \cdots & (1-\lambda'(i,t,\boldsymbol{\sigma})) & \cdots \\ & & \lambda'(i,t,\boldsymbol{\sigma}) & \cdots \\ 0 & & & \cdots \end{bmatrix}. \tag{3.28}$$

This shows that the filter/predictor can be implemented modularly by adding a processor module with the additional detector.

IV. Test and Evaluation

A. Introduction

The ultimate test of an algorithm for estimating traffic flow parameters is to include it in the software for an operating computer-controlled traffic network. If, under those circumstances, the filter/predictor algorithm performs well then it is a good algorithm regardless of its performance on any other tests. Unfortunately, we do not have an operational computer-controlled traffic network for use as a test. However, the Urban Traffic Control System Number One (UTCS-1) simulation model provides a reasonable and comparatively inexpensive means to test our filter/predictors.

The UTCS-1 simulation model is a very detailed simulation of urban traffic, developed under the auspices of the Federal Highway Administration. It is believed to be a fairly accurate simulation of urban traffic [16]. Furthermore, it is based on a model of traffic flow that is very different from any of our models [17]. For our tests, we simulated two simple urban networks, of which only the simplest is included in this paper. This network is shown in Fig. 3, where all streets are single-lane streets on which traffic flows in the arrow direction. The rectangles represent detectors which give occurrence time and correct velocity for each vehicle crossing the detector.

The following assumptions (actually inputs to the simulation) are held constant throughout all the tests:

1) All streets are 500 ft long from node to node and have zero grade.

2) The detectors are 210 ft from the downstream stopline.

3) The traffic signals are:

- Left-most intersection:
  - 80 s cycle time;
  - 40 s red—40 s green—0 s amber.
- Center intersection:
  - 80 s cycle time;
  - 40 s red—40 s green—0 s amber;
  - 20 s offset from signal at node 5.
- Right-most intersection:
  - 80 s cycle time;
  - 40 s red—40 s green—0 s amber;
  - 40 s offset from signal at node 5.

Other parameters of the simulation are given in the report [5] but are not essential in the following.

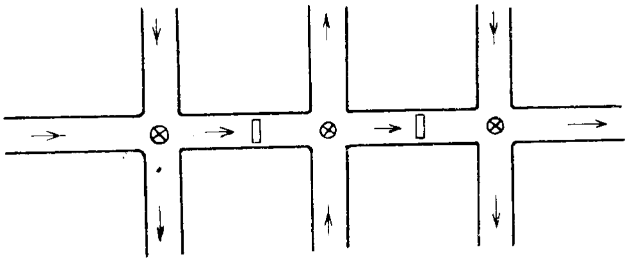


Fig. 3. Street configuration Test Network #I. Nodes are 5, 6, 7 in the direction of flow.

Four tests were run using the above network. *Test 1* corresponded to "moderate" traffic flow on which the traffic signals dominate the traffic. *Test 2* is again moderate traffic but is more "random" than Test 1. *Test 3* corresponds to moderate to heavy traffic while *Test 4* corresponds to heavy traffic.

B. Filter/Predictor Using Occurrence Times Only

The first queue estimation scheme that we tested was based on Model *A*. This filter/predictor, hereinafter denoted F/P I, depends only on two functions, $\lambda(i,t)$ and $\mu(i,t)$, where i expresses the dependence on queue length and t the dependence on time. For simplicity, we eliminated the dependence on i in the actual implementation of F/P I. Thus, F/P I depends only on *three* parameters:

$$\lambda(i,t)=\begin{cases}\lambda_g, & \text{upstream traffic signal green} \\ & i=0,1,\cdots,N-1 \\ \lambda_r, & \text{upstream traffic signal red} \\ & i=0,1,\cdots,N-1\end{cases}$$

$$\mu(i,t)=\begin{cases}\mu, & \text{downstream traffic signal} \\ & \text{green for 5 s or more} \qquad i=1,2,\cdots,N \\ 0, & \text{otherwise} \qquad i=0,1,2,\cdots,N.\end{cases}$$

In all of the tests, the detector was 210 ft from the stopline so $N=10$ was the value used. It will be seen that, once or twice, there were actually 11 vehicles in the segment between detector and stopline. The value for λ_g was estimated by averaging the number of vehicles crossing the detector during the upstream green over the upstream green time. λ_r was estimated similarly. The results are fairly insensitive to the values for λ_g and λ_r. On the other hand, μ is very important. Thus, several different values of μ were tried for each simulation. We expect that, because of this sensitivity to μ, it will be possible to use an adaptive procedure to compute it in an actual implementation of F/P I. The performance evaluation needed for adaptivity would be based on data from downstream detectors and the correlation between queues and downstream platoons discussed at length in [13]. We have not had sufficient time to do this as yet.

This dependence of μ is well illustrated in a number of our tests. For example, see Figs. 4 and 5. Both figures refer to tests using F/P I and Test 3. In both figures F/P I is implemented using $\lambda_g=0.25$ and $\lambda_r=0.08$. In Fig. 4, $\mu=0.50$ and the estimate is off by slightly more than two vehicles from $t=360$ to almost 400. There are even larger errors at approximately 260 and 340 s. The errors at 260 and 340 s are not as serious as the other error because they occur during rapidly changing queues and it is almost impossible to track such rapid changes accurately. However, in Fig. 5, $\mu=0.45$ and the estimate is almost always within one vehicle of the correct value. There are some brief large errors but these occur during rapid changes and the errors are quickly eliminated.

In Figs. 4 and 5, the estimate given is the minimum error variance estimate or, equivalently, the conditional mean. In fact, the estimation procedure used gives much more information than this. This is demonstrated in Table I which corresponds to Fig. 5 exactly. Notice that the conditional probability of a queue of one vehicle, two vehicles, etc. is given. To see what this means, take $T=50$ s. Note that $\Pr[x_1(t)=1|\mathcal{F}_t]=0.40$ and $\Pr[x_2(t)=1|\mathcal{F}_t]=0.52$. In fact, there is one vehicle in the queue and the estimator assigns almost all the probability to there being either one or two vehicles in the queue. Thus, although the conditional mean estimate is about 0.7 vehicles too large, the estimator "knows" that there are one or two vehicles.

Similarly, at $T=260$, the estimator assigns significant probability to every queue length from one to six vehicles. The actual number is either five or six and the conditional mean is 3.8. The point is that the filter/predictor "tells" us that it is not very sure of itself. This information is available and may be used to greatly improve the control algorithm.

The test involving heavy traffic, Test 4, is shown in Fig. 6. This figure shows fairly good performance of the filter/predictor. The steep increases in the estimate that one sees, for example, in Fig. 6 at $t=210$–240 s demonstrate why F/P I performs as it does. If no vehicle arrives during a time that F/P I expects the queue to grow, then F/P I assumes this is because the queue has extended to the detector and no more vehicles can cross the detector. Thus, whenever there is a "long" gap between arriving vehicles, the estimated queue length increases. This occasionally (as at $t=240$ s in Fig. 6) causes large errors but usually it eliminates errors. It should also be noted that, when a vehicle crosses the detector after a long delay the estimated queue decreases instantaneously and then increases again. This reflects the fact that, if a vehicle arrives, the queue could not have been ten prior to the arrival. This causes $\hat{x}_{10}(t)=\Pr[\text{queue}=10|\mathcal{F}_t]$ to drop to zero. A moment later, probability increases from zero because one more vehicle, the new arrival, has been added to the queue.

In the heavy traffic case, the actual queue extends beyond the detector. Thus, we would have to use a similar estimator to estimate the number of vehicles that are "stored" upstream from the detector. We have not done this as yet.

In addition, we compared the performance of F/P I, with ASCOT [14], [18], one of the "queue" estimators currently in use. ASCOT is regarded as one of the best single-detector queue estimators [15]. However, it uses the velocity data from the detector, it gives only a single number as

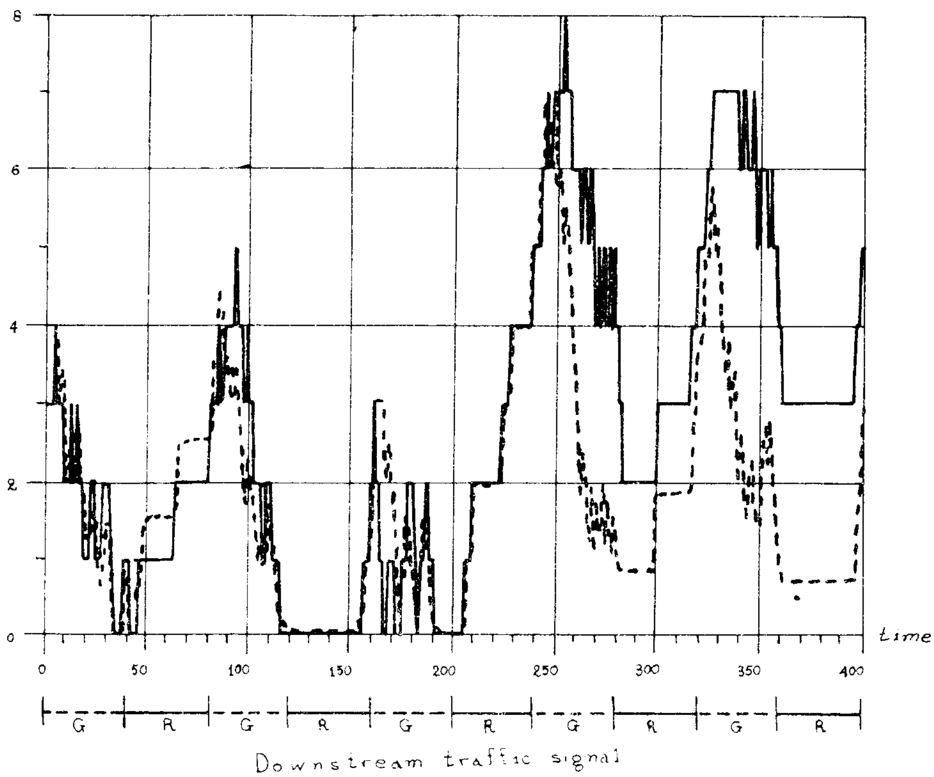


Fig. 4. Performance of F/P I. The data are from Test 3, link 5–6. The dotted line indicates the minimum error variance estimate of the queue. The solid line is the true value. The parameters of F/P I are $\lambda_g=0.25$, $\lambda_r=0.08$, $\mu=0.50$.

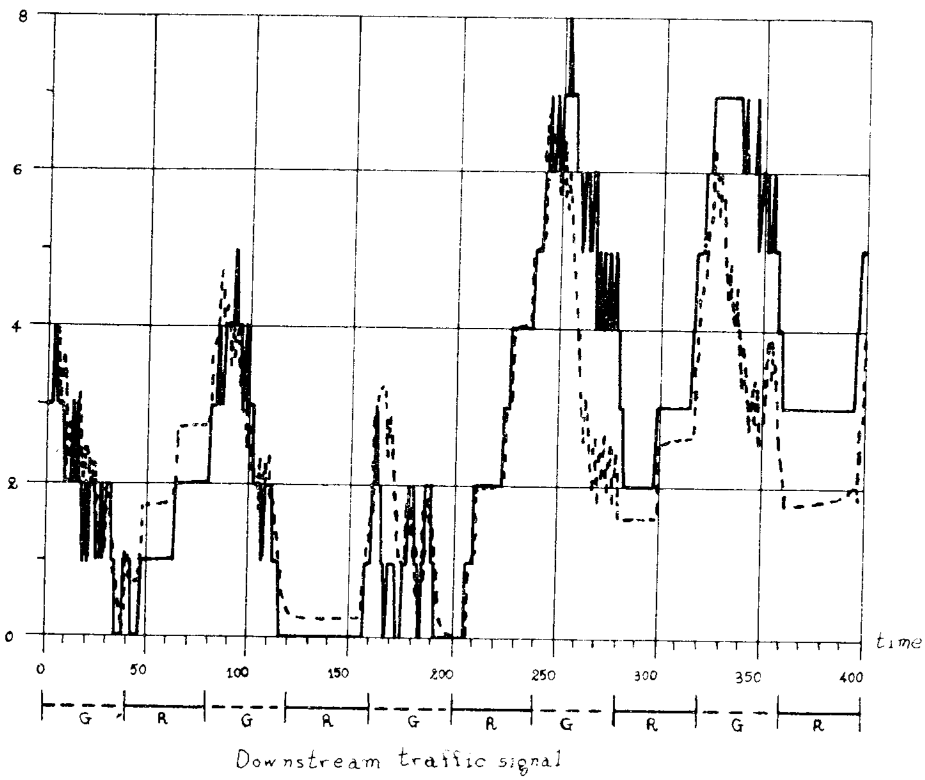


Fig. 5. Performance of F/P I. The data are from Test 3, link 5–6. The dotted line indicates the minimum error variance estimate of the queue. The solid line is the true value. The parameters of F/P I are $\lambda_g=0.25$, $\lambda_r=0.08$, $\mu=0.45$.

TABLE I
PERFORMANCE OF F/P I. THE DATA ARE FROM TEST 3, LINK 5–6. THE PARAMETERS OF F/P I ARE $\lambda_g = 0.25$, $\lambda_r = 0.08$, AND $\mu = 0.45$. SEE FIG. 5 FOR THE ACTUAL QUEUE.

t	n(t)	$\hat{x}_0(t)$	$\hat{x}_1(t)$	$\hat{x}_2(t)$	$\hat{x}_3(t)$	$\hat{x}_4(t)$	$\hat{x}_5(t)$	$\hat{x}_6(t)$	$\hat{x}_7(t)$	$\hat{x}_8(t)$	$\hat{x}_9(t)$	$\hat{x}_{10}(t)$	$ML(\hat{\underline{x}}(t))$	$E(\hat{\underline{x}}(t))$
38	0	0.83	0.09	0.05	0.02	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0	0.3
39	1	0.87	0.07	0.04	0.02	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0	0.2
40	0	0.00	0.90	0.05	0.03	0.01	0.01	0.00	0.00	0.00	0.00	0.00	1	1.2
41	0	0.40	0.52	0.04	0.02	0.01	0.00	0.00	0.00	0.00	0.00	0.00	1	0.7
42	0	0.40	0.52	0.04	0.02	0.01	0.00	0.00	0.00	0.00	0.00	0.00	1	0.7
43	0	0.40	0.52	0.04	0.02	0.01	0.00	0.00	0.00	0.00	0.00	0.00	1	0.7
44	0	0.40	0.52	0.04	0.02	0.01	0.00	0.00	0.00	0.00	0.00	0.00	1	0.7
45	0	0.40	0.52	0.04	0.02	0.01	0.00	0.00	0.00	0.00	0.00	0.00	1	0.7
46	0	0.40	0.52	0.04	0.02	0.01	0.00	0.00	0.00	0.00	0.00	0.00	1	0.7
47	1	0.40	0.52	0.04	0.02	0.01	0.00	0.00	0.00	0.00	0.00	0.00	1	0.7
48	0	0.00	0.40	0.52	0.04	0.02	0.01	0.00	0.00	0.00	0.00	0.00	2	1.7
49	0	0.00	0.40	0.52	0.04	0.02	0.01	0.00	0.00	0.00	0.00	0.00	2	1.7
50	0	0.00	0.40	0.52	0.04	0.02	0.01	0.00	0.00	0.00	0.00	0.00	2	1.7
51	0	0.00	0.40	0.52	0.04	0.02	0.01	0.00	0.00	0.00	0.00	0.00	2	1.7
52	0	0.00	0.40	0.52	0.04	0.02	0.01	0.00	0.00	0.00	0.00	0.00	2	1.7
53	0	0.00	0.40	0.52	0.04	0.02	0.01	0.00	0.00	0.00	0.00	0.00	2	1.7
54	0	0.00	0.40	0.52	0.04	0.02	0.01	0.00	0.00	0.00	0.00	0.00	2	1.7
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
254	1	0.00	0.00	0.02	0.07	0.17	0.26	0.26	0.16	0.06	0.01	0.00	5	5.4
255	0	0.00	0.00	0.01	0.04	0.11	0.21	0.26	0.21	0.11	0.04	0.01	6	6.0
256	0	0.00	0.00	0.02	0.07	0.16	0.23	0.24	0.17	0.08	0.02	0.01	6	5.5
257	0	0.00	0.01	0.05	0.11	0.19	0.23	0.21	0.13	0.05	0.01	0.00	5	5.1
258	0	0.01	0.03	0.08	0.15	0.21	0.22	0.17	0.09	0.04	0.01	0.00	5	4.6
259	0	0.02	0.05	0.11	0.18	0.21	0.20	0.14	0.07	0.02	0.01	0.00	4	4.2
260	0	0.04	0.08	0.14	0.19	0.21	0.17	0.10	0.05	0.02	0.00	0.00	4	3.8
261	0	0.08	0.10	0.16	0.20	0.19	0.14	0.08	0.03	0.01	0.00	0.00	3	3.3
262	0	0.12	0.13	0.18	0.20	0.17	0.11	0.06	0.02	0.01	0.00	0.00	3	2.9
263	1	0.18	0.15	0.19	0.18	0.14	0.09	0.04	0.02	0.01	0.00	0.00	2	2.5
264	0	0.00	0.25	0.17	0.18	0.16	0.12	0.07	0.03	0.01	0.00	0.00	1	3.2
265	0	0.11	0.21	0.18	0.18	0.14	0.10	0.05	0.02	0.01	0.00	0.00	1	2.7

Note: Characters with underbars appear boldface in text.

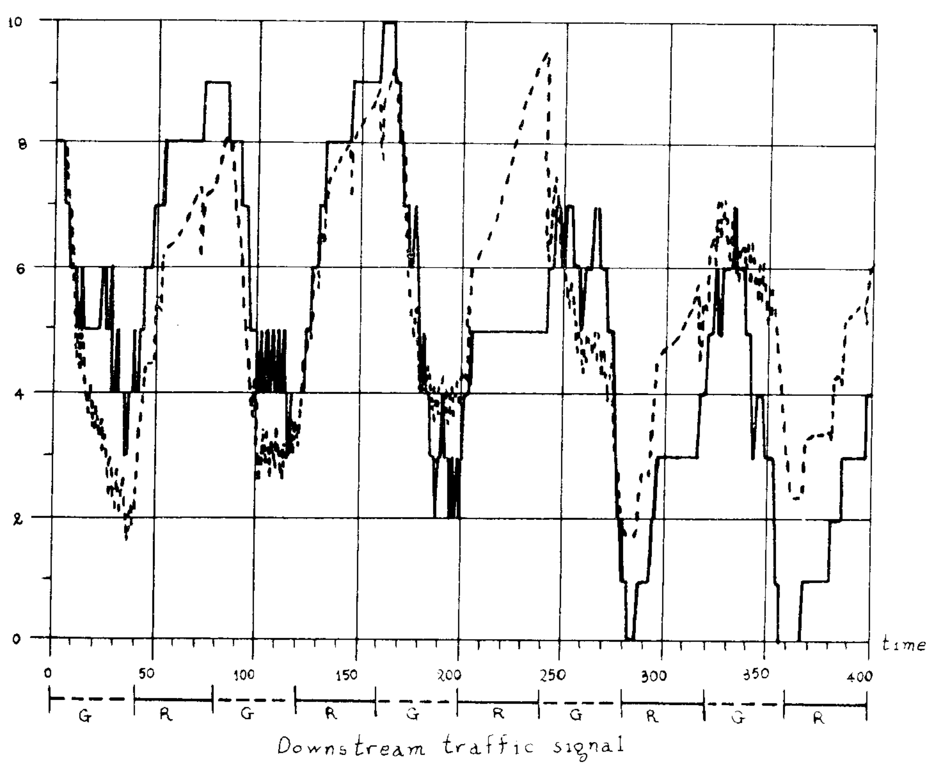


Fig. 6. Performance of F/P I. The data are from Test 4, link 6–7. The dotted line indicates the minimum error variance estimate of the queue. The solid line is the true value. The parameters of F/P I are $\lambda_g = 0.23$, $\lambda_r = 0.12$, $\mu = 0.45$.

TABLE II
COMPARISON OF ASCOT WITH F/P I. DATA ARE FROM TEST 3, LINK 5–6.

Time	Actual Queue	ASCOT Estimate	F/P I Estimate
80	2	3	2.7
160	2	2	2
240	5	5	5
320	5	7	4.5
400	5	7	3.8

TABLE III
COMPARISON OF ASCOT WITH F/P I. DATA ARE FROM TEST 4, LINK 5–6.

Time	Actual Queue	ASCOT Estimate	F/P I Estimate
80	8	3	5.8
160	9	7	9
240	10	7	9.5
320	10	12	9.5
400	7	7	9.5

its estimate and this estimate is given only at the instant of red to green transition of the traffic signal. Thus, F/P I can give much more information, based on less data but more computation than ASCOT. The comparisons are summarized in the Tables II and III.

C. Single Detector Filter Predictor Using Velocity and Occurrence Time

The second queue estimator that we tested was based on Model B. This is a much more complex model than Model A. However, the filter/predictor which we denote by F/P II, depends on the three functions $\lambda^t(i,t,\sigma)$, $\lambda^h(i,t)$, and $v_{ji}\lambda^a(i,t)$. Of course, σ is also a function of the input data and this also affects F/P II. For simplicity, we again eliminated the dependence on i of $\lambda^h(i,t)$ and $\lambda^a(i,t)$ in the actual implementation of F/P II.

Thus, as before:

$$\lambda^a(i,t)=\begin{cases}\lambda_g, & \text{upstream traffic signal green} \\ & i=0,1,\cdots,N-1 \\ \lambda_r, & \text{upstream traffic signal red} \\ & i=0,1,\cdots,N-1\end{cases}$$

$$\lambda^h(i,t)=\begin{cases}\mu, & \text{downstream traffic signal green for 5 s or more} \quad i=1,2,\cdots,N \\ 0, & \text{otherwise} \quad i=0,1,2,\cdots,N.\end{cases}$$

All of the other assumptions and parameters of Model B and F/P II are described in Section II of this paper. These values were fixed at the values given in Section II throughout these tests. This is because we believe it is reasonable to adjust two or three filter/predictor parameters as a function of time-of-day or adaptively. Furthermore, this parameter adjustment can greatly improve the performance of any filter/predictor.

We remark that the parameters λ_g, λ_r, and μ were chosen by the same technique used for F/P I. That is, λ_g and λ_r were determined by averaging the number of vehicles crossing the detector during the appropriate time period. Several different values of μ were tried for each simulation. In this case, our definition of the queue corresponds quite closely to the definition used in the UTCS-1 simulation.

Fig. 7 shows the performance of F/P II on link 6–7 during Test 1. There is an error of almost two vehicles at $t=240$ s. This is caused by a vehicle which "runs" the red light at $t=204$ s. This sort of thing certainly happens in real traffic but is bound to cause filtering and prediction errors. It should be noted that the filter/predictor recovers from the error almost immediately. The other large error occurs at $t=160$ s. This is caused by a combination of two factors. A vehicle just beats the signal transition from green to red at $t=120$ s. The filter/predictor assigns some probability to this vehicle having been caught by the light. This causes the gradual increase in queue estimate from $t=120$ to $t=158$ s. The jump at $t=160$ s is caused by a vehicle that crosses the detector 5 s before the signal changes to green. Thus, this is a "reasonable" error in that a similar traffic situation would result in a queue of 1–2 vehicles with some probability. The one vehicle error at approximately $t=325$ s is not very serious and seems to be due to a problem in the simulation. In this simulation, a vehicle crosses the detector at $t=320$ s with a velocity of 34 ft/s. According to the simulation, that vehicle departs 14 s later without ever joining the queue. This is quite unlikely. Thus, F/P II performs extremely well in Test 1.

We did many other tests on F/P II with results that are similar to the above. To summarize these results, F/P II performs very well in light to moderate traffic and should not be used in heavy traffic. In this case we were unable to get ASCOT to work in a satisfactory manner so we have not compared F/P II with ASCOT. We would expect the comparison to be similar to that for F/P I and ASCOT.

We did very little adjustment of the filter/predictor parameters for this model. In a few cases, two values of λ^d were tried. The figures show the results for the best choice of λ^d. In every case, we set $\lambda^t(0,t)=0$ when the downstream traffic signal was green. This is slightly different from the value given in the model. However, the change results in an improvement in performance and so, should be incorporated in any implementation of F/P II.

D. Two-Detector Filter Predictor

The third queue estimator that we tested was based on Model C and is denoted F/P III. Since we assume $\lambda^d(i,t)=\lambda^h(i,t)$ there are not more parameters in F/P III than there were in F/P II. In fact, the parameter values used in the tests of F/P III are identical to those used in F/P II with two exceptions. In F/P III

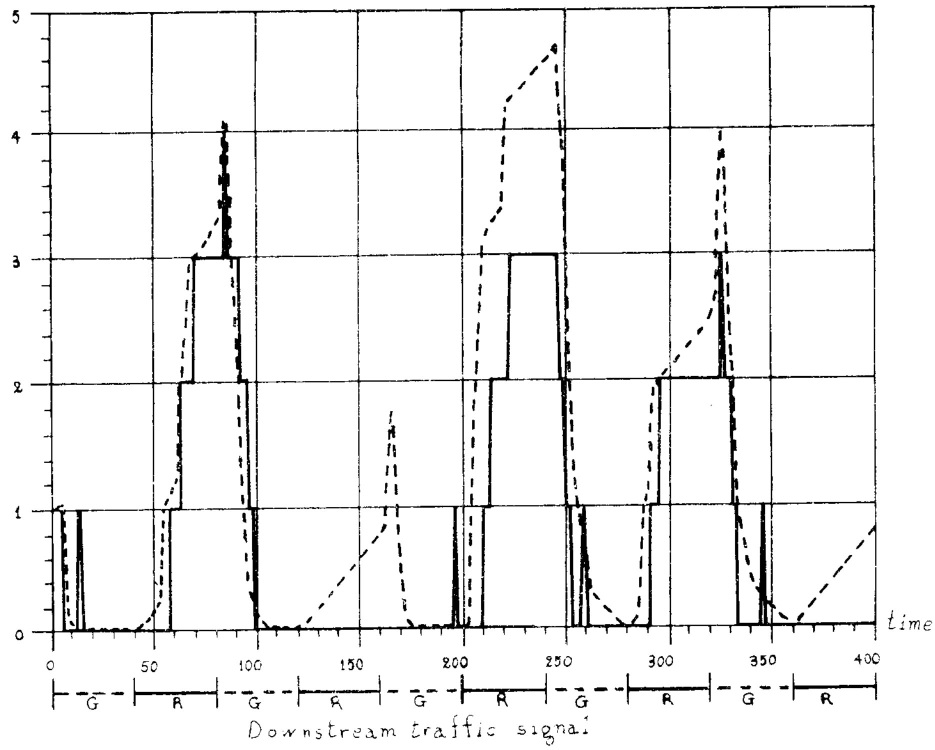


Fig. 7. Performance of F/P II. The data are from Test 1, link 6–7. The dotted line indicates the minimum error variance estimate of the queue. The solid line is the true value. The parameters of F/P II are $\lambda_g = 0.22$, $\lambda_r = 0.08$, $\mu = 0.75$.

$$\lambda^d(0,t) = \lambda_g^a, \quad \text{when the downstream traffic signal is green}$$

$$\lambda^d(i,t) = \begin{cases} 0.02, & i = 0, 1, \cdots, N \text{ when the downstream traffic signal is red} \\ \lambda^h(i,t), & i = 1, 2, \cdots, N \text{ when the downstream traffic signal is green.} \end{cases}$$

These two changes reflect the facts that 1) departures occur even when the queue is empty at approximately the free flow rate, and 2) departures occur even though the traffic signal is red. It is necessary to model these departures more carefully for F/P III because the filter/predictor can be badly "confused" when its observation, $n^d(t)$, is allegedly impossible. That is, if $\lambda^d(i,t) = 0$ when the traffic signal is red and $n^d(t) = 1$ then F/P III concludes that an impossible event has occurred.

In any case, F/P III then depends on three variable parameters λ_g, λ_r, and μ. These parameters are identical to the parameters of F/P II and they are given the same values they were given in the tests of F/P II.

Some test results are exhibited in Fig. 8. Since the data input to F/P III is identical to the data input to F/P II it is easiest to simply re-read the discussion of the results for F/P II. There are two points that should be noted:

1) F/P III appears to perform slightly *worse* than F/P II;

2) F/P III is much more oscillatory than F/P II.

The apparent degradation in performance is caused by the "rolling queue" effect that we described earlier. In many of these simulations, a platoon of traffic arrives at the queuing area just as the original queue departs. When this happens, the simulation usually reports that the queue is empty. However, there are usually several vehicles that have been forced to decelerate or, in effect, to queue even though they do not stop. The filter/predictor tends to include these vehicles in its estimate of the queue. In practical applications, this may well be better than an exact estimate of the number of stopped vehicles.

The oscillations in the output from F/P III are unimportant from a practical viewpoint since a simple low-pass filter will remove them. They are caused by the fact that the filter tends to increase its estimate of the queue at the time a departure occurs. This is because departures are more probable when there is a queue. Immediately after the departure the filter decreases its estimate of the queue by approximately one vehicle. This is for the obvious reason that a vehicle is known to have departed.

We believe that these simulation results have implications concerning the value of different measurements, which parameters of traffic are easiest to estimate, as well as the potential value of these filter/predictors.

V. Conclusions

At first glance it appears that F/P I, the simplest estimator, performs best. It should be noted that this is not our intended conclusion. F/P I estimates the number of vehicles between the detector and the stopline. F/P II and F/P III estimate the number of stopped vehicles. It is much harder to correctly estimate the number of stopped vehicles. It is also true that F/P II and F/P III depend

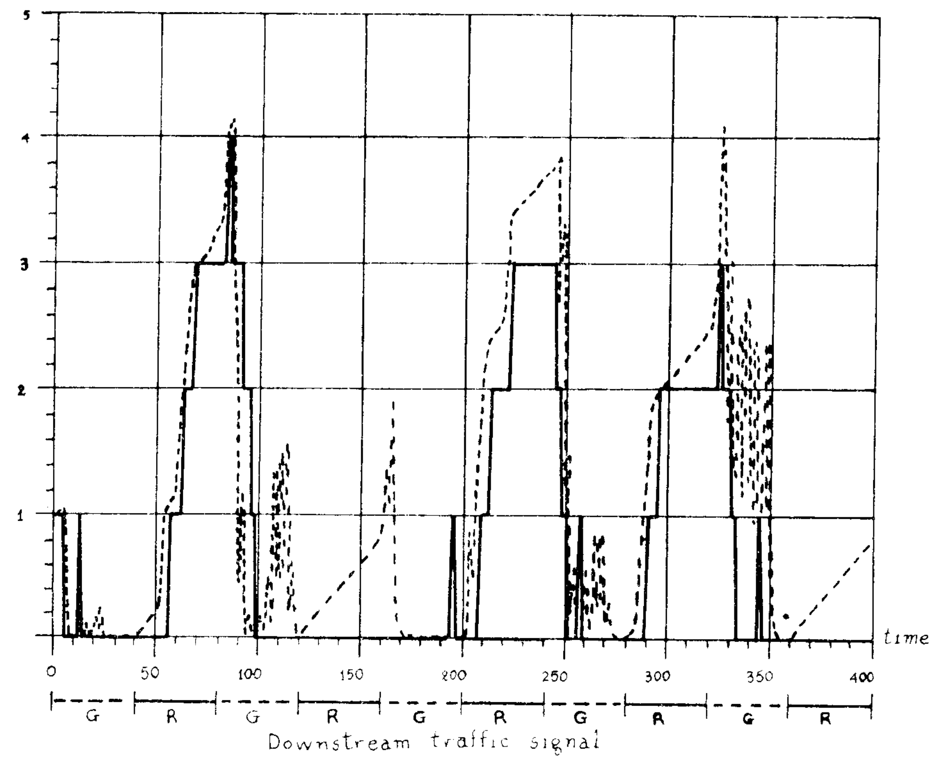


Fig. 8. Performance of F/P III. The data are from Test 1, link 6–7. The dotted line indicates the minimum error variance estimate of the queue. The solid line is the true value. The parameters of F/P III are $\lambda_g = 0.22$, $\lambda_r = 0.08$, $\mu = 0.75$.

on several more parameters than F/P I, and no effort was devoted to optimizing any of these parameters other than λ_g, λ_r, and μ. It was felt that this was a more realistic test since in actual practice tuning many parameters is too expensive to be practical.

In the light of the above caveat, all three of the queue estimators described in this paper appear to perform well. Of course, their performance on a real street might not be as good. However, the fundamental test of these estimators is to incorporate them in a closed-loop (or traffic-responsive) traffic control system. If the control based on these queue estimates is an improvement over current controls, then these estimators are useful. Our current research is devoted to the closed-loop control of urban traffic.

Another important, and not fully explored question concerning these estimators is how to make them adaptive. Some preliminary results along this line are presented along with another estimator for a relevant urban traffic parameter in a companion paper [13].

References

[1] C. R. Stockfisch, "Selecting digital computer systems", Rep. U.S. Dep. of Transportation, Federal Highway Admin., Offices Res. Dev., Dec. 1972.

[2] E. A. Torrero, "Unjamming traffic congestion," *IEEE Spectrum*, pp. 77–80, Nov. 1977.

[3] *Urban Traffic Control and Bus Priority System, Vol. 1, Design and Installation*, Prepared by Sperry Systems Management Division, Sperry Rand Corp., for Federal Highway Administration, Office Res. Dev. under Contract FH-11-7605, 1972.

[4] P. J. Tarnoff, "The results of FHWA urban traffic control research: An interim report," *Traffic Engrg.*, vol. 45, pp. 27–35, Apr. 1975.

[5] J. S. Baras, W. S. Levine, A. J. Dorsey, and T. L. Lin, "Advanced filtering and prediction software for urban traffic control systems," Dep. Transportation, Contract DOT-05-60134, Final Rep., Dec. 1977.

[6] P. Bremaud, "A Martingale approach to point processes," Electronics Res. Lab., Univ. California, Berkeley, Memo M 345, 1972.

[7] R. Boel, P. Varaiya, and E. Wong, "Martingales on jump processes I: Representation results," *SIAM J. Contr.*, vol. 13, no. 5, pp. 999–1021, Aug. 1975.

[8] ——, "Martingales on jump processes II: Applications," *SIAM J. Contr.*, vol. 13, no. 5, pp. 1022–1061, Aug. 1975.

[9] D. Snyder, *Random Point Processes*. New York: Wiley, 1975.

[10] A. Segall, M. H. A. Davis, and T. Kailath, "Nonlinear filtering with counting observations," *IEEE Trans. Inform. Theory*, vol. IT-21, pp. 125–134, Mar. 1975.

[11] M. H. A. Davis, "The representation of Martingales of jump processes," *SIAM J. Contr. Optimiz.*, vol. 14, no. 4, pp. 623–638, 1976.

[12] A. Segall, "Recursive estimation from discrete time point processes," *IEEE Trans. Inform. Theory*, vol. IT-22, pp. 422–431, July 1976.

[13] J. S. Baras, W. S. Levine, and A. S. Dorsey, "Estimation of traffic platoon structure from headway statistics", to be published.

[14] D. W. Ross *et al.*, "Improved control logic for use with computer controlled traffic—Interim report for NCHRP project 3-18(1)," Stanford Res. Inst., Stanford, CA, prepared for the Highway Research Board, Washington, DC, July 1972.

[15] C. J. MacGowan, "A single detector queue determination algorithm," M.S. thesis, Dep. Civil Eng., Univ. Maryland, College Park, MD, 1975.

[16] *Network Flow Simulation for Urban Traffic Control System—Phase II, Vol. 5 Applications Manual for UTCS-1 Network Simulation Model*, prepared by Peat, Marwick, Mitchell & Co. and KLD Associates, Inc. for Federal Highway Administration, Office Res. Dev. under Contract DOT-FH-11-7885, 1974.

[17] *Network Flow Simulation for Urban Traffic Control System—Phase II, Vol. 4 User's Manual for UTCS-1 Network Simulation Model*, prepared by E. Lieberman, D. Wicks and J. Woo for Federal Highway Administration Office Res. Dev. under Contract DOT-FH-11-8502, 1977.

[18] D. W. Ross, "Single detector queue measurement for computerized traffic systems," DARO Associates Inc., Sunnyvale, CA (unpublished), 1975.

John S. Baras (S'73-M'73) was born in Piraeus, Greece, on March 13, 1948. He received the Electrical Engineeering Diploma from the National Technical University of Athens, Athens, Greece in 1970, and the M.S. and Ph.D. degrees in applied mathematics from Harvard University, Cambridge, MA, in 1971 and 1973, respectively.

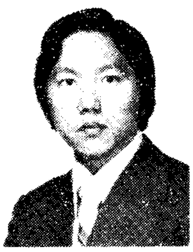
Since 1973 he has been with the Department of Electrical Engineering, University of Maryland, College Park, where he is an Associate Professor in the Control Systems group. His current research interests include distributed parameter systems, quantum estimation, stochastic control of queuing systems, and applications.

Dr. Baras is a member of Sigma Xi, the American Mathematical Society, and the Society for Industrial and Applied Mathematics.

William S. Levine (S'66-M'68) was born in Brooklyn, NY, on November 19, 1941. He received the B.S., M.S., and Ph.D. degrees in Electrical Engineering from the Massachusetts Institute of Technology, Cambridge, in 1962, 1965, and 1969, respectively.

From 1962 to 1964 he was employed by the Data Technology, Inc. as a design engineer. From 1964 to 1969 he was a research Assistant in the Department of Electrical Engineering at M.I.T. with the exception of one ten month period as a Teaching Assistant. In 1969 he joined the faculty of the Department of Electrical Engineering at the University of Maryland, College Park, where he is currently an Associate Professor. His current research interests include the theory and application of optimal control and estimation to problems in traffic control and animal locomotion.

Dr. Levine is a member of Tau Beta Pi, Eta Kappa Nu, and Sigma Xi. He is also a member of the Society for Neuroscience.



Tahsin L. Lin was born in Taiwan, Republic of China, on February 3, 1950. He received the B.S.E.E. degree from National Cheng Kung University, Tainan, Taiwan, in 1971, and the M.S. degree from Southern Illinois University, Carbondale, IL, in 1976. He is currently a candidate for the Ph.D. degree in the Electrical Engineering Department, University of Maryland, College Park.

Mr. Lin's main interests are in the area of signaling and decentralized control of large queuing networks.