Optimization of Traffic Signal Settings by Mixed-Integer Linear Programming*

Part I: The Network Coordination Problem

NATHAN H. GARTNER,
JOHN D. C. LITTLE
and
HENRY GABBAY

Massachusetts Institute of Technology, Cambridge, Massachusetts

Setting traffic signals in a signal-controlled street network involves the determination of cycle time, splits of green time, and offsets. Part I of this paper considers the network coordination problem, i.e., given a common cycle time and green splits at each intersection, determine offsets for all signals. Part II considers the more general synchronization problem, i.e., determine simultaneously all the control variables for the network including offsets, splits, and cycle time. In Part I, a link performance function is developed to express the loss incurred by platoons traveling through a signal-controlled intersection as a function of link offset. Integer variables enter the formulation because of the periodicity of the traffic lights: The algebraic sum of the offsets around any closed loop of the network must equal an integral number of cycle times. The optimization problem is formulated as a mixed-integer linear program and a test network is solved by branch-and-bound techniques using IBM's MPSX package.

Traffic signals, when appropriately set, help smooth the flow of traffic through a street network, thereby reducing delays and stoppages. The

* This research was supported (in part) by KLD Associates under DOT Contract FH-11-7924 and (in part) by ARO Contract DAHCO4-73-COO32.

signal control variables affecting performance are the signal period or *cycle time*, the fraction of a cycle or *green split* allocated to each competing traffic stream at an intersection, and the relative phases or *offsets* among signals.

The goal of our work is to develop a computationally feasible method for simultaneously optimizing all the signal control variables of a network. This has not previously been achieved. Where a systematic procedure has been used to set the signal variables, the usual approach has been to do so in three stages. First, a common cycle time for the network is selected. The signals at the different parts of the network can then be said to be *synchronized*. This is widely considered to enhance system performance provided there is a noticeable platooning effect on the street links connecting the signals. Since the capacity of a signalized intersection is a function of the cycle length, the common cycle is usually determined according to the requirements of the most heavily loaded intersection.

The second stage in setting signals is to calculate green splits. This is done separately for each signal by apportioning green times for conflicting streams in proportion to their representative values of flow/capacity.

The third stage is to *coordinate* the signals by establishing a set of offsets which determine relative timing among signals. Only at this stage is a network-wide optimization commonly employed. Since the three signal-setting stages are not really independent, some methods iterate among them seeking to improve the solution.

The most frequently used coordination method is arterial progression, in which signals along arterials are assigned offsets to maximize *bandwidth*. By selecting arterials that do not form closed loops, one can also coordinate a network in this manner. In the earliest form of the method, green bands are constructed by a graphical trial-and-error procedure.[1,2] In 1964, MORGAN AND LITTLE[3] presented a rigorous computational algorithm for the process. LITTLE[4] has since formulated the problem for optimization by mixed-integer linear programming and has generalized the formulation to networks. However, it has long been recognized that bandwidth maximizes a geometric quantity that may or may not be appropriate for the actual traffic flows on a specific street. As an extreme example of the potential artificialities of the method, it is possible under certain circumstances, such as a large number of signals or adverse combinations of block lengths and travel speeds, to be unable to provide any bandwidth at all, even though some form of coordination will be beneficial.

In recent years several coordination methods have been developed that seek to take explicit account of traffic performance and attempt to optimize it. In North America, SIGRID[5] and SIGOP[6] fall in this category. These methods hypothesize a link performance function that is quadratic in the link offset. This function is intended to represent delays and stops incurred by traffic on the link. A search procedure then tries to set the offsets to minimize the sum of the measures of delays and stops over all the links of the network. However, since networks with loops contain physical constraints with integer variables that are not completely considered, global optimality cannot be guaranteed.

In Britain, as part of the Glasgow study in area traffic control,[7] the Road Research Laboratory devised a series-parallel combination procedure for setting offsets on the basis of link performance functions. Each function expresses delay in terms of offset and is obtained by a simple model of traffic behavior on the link. The method came to be known as the Combination Method and was subsequently extended to nonseries-parallel networks by ALLSOP[8] and GARTNER[9] using dynamic programming approaches. Though this method can determine, in principle, globally optimal offsets, its computational feasibility depends on the topological form of the network and may break down from having to consider too many state variables. Later, TRANSYT[10] was developed at the Road Research Laboratory. In this method performance is evaluated by a more detailed simulation of traffic flow throughout the network. Optimization is performed by a one-at-a-time variable search that requires a network simulation pass at each stop.

Although these methods have been found to achieve significant improvements in traffic performance, they cannot guarantee a mathematically optimal solution over all the network decision variables in a reasonable amount of computer time. There remains, therefore, an opportunity for further improvements in signal settings if a method can be devised that simultaneously optimizes all the network variables.

The approach in the present work has been first to develop a traffic model and performance measure and then to bring contemporary optimization techniques to bear on it. The principal technique used is mixed-integer linear programming, for which effective computer programs have recently become available[11] and for which better ones are likely to be developed. For reasons of methodology and clarity, this work is presented in two parts. Part I considers only the network *coordination* problem, i.e., for a given common cycle time and set of green splits, optimize offsets for the network. Part II develops the traffic model further[12] to consider the full network *synchronization* problem, i.e., to optimize all the decision variables simultaneously including offsets, greentime splits, and cycle time.

1. SYSTEM DEFINITION

THE TRAFFIC NETWORK is represented by nodes and links. The nodes are the signalized intersections. The links represent sections of street that carry traffic in one direction between two intersections. In this section, we set up the coordination problem as a mixed-integer nonlinear program with the signal offsets as decision variables.

Network Definitions:

$G[N,L]$ = the graph of the network,
$N = \{i\}$ = the set of nodes of G,
(i,j) = the link from node i to node j,
$L = \{(i,j)\}$ = the set of links of G,
l = a set of links that form a loop (undirected circuit) in G,
$\mathcal{L} = \{l\}$ = the set of all loops in G.

Node Definitions:

S_j = traffic signal at node j,
r_{ij} = effective red time at S_j facing (i,j),
g_{ij} = effective green time at S_j facing (i,j),
$\psi_j(AB)$ = intra-node offset at S_j, the time from the beginning of green on phase A to the beginning of green on phase B,
C = cycle time.

The relations between physical and effective signal timings are illustrated in Fig. 1 for a single two-phase signal, following the basic model of traffic signal operation used by CLAYTON[13] and others.[14-16] A phase faces some incoming link and ordinarily feeds some primary outgoing link. Because of turn traffic, a phase may provide secondary flow for other outgoing links. Phases are denoted by single letters, A, B, etc., which are replaced by letter pairs (i,j), (j,k), etc., when we wish to work with them on their assigned links or loops, e.g., $r_{ij} = r_A$, $g_{ij} = g_A$, $\psi_j(l) = \psi_j(AB)$, etc.

Link Definitions:

Vehicle platoons released at the start of green at S_i travel to S_j on link (i,j). Fig. 2 relates travel time and offset. Let

τ_{ij} = travel time of platoon's leading edge from S_i to S_j,
γ_{ij} = arrival time of the platoon's leading edge at S_j, measured relative to start of green there. Without loss of generality, take: $-r_{ij} \leqq \gamma_{ij} \leqq g_{ij}$.
ϕ_{ij} = internode offset in extended form:

$$\phi_{ij} = \tau_{ij} - \gamma_{ij}. \tag{1.1}$$

This is the time from the start of a green at S_i to the start of a particular green at S_j, namely, the one first encountered by the head of a platoon leaving at the start of green at S_i.

$\theta_{ij} = [\phi_{ij}] \bmod C$ = internode offset in reduced form; this is the time from the start of green at S_i to the start of the green at S_j occurring next, $0 \leqq \theta_{ij} < C$.

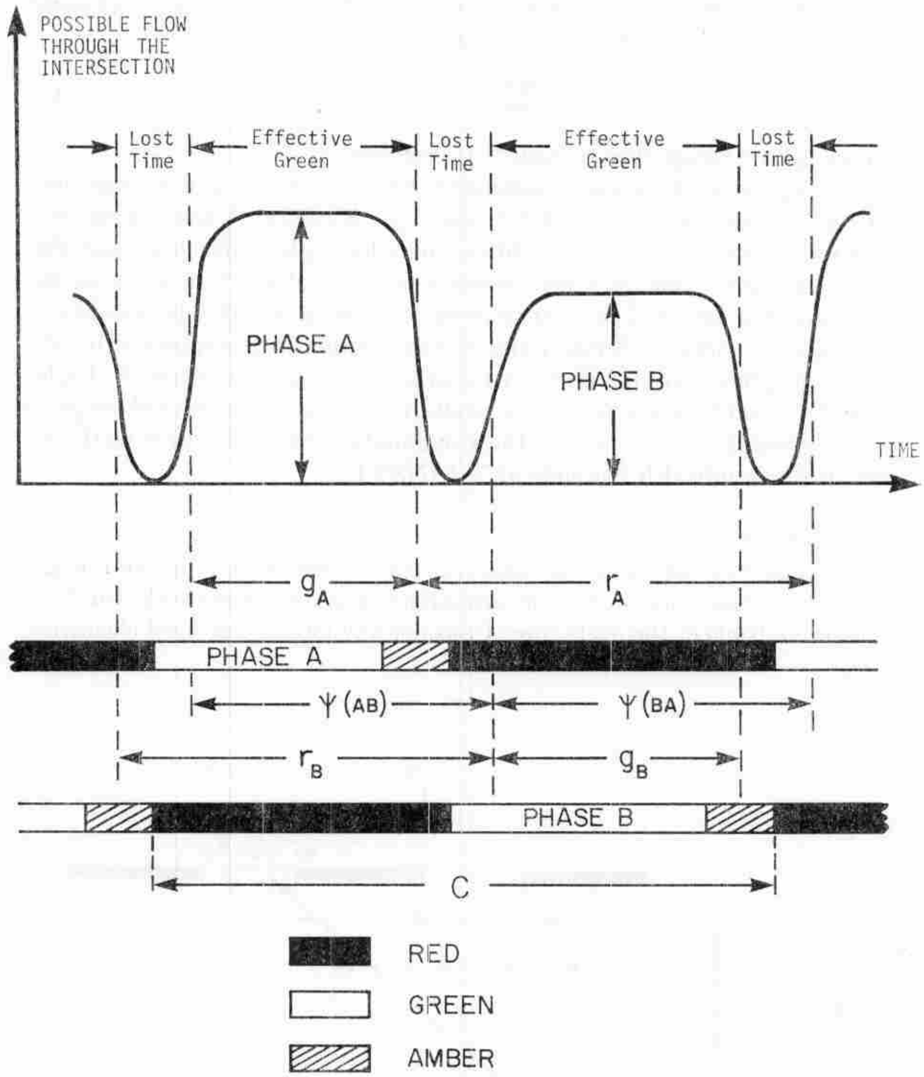

Fig. 1. Basic model for traffic signal operation.

Objective Function

Let, $z_{ij}(\phi_{ij})$ = link performance function; the average loss per vehicle on link (i,j) for traveling through S_j. By loss is meant a disutility such as delay, stops, acceleration noise, etc., or some chosen combination of these quantities. Although z_{ij} depends on splits and cycle time as well as offset, these quantities are constants for the coordination problem addressed first. The overall goal is to minimize the total *loss rate*

$$\min \sum_{(i,j)\epsilon L} f_{ij} z_{ij}(\phi_{ij}), \tag{1.2}$$

where f_{ij} = average flow on link (i,j) (veh/sec).

This objective function is nonlinear (because of z_{ij}) and is *separable*, i.e., z_{ij} depends on the offset for link (i,j). NEWELL[17] has argued in a theoretical study that this condition holds for high traffic flows and also when platoons spread so much between signals that the time between the arrival of the first and last cars exceeds the green period. The validity of separability under moderately heavy flow conditions is supported by the empirical studies of HILLIER AND ROTHERY[18] and GARTNER.[19] Under light flow conditions separability would not always be expected. Separability is implicitly assumed in the Combination Method and SIGOP but not in the bandwidth methods or TRANSYT.

Loop Constraints

An important physical constraint on the system is that the sum of the offsets, internode plus intranode, around any loop of the network equals an integer multiple of the cycle time. Consider any loop l, traversed clockwise. Let,

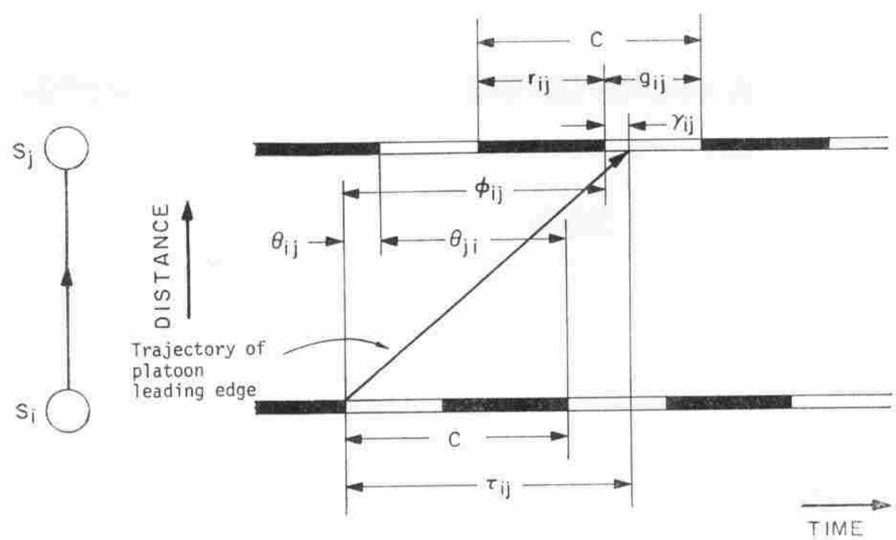

Fig. 2. Parameters of signal-controlled link.

$F(l)$ = set of forward links in l,
$R(l)$ = set of reverse links in l,
$N(l)$ = set of nodes of l,
$\psi_j(l) = \psi_j(AB)$, where A = phase assigned to the incoming link at S_j, and B = phase assigned to the outgoing link, $j \epsilon N(l)$.

$$\sum_{(i,j)\epsilon F(l)} \phi_{ij} - \sum_{(i,j)\epsilon R(l)} \phi_{ij} + \sum_{j\epsilon N(l)} \psi_j(l) = n_l C. \quad (n_l \text{ integer}) \quad \forall l \epsilon \mathcal{L} \quad (1.3)$$

Optimization

The signal coordination problem can now be stated as a mixed-integer nonlinear program:

MINLP: *Given* $G, C, \{f_{ij}, r_{ij}, g_{ij}\}$,
Find values of $\{\phi_{ij}\}, \{n_l\}$ *to*:
$$\min \sum_{(i,j)\epsilon L} f_{ij} z_{ij}(\phi_{ij}),$$
subject to Constraint (1.3); ϕ_{ij}, n_l unrestricted in sign.

2. LINK PERFORMANCE FUNCTION

A LINK PERFORMANCE function (LPF) expressing average delay per vehicle is developed. [Since the section deals exclusively with traffic on link (i,j), we simplify notation by dropping subscripts.]

2.1 Basic Relations

Let the beginning of green be at time zero. Thus, the interval $[-r, g]$ denotes one cycle period consisting of an effective red period $[-r, 0]$ and an effective green period $[0, g]$. Define:

$q(t)$ = arrival rate (vehicles/second),
$A(t)$ = cumulative number of arrivals between start of red and time t (vehicles),

$$A(t) = \int_{-r}^{t} q(\tau)\, d\tau, \qquad (2.1)$$

$A_C = A(g)$ = total number of arrivals in one cycle (vehicles),
$Q(t)$ = queue length at time t (vehicles),
s = saturation flow rate (veh/sec).

Saturation flow is maintained during green as long as a queue exists; thereafter, departure rate equals arrival rate.

For the present, we consider deterministic flows only. Stochastic effects are taken up in Part II. We assume:

(A.1) Arrivals are periodic: $q(t) = q(t \pm C)$.
(A.2) The signal is undersaturated: $A_C \leqq gs$.

(A.3) Arrival rate during green does not exceed saturation flow.

(A.4) At some point in the past, a zero queue has occurred.

(A.2) says that the total of vehicles arriving in a cycle does not exceed the number of possible departures. (A.3) implies that once a queue has vanished during green it cannot rebuild before the next red. Together the four assumptions imply that a periodic situation exists in which all vehicles arriving during $[-r, 0]$ are accommodated during $[0, g]$ along with any arrivals during $[0, g]$. It follows that the queue is always empty at the end of green and delay calculations can be confined to the single interval $[-r, g]$.

The queue length is the difference between cumulative arrivals and cumulative departures:

$$Q(t) = \begin{cases} A(t), & t\epsilon[-r, 0], \\ A(t) - ts, & t\epsilon[0, t_0], \\ 0, & t\epsilon[t_0, g], \end{cases} \tag{2.2}$$

where t_0 is the point during green when the queue disappears and is obtained by the solution of

$$A(t_0) - t_0 s = 0. \tag{2.3}$$

The delay incurred by $Q(t)$ vehicles during dt is $Q(t)dt$ so that the total delay, Z, incurred by traffic during one cycle is

$$Z = \int_{-r}^{g} Q(t)\, dt = \int_{-r}^{t_0} Q(t)\, dt \tag{2.4}$$

and the average delay per vehicle is $z = Z/A_C$. Subsequent discussions consider delay as the only measure of loss, but the models can be developed to calculate more general link performance functions such as weighted combinations of delay, stops, acceleration noise, etc.[20,21]

2.2 Traffic Flow Model

We construct a simple model to calculate delay as a function of arrival time. Traffic flow will be represented by a rectangular pulse of height q (veh/sec), length p (sec) and net size qp (veh); *see* Fig. 3. The platoon must correspond to the average flow, f (veh/sec) over the whole cycle. Flow is generally made up of a primary component f' and a secondary component f''. If the upstream green time feeding the primary flow is g', then the platoon length of the primary flow is assumed to be $p' = g'$ (sec) with height $q = f'C/g'$ (veh/sec). In other words, the flow is taken uniform over the upstream green. Alternative assumptions, such as a 'tadpoling' flow pattern,[22] could equally well be made. Secondary flow is added on with

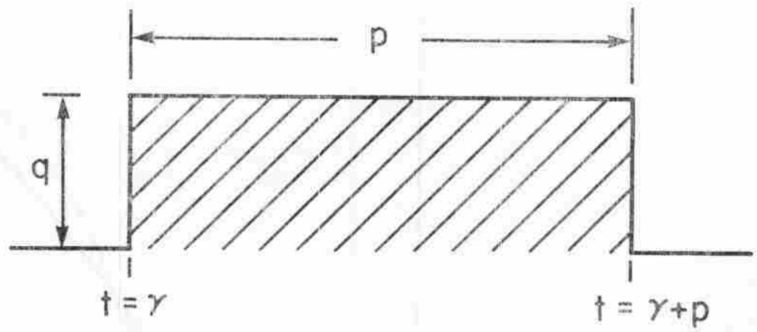

Fig. 3. Traffic flow pattern.

the same height in order to preserve the platoon shape. This implies an additional length $p'' = f''C/q$ for a total length of

$$p = p' + p'' = g'[1 + (f''/f')] \text{ (sec)}. \tag{2.5}$$

In certain cases, usually when $f'' > f'$, the roles of the primary and secondary components may have to be interchanged.

We need p and q at the downstream end of the link. Taking into account dispersion effects, the platoon length is:

$$p(d) = \begin{cases} k(d)\ p(0) & \text{if } k(d)\ p(0) \leqq C, \\ C & \text{otherwise,} \end{cases} \tag{2.6}$$

where d is the link length and $k(d)$ is the platoon dispersion factor, which may be calibrated for each link. See, for example, references 23 and 24. As indicated in (2.6), the platoon length cannot exceed the cycle time in a periodic system, since then flow becomes continuous. The value of $q(d)$ is determined by conservation of vehicles, i.e., $q(d) = q(0)p(0)/p(d)$. An example of a comparison between delay computed for an actual platoon and for a rectangular approximation is shown in Fig. 4. A more extensive evaluation is given in reference 25.

2.3 LPF for Rectangular Platoons

In this section, the link performance function, $z(\gamma)$, is calculated for the traffic flow model described above. Since the rectangular platoon leads to a nonlinear $z(\gamma)$, which we shall eventually want in piecewise linear form, several results are proven on the location of maxima and minima of $z(\gamma)$. These will subsequently provide anchor points for the linearization.

THEOREM 2.1. *Delay is minimal when the trailing edge of the platoon arrives at the light just before it turns red and is able to clear the intersection entirely.*

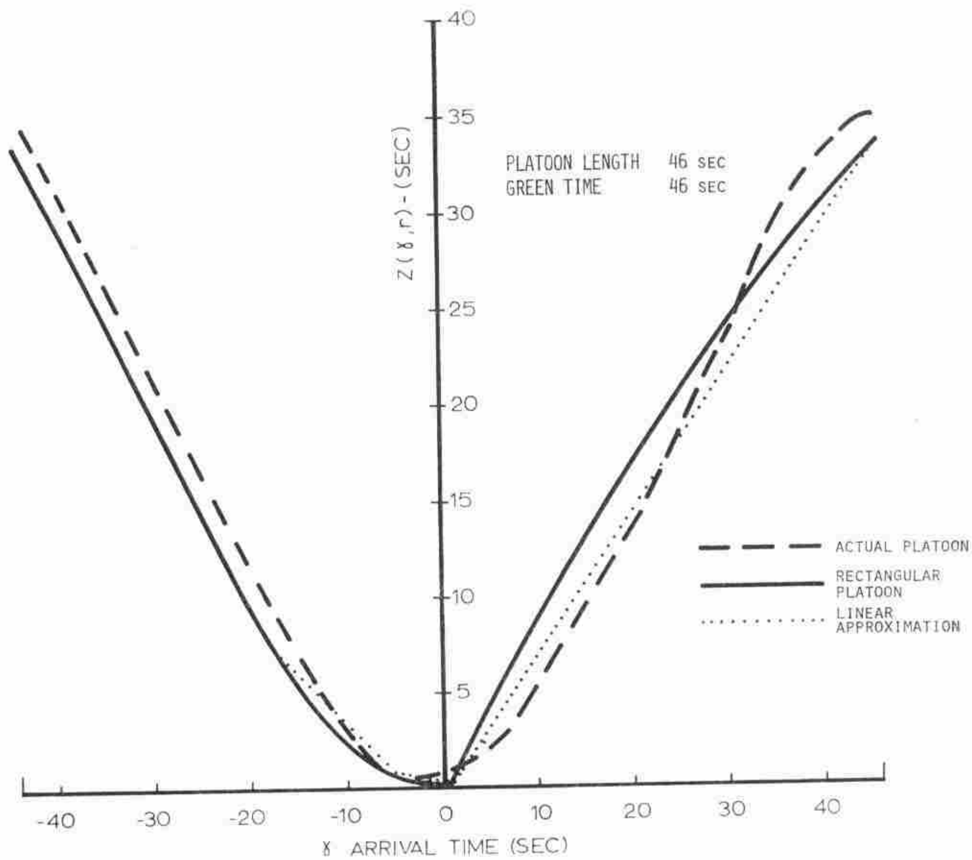

Fig. 4. Comparison of link performance function for actual and rectangular platoons.

This occurs for $\gamma = \underline{\gamma} \equiv g - p$ *and (with* $y \equiv q/s$*) yields the minimal delay*

$$z_{\min} = \begin{cases} 0, & p \leqq g, \\ \underline{\gamma}^2/2p(1-y), & p \geqq g. \end{cases} \tag{2.7}$$

Proof. Referring to Fig. 5, we distinguish two cases:

Case A: $p \leqq g$. The platoon is able to fit into the green opening. A degeneracy occurs for $\gamma \epsilon [0, \underline{\gamma}]$ since no cars are stopped. This is shown in Fig. 5A.1. Minimal delay is zero. For $\gamma < 0$, *headstopping* occurs: $q(-\gamma)$ vehicles arrive during red and incur a delay shown in Fig. 5A.2. For $\gamma > \underline{\gamma} = g - p$, a length p_x of the platoon is *tailchopped* by the red light and qp_x cars are delayed as shown in Fig. 5A.3. Therefore, delay is minimal only for $\gamma \epsilon [0, \underline{\gamma}]$.

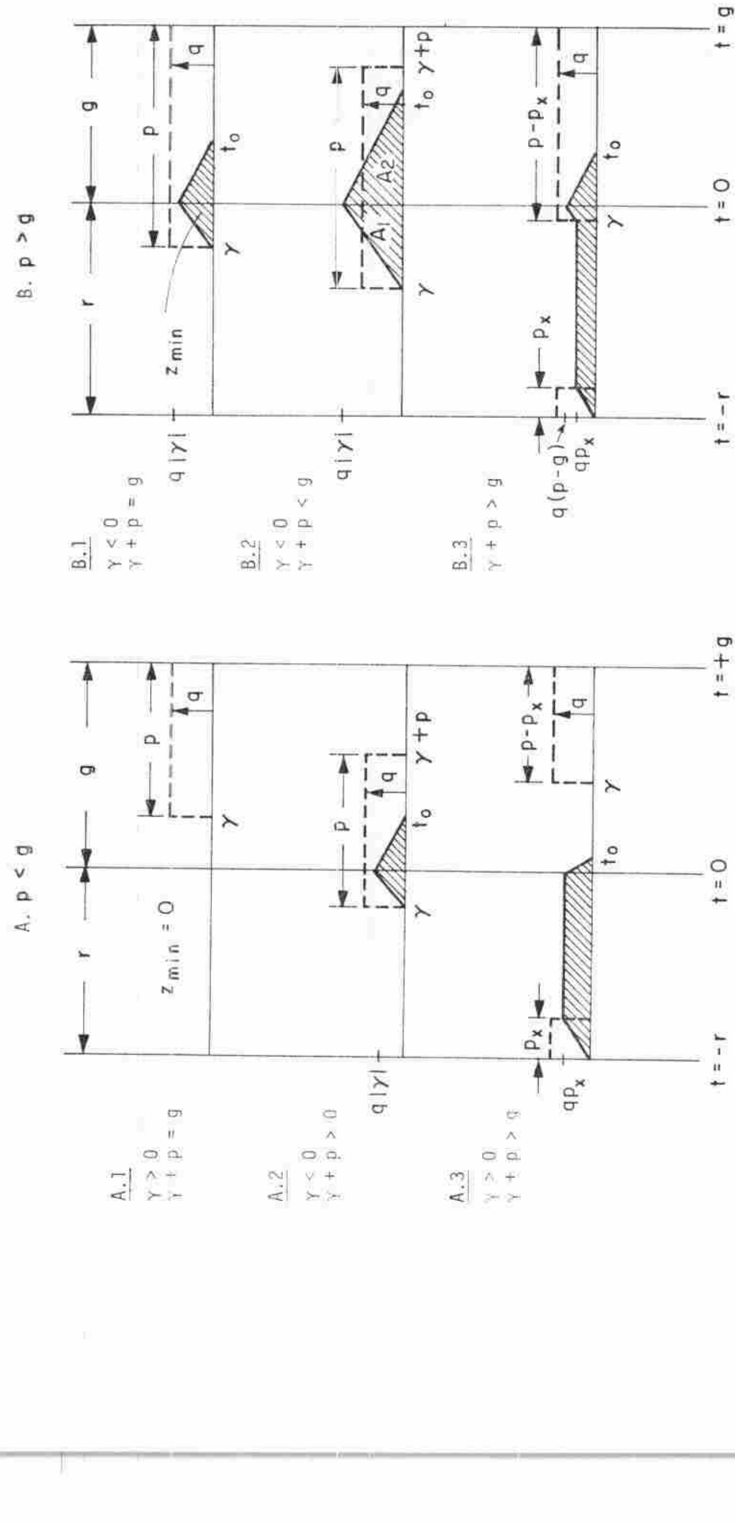

Fig. 5. Arrivals and queuing at minimal delay (two cases).

Case B: $p \geqq g$. When $\gamma = \underline{\gamma} = g - p$, $q(-\gamma)$ vehicles are stopped and delay is as shown by the shaded area of Fig. 5B.1. For $\gamma < \underline{\gamma}$, headstopping increases and with it delay as shown in Fig. 5B.2. For $\underline{\gamma} < \gamma < 0$, the total vehicles arriving in red is the same as in Fig. 5B.1, but both headstopping and tailchopping occur: qp_x vehicles are stopped right after the light turns red and total delay is larger. This is shown in Fig. 5B.3. Further increases in γ cause more cars to wait throughout red with even greater delay. Minimal delay is therefore at $\gamma = \underline{\gamma}$.

For case B, the minimal delay can be calculated as follows. For $\gamma \leqq \underline{\gamma}$, but close to $\underline{\gamma}$, total delay is the sum of two areas:

$$Z(\gamma) = A_1 + A_2 = (\tfrac{1}{2})q\gamma^2 + \int_0^{t_0} [q(t - \gamma) - st]\,dt \qquad \text{(veh/sec)}.$$

By (2.4), $t_0 = (-\gamma)y/(1 - y)$. Performing the integration and simplifying, the average delay is

$$z(\gamma) = \gamma^2/2p(1 - y) \qquad \text{(sec)}. \tag{2.8}$$

Substitution of $\gamma = \underline{\gamma}$ completes the proof.

THEOREM 2.2. *Delay is maximal when the leading edge of the platoon arrives at the signal stop line right after the light rurns red. This occurs for $\gamma = \underline{\gamma} \equiv -r$ and yields the maximal delay:*

$$z_{\max} = \begin{cases} r + p(y - 1)/2, & p \leqq t_0 + r, \\ r^2/2p(1 - y), & p > t_0 + r. \end{cases} \qquad \text{(sec.)} \tag{2.9}$$

Furthermore, the delay for $\gamma = g$ is the same as for $\gamma = -r$.

Proof. Figure 6 displays three cases that are exhaustive.

Case A: $p \leqq r$. Figure 6A.1 shows queue evolvement and delay for $\gamma = \bar{\gamma} = -r$. If we move to $\gamma > \bar{\gamma}$, the platoon arrives later in red and stopped vehicles wait a shorter time, as shown in Fig. 6A.2. Movement of γ in the other direction is depicted in Fig. 6A.3. We have $\gamma < g$ and a length $(g - \gamma)$ of the platoon passes through the intersection unimpeded, thereby reducing delay relative to $\gamma = \bar{\gamma}$. Thus, $\bar{\gamma}$ is locally maximizing and, by inspection of cases, globally maximizing.

To calculate maximal delay, consider Figs. 6A.1 and 6A.2. For $\gamma \geqq \bar{\gamma}$ but close to it, total delay is

$$Z(\gamma) = A_1 + A_2 + A_3 = qp^2/2 + pq(-\gamma - p) + \int_0^{t_0} (pq - st)\,dt.$$

Queue clearance occurs when the integrand is zero, so that $t_0 = yp$ and

$$z(\gamma) = p(y - 1)/2 - \gamma \qquad \text{(sec)}. \tag{2.10}$$

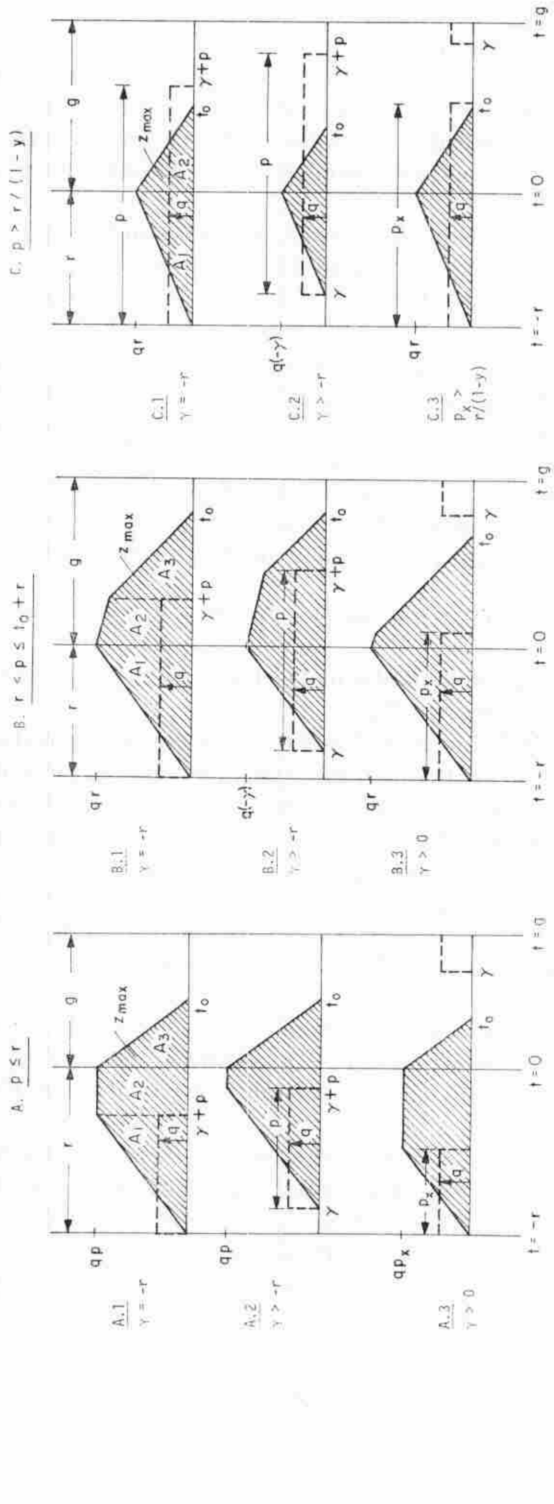

Fig. 6. Arrivals and queuing at maximal delay (three cases).

The maximum occurs at the extreme point $\gamma = \bar{\gamma} = -r$ as expected. Substitution completes case A.

Case B: $r < p \leqq t_0 + r$. In this situation, the platoon length is such that the platoon tail arrives during green. We exclude, however, the possibility that arrivals extend beyond the queue clearance time. Figure 6B depicts the case. The analysis is similar to case A; (2.10) holds and results are identical.

Case C: $t_0 + r \leqq p$. Now the arrivals extend beyond the queue clearance time and only two areas are required for delay calculations as shown in Fig. 6C. Equation (2.8) holds in this case and maximal z occurs at $\gamma = \bar{\gamma} = -r$.

We notice, however, that degeneracy occurs for large platoons. Provided that some part of the platoon is caught by the start of red and continues until after queue clearance, it does not matter how much of the platoon passes through unscathed before or after. These pieces add no delay. The situation is shown in Fig. 6C.3. The conditions for this degeneracy are $\gamma \leqq g$ and $p_x \geqq r + t_0$, where $p_x = p - (g - \gamma)$. Noting that $t_0 = ry/(1 - y)$ for p_x, we find a constant maximal z for

$$C - p + ry/(1 - y) \leqq \gamma \leqq g. \tag{2.11}$$

Figures 6A.3, 6B.3, and 6C.3 also make clear that $z(-r) = z(g)$. This completes the proof of Theorem 2.2.

The cases illustrated in Figs. 5 and 6 do not cover all possible combinations of arrival time, platoon length, split, flow, and saturation flow, but they cover all cases in the vicinity of the maximum and minimum of $z(\gamma)$. In the course of proving Theorems 2.1 and 2.2, we have also established the following corollaries. (*See* Fig. 7).

COROLLARY 2.1. *The locus of minimal delay in the* (γ, r) *plane consists of the line* $\gamma = g - p = C - r - p$ *plus the triangular area defined by* $\gamma \geqq 0$, $\gamma \leqq C - r - p$, *and* $r \geqq 0$. *Delay is minimal in the latter area and equals zero.*

COROLLARY 2.2. *The locus of maximal delay in the* (γ, r) *plane consists of the lines* $\gamma = -r$ *and* $\gamma = g = C - r$, *plus the area defined by* $\gamma \leqq C - r$, $\gamma \geqq C - p + ry/(1 - y)$, *and* $r \geqq 0$. *Delay is maximal in the latter area and, by* (2.8), *equals* $r^2/2p(1 - y)$.

THEOREM 2.3. *For* $\gamma \epsilon [-r, 0]$, *the delay function* $z(\gamma)$ *is strictly decreasing.*

Proof. Referring to Figs. 6A.2, 6B.2, and 6C.2, we see that shifting γ from γ_1 to γ_2, where $\gamma_1 < \gamma_2 \leqq 0$, results in a queue length $Q_2(t) \leqq Q_1(t)$ for all t and $Q_2(t) < Q_1(t)$ for some t. Therefore, $z(\gamma_2) < z(\gamma_1)$.

THEOREM 2.4. *For* $\gamma \epsilon [0, g]$, *the delay function* $z(\gamma)$ *is nondecreasing.*

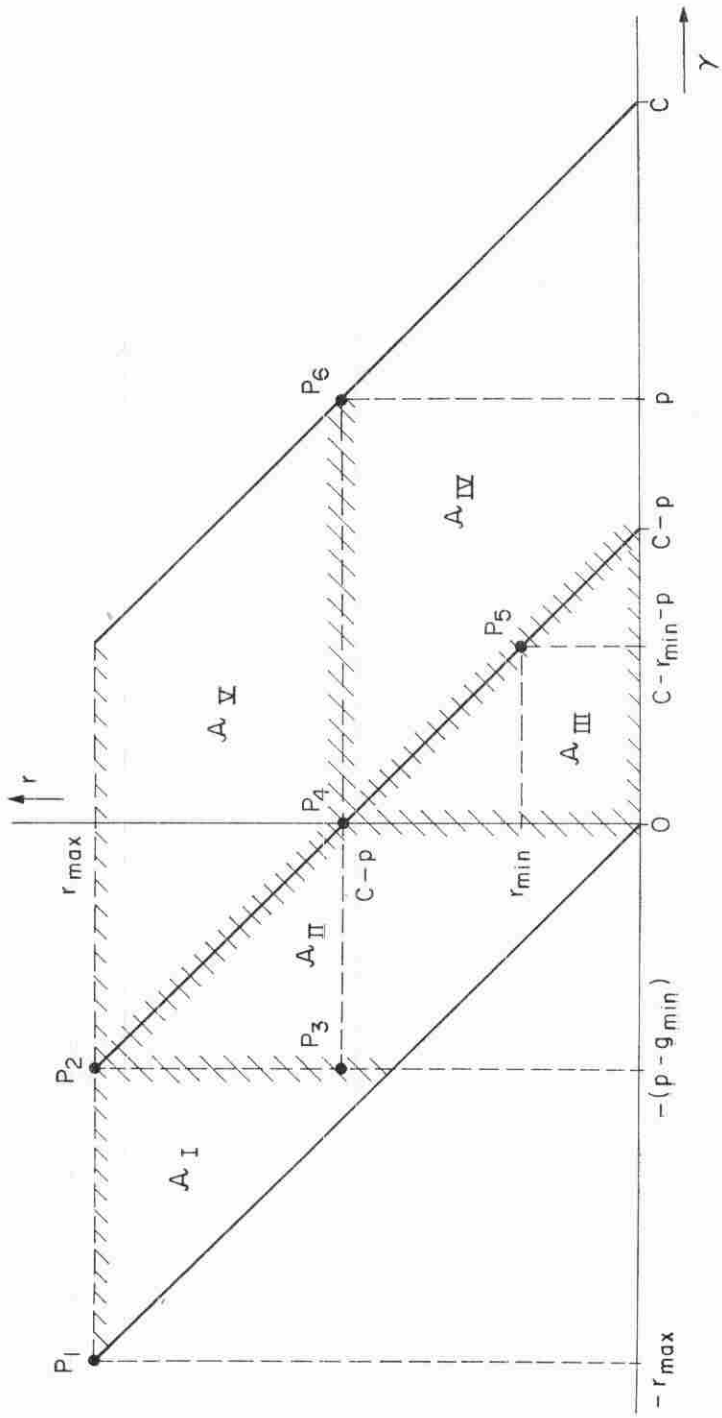

Fig. 7. Loci of z_{min} and z_{max} in (γ, r) plane.

Proof. We distinguish four cases.

p	$\leqq t_0 + r$	$> t_0 + r$
$> g$	I	III
$\leqq g$	II	IV

I. Referring to Fig. 5A.3 and 5B.3, any shift of γ from γ_1 to γ_2, where $\gamma_2 > \gamma_1 \geqq 0$, results in a queue length $Q_2(t) \geqq Q_1(t)$ for all t and $Q_2(t) > Q_1(t)$ for some t. Therefore, $z(\gamma_2) > z(\gamma_1)$ and z is strictly increasing.

II. $z(\gamma)$ has a flat, zero portion for $\gamma \epsilon [0, g - p]$ and is strictly increasing for $\gamma \epsilon [g - p, g]$.

III. $z(\gamma)$ is strictly increasing for $\gamma \epsilon [0, C - p + ry/(1 - y)]$ and has a flat maximum (varying, however, with split) for $\gamma \epsilon [C - p + ry/(1 - y), g]$.

IV. $z(\gamma)$ is flat and zero for $\gamma \epsilon [0, g - p]$, increases strictly for $\gamma \epsilon$ $[g - p, C - p + ry/(1 - y)]$ and enters the flat maximum thereafter.

THEOREM 2.5. *Delay is invariant with split for* $\gamma \epsilon [\bar{\gamma}, \underline{\gamma}]$ if:

(a) $\gamma + p < t_0$, i.e., $p < \gamma/1 - y$, and $t_0 \leqq g$;

(b) $t_0 < \gamma + p$, i.e., $p > r/1 - y$, and $\gamma + p \leqq g$.

Proof. (a) We are talking about Figs. 5A.2 and 5B.2, which are algebraically the same, plus Figs. 6A.2 and 6B.2. Equation (2.8) holds in the former case and (2.10) in the latter.

(b) Fig. 6C.2 and equation (2.8) apply in this case.

None of the expressions in equations (2.8), or (2.10) depends on r or g.

The theorems and expressions developed so far permit us to devise linear approximations. The natural breakpoints of $z(\gamma)$ identifying maxima and minima can be determined and as many intermediate points as desired calculated. Our convention is to use secant approximating lines that are chosen to make the resulting LPF convex. An example of the piecewise linearization of $z(\gamma)$ is illustrated by the dotted lines in Fig. 4. It is noteworthy that in certain regions, the linearized approximation is closer to the original function derived from the actual platoon than to the function obtained from the rectangular platoon. We note that piecewise linear approximations have been proposed in the past. Two-piece V-shaped functions have been used to determine offsets on arterials[26] and tree networks.[27] Several authors have also suggested the application of linear programming techniques to minimize a linearized objective function of the offsets in a network.[28-30] However, none of these authors have developed the proposal into an operational model and no computational results have been reported.

3. MIXED-INTEGER LINEAR PROGRAM

WE HAVE A piecewise linear delay function $z(\gamma)$, $\gamma \epsilon [-r, g]$, for each link. This can be converted to $z(\phi)$, by equation (1.1). Although $z(\phi)$ must

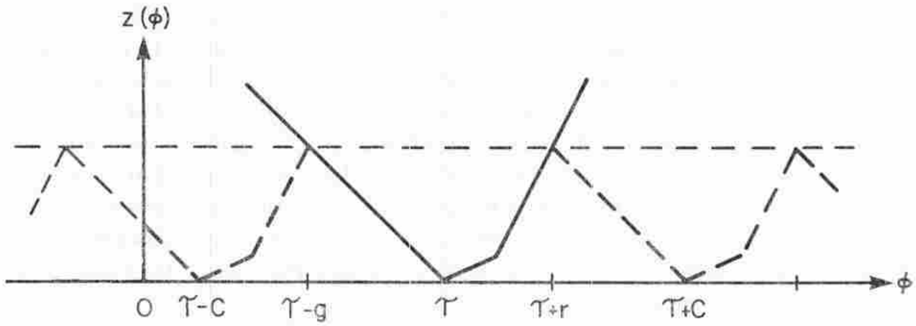

Fig. 8. Piecewise linear convex periodic objective function.

have a value for any ϕ, we can constrain ϕ to $[\tau - g, \tau + r]$ because z is periodic with C. This is sketched in Fig. 8. Suppose, for a given link, the segments forming $z(\phi)$ come from lines in the (ϕ,z) plane denoted $z = z^k(\phi)$; $k = 1, \ldots, K$. In the linear program we can introduce the constraints

$$z \geqq z^k(\phi), \qquad k = 1, \ldots, K \quad (3.1)$$

$$\tau - g \leqq \phi \leqq \tau + r, \qquad (3.2)$$

and bring z into the objective function to represent the link. The minimization will force an equality in one or more of (3.1) that, because of the convexity of $z(\phi)$ over $[\tau - g, \tau + r]$, will always be a point of $z(\phi)$. We now observe that (3.2) is not strictly required.

THEOREM 3.1. *Provided that the line segment of $z(\phi)$ at $\tau - g$ has a negative slope and at $\tau + r$ a positive slope, any optimal offset ϕ will be in the interval $[\tau - g, \tau + r]$.*

Proof. Since $z(\phi)$ is convex and has the indicated slopes, it is everywhere larger outside of $[\tau - g, \tau + r]$ than inside. However, there is always an integer m such that a translation of mC will bring ϕ inside. The only constraints on ϕ besides (3.1) are the loop constraints but these will not be violated if an offset is changed by a multiple of a cycle time. Hence, optimization will choose a value of ϕ in $[\tau - g, \tau + r]$, as was to be shown.

The loop constraints, equation (1.3), state that the sum of offsets around any loop of the network, taking into account the direction each link is traversed and considering the internode as well as intranode offsets, is an integral number of cycle times. Only a *fundamental* or independent set, $\mathfrak{L}_f$, of loops is required since all other loop equations are linear combinations of an independent set. In a network with $|N|$ nodes and $|L|$ links, exactly $|L| - |N| + 1$ loops are in an independent set. [31] A corollary to this is that $|N| - 1$ offset variables are independent and correspond to links that form a tree subnetwork of G.

A goal in choosing a set $\mathfrak{L}_f$ relates to the computational efficiency of mixed-integer programming. It is desirable to limit the number of integer lattice points of the problem. This may be achieved by choosing loops for $\mathfrak{L}_f$ that contain as few links as possible, which in a planar network are the meshes of the planar sketch.[32] In addition, the integer values that the program has to consider can be reduced by further constraints. For example, (3.2) can be included for each link. Alternatively, upper and lower bounds n_l^u and n_l^1 can be set on n_l by substituting (3.2) properly subscripted by link into the sums of (1.3) and then rounding each bound to its nearest appropriate integer.

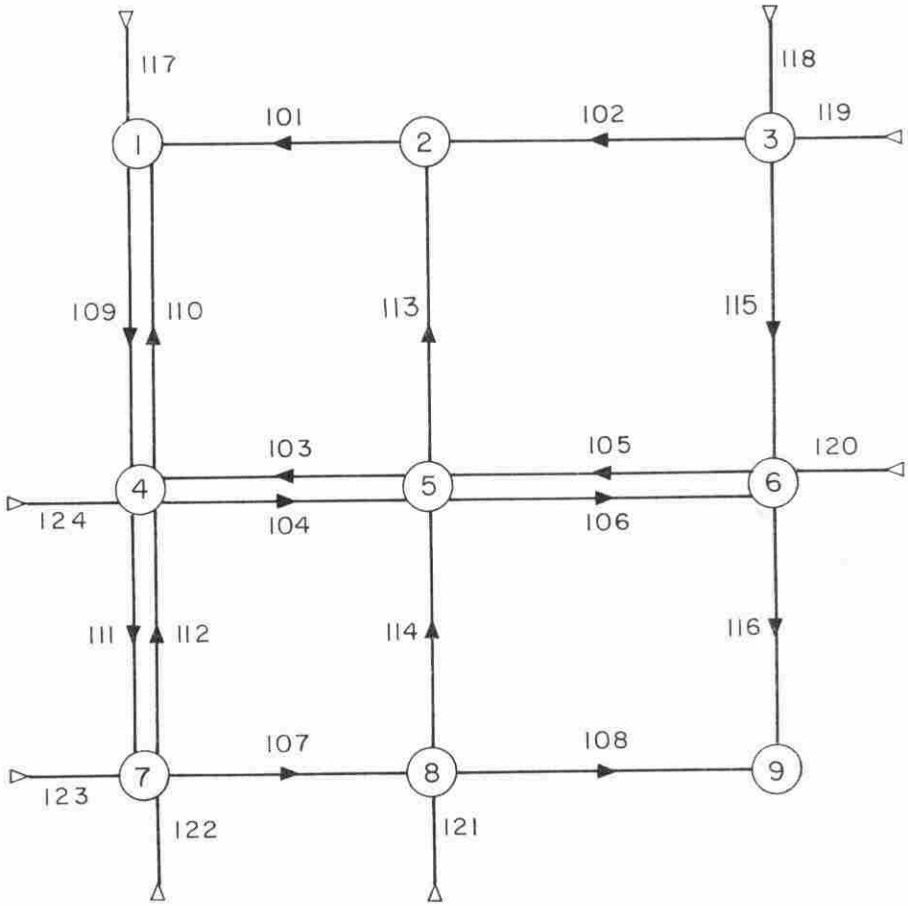

Fig. 9. Test network diagram.

We can now state the network coordination problem as a mixed-integer linear program.

MILP: Find $\{\phi_{ij}\}$ to

$$\min \sum_{(i,j)} f_{ij} z_{ij},$$

subject to

TABLE I

Data for Test Network

Link No.	Length (ft)	Velocity (ft/sec)	Travel Time (sec)	Flow (veh/sec)	Sat. Flow (veh/sec)	Platoon Length (cycle)
101	600	32.27	18.60	0.152	0.835	0.335
102	800	32.27	24.79	0.152	0.835	0.344
103	600	32.27	18.60	0.152	0.835	0.418
104	600	41.07	14.61	0.22	0.835	0.407
105	800	32.27	24.79	0.152	0.835	0.463
106	800	41.07	19.48	0.22	0.835	0.444
107	600	41.07	14.61	0.25	0.600	0.515
108	800	41.07	19.48	0.25	0.600	0.560
109	1000	35.20	28.41	0.175	0.500	0.701
110	1000	41.07	24.35	0.11	0.500	0.608
111	550	35.20	15.63	0.175	0.500	0.532
112	550	41.07	13.39	0.11	0.500	0.425
113	1000	41.07	24.35	0.12	0.360	0.595
114	550	41.07	13.39	0.12	0.360	0.414
115	1000	35.20	28.41	0.175	0.500	0.701
116	550	35.20	15.63	0.175	0.500	0.503

Input Links						
117				0.175	0.500	
118				0.175	0.500	
119				0.152	0.835	
120				0.152	0.835	
121				0.12	0.360	
122				0.11	0.500	
123				0.25	0.600	
124				0.22	0.835	

Notes: (a) Lost time for each link, $l = 4.5$ sec.

(b) Platoon length on input links, $p = 1$ cycle (i.e., flow is assumed to be continuous, though random fluctuations will be introduced in subsequent formulations).

(c) Data on input links will be used in numerical examples given in Part II of this paper.

TABLE II

Test Network Results and Statistics

			Time (min)	Functional (veh×sec/sec)
Rows	161	Continuous Optimum	0.03	43.3461
Columns	97	First Integer Solution	0.06	62.6695
Variables	258	Optimal Integer Solution	0.07	60.8741
Integer Variables	8	Optimality Proved	0.21	
Nonzero Elements	709	Time of Search	0.21	
Density	1.70			

$$z_{ij} \geqq z_{ij}^k(\phi_{ij}), \qquad (k = 1, \ldots, K_{ij}) \qquad (\forall (i,j)\epsilon L)$$

$$\sum_{(i,j)\epsilon F(l)} \phi_{ij} - \sum_{(i,j)\epsilon R(l)} \phi_{ij} + \sum_{k\epsilon N(l)} \psi_k(l) = n_l C, \qquad (\forall l \epsilon \mathfrak{L}_f)$$

$$n_l^1 \leqq n_l \leqq n_l^u. \qquad (n_l \text{ integer})$$

4. TEST NETWORK SOLUTION

THE FOLLOWING DATA describe the test network for which a sample problem is actually solved on the IBM 370/165 using IBM's Mathematical Programming Systems (MPSX) package. Figure 9 is a sketch of the test network. There are 9 nodes and 24 links of which 16 are internal to the network and 8 are input. The input links although not considered in this problem will be mentioned in later solutions. Table I presents the data for this and later examples using the test network. In this example only offsets are decision variables with green splits and cycle time fixed according to Webster. The critical intersection is No. 7 in Fig. 9, yielding an 80-sec cycle time. Tables II, III, and IV present the main output data. The functional value in Table II includes both deterministic and stochastic delays.

TABLE III

Loops of the Test Network and Their Corresponding Integer Variables

Loop No.	Integer Variable	Link Numbers				First Integer Solution	Optimal Integer Solution
1	N1	109	110			1	1
2	N2	103	104			1	1
3	N3	105	106			1	1
4	N4	111	112			1	1
5	N5	101	109	104	113	−1	−1
6	N6	103	111	107	114	−1	−1
7	N7	115	105	113	102	0	−1
8	N8	114	106	116	108	0	0

TABLE IV

Result of Computations

(C = 80 sec; cycle time and green times fixed according to Webster)

Link No.	Offset sec (cycle)	Green Time sec (cycle)	Avg. Delay sec
101	21.6 (0.27)	24.0 (0.30)	1.36
102	27.2 (0.34)	24.8 (0.31)	1.36
103	64.0 (0.80)	30.4 (0.38)	36.00
104	16.0 (0.20)	31.2 (0.39)	0.72
105	47.2 (0.59)	31.2 (0.39)	11.28
106	32.8 (0.41)	32.8 (0.31)	6.64
107	16.8 (0.21)	39.2 (0.49)	1.04
108	25.6 (0.30)	38.4 (0.48)	3.20
109	26.4 (0.33)	40.8 (0.51)	10.88
110	53.6 (0.67)	46.4 (0.58)	14.80
111	32.8 (0.41)	32.0 (0.40)	9.60
112	47.2 (0.59)	40.8 (0.51)	25.92
113	16.8 (0.21)	45.6 (0.57)	5.36
114	47.2 (0.59)	40.0 (0.50)	30.64
115	46.4 (0.58)	38.4 (0.48)	8.88
116	24.0 (0.30)	32.0 (0.40)	4.16
Input Links			
117		46.4 (0.58)	12.16
118		46.4 (0.58)	12.16
119		24.0 (0.30)	24.56
120		32.8 (0.41)	18.64
121		31.2 (0.39)	22.16
122		32.0 (0.40)	19.76
123		38.4 (0.48)	18.64
124		30.4 (0.38)	21.52

5. DISCUSSION

THE TRAFFIC SIGNAL network coordination problem has been formulated for mixed-integer linear programming. Thus, a global optimal solution to the problem can be achieved via existing optimization packages. Whereas much previous research has devoted its main effort to optimization methods, the emphasis here is on accurately modeling the traffic process.

It is shown that the piecewise linear LPF constitutes a good approximation to the behavior of actual platoons under moderate to heavy traffic. The degree of accuracy of the approximation can be determined by the modeler by choosing the number of linear constraints, but must be weighed against increased computation time.

The model characteristics that pose the most serious demand on computation time are the loop constraints. They contain integer variables, thus enforcing a search procedure of a combinatorial nature. The problem attacked here has relatively few of these constraints and practical size networks can be solved in a reasonable time.[25]

An important aspect of the approach is that it forms the basis for a more comprehensive formulation that will include all the interdependent control variables, i.e., offsets, splits and cycle time, as simultaneous decision variables. This extension to the model is developed in Part II of this paper.[12]

REFERENCES

1. J. E. Baerwald (ed.), *Traffic Engineering Handbook*, 3rd Edition, Institute of Traffic Engineers, Washington, D. C., 1965.
2. F. V. Webster and B. M. Cobbe, *Traffic Signals*, Road Research Technical Paper No. 56, H. M. Stationery Office, London, 1966.
3. J. T. Morgan and J. D. C. Little, "Synchronizing Traffic Signals for Maximal Bandwidth," *Opns. Res.* **12,** 896–912 (1964).
4. J. D. C. Little, "The Synchronization of Traffic Signals by Mixed-Integer Linear Programming," *Opns. Res.* **14,** 568–594 (1966).
5. B. S. Marrus and M. F. Main, "New Method Improves Traffic Signal Timing," *Traffic Eng.*, 23–26, June 1964.
6. "SIGOP: Traffic Signal Optimization Program—A Computer Program to Calculate Optimum Coordination in a Grid Network of Synchronized Traffic Signals," Traffic Research Corporation, New York, September 1966, Clearinghouse PB 173 738.
7. J. A. Hillier, "Appendix to Glasgow's Experiment in Area Traffic Control," *Traffic Eng. and Control* **7,** 569–571 (1966).
8. R. E. Allsop, "Selection of Offsets to Minimize Delay to Traffic in a Network Controlled by Fixed-Time Signals," *Trans. Sci.* **2,** 1–13 (1968).
9. N. Gartner, "Optimal Synchronization of Traffic Signal Networks by Dynamic Programming," *Traffic Flow and Transportation* (G. F. Newell, ed.), pp. 281–295, American Elsevier, New York, 1972.
10. D. I. Robertson, "TRANSYT: A Traffic Network Study Tool," Road Research Laboratory Report LR 253, Crowthorne, Berkshire, 1969.
11. "Mathematical Programming System Extended (MPSX)—Mixed Integer Programming (MIP) Program Description," Program No. 5734-XM4, IBM Corp., Technical Publications Dept., White Plains, N. Y., 1971.
12. Part II of this paper, "The Network Synchronization Problem," *Trans. Sci.* **9,** 344–363 (1975).
13. A. J. H. Clayton, "Road Traffic Calculations," *J. Instn. of Civil Engrs.* **16,** p. 247 (1940–41).
14. J. G. Wardrop, "Some Theoretical Aspects of Road Traffic Research," *Proc. ICE* **1,** 325–378 (1952).
15. F. V. Webster, "Traffic Signal Settings," Road Research Technical Paper No. 39, H. M. Stationery Office, London, 1958.

16. A. J. MILLER, "The Capacity of Signalized Intersections in Australia," Bulletin No. 3, Australian Road Research Board, March 1968.
17. G. F. NEWELL, "Synchronization of Traffic Lights for High Flow," *Quart. of Appl. Math.* **21,** 315–324 (1964).
18. J. A. HILLIER AND R. ROTHERY, "The Synchronization of Traffic Signals for Minimum Delay," *Trans. Sci.* **1,** 81–94 (1967).
19. N. GARTNER, "Microscopic Analysis of Traffic Flow Patterns for Minimizing Delay on Signal Controlled Links," *Highway Research Record No. 445,* 12–23 (1973).
20. K. W. HUDDART, "The Importance of Stops in Traffic Signal Progressions," *Trans. Res.* **3,** 143–150 (1969).
21. C. C. CHUNG AND N. GARTNER, "Acceleration Noise as a Measure of Effectiveness in the Operation of Traffic Control Systems," Working Paper OR 015-73, Operations Research Center, M.I.T., Cambridge, Mass., March 1973.
22. K. W. HUDDART AND E. D. TURNER, "Traffic Signal Progressions—G.L.C. Combination Method," *Traffic Eng. and Control* **11,** 320–322 (1969).
23. R. WIEDEMANN, "Verkehrsablauf hinter Lichtsignalanlagen," *Strassenbau und Strassenverkehrstechnik,* Heft **74,** Bundesminister für Verkehr, Bonn, Germany, 1968.
24. Z. A. NEMETH AND R. L. VECELLIO, "Investigation of the Dynamics of Platoon Dispersion," *Highway Research Record No. 334,* 23–33 (1970).
25. N. GARTNER, J. D. C. LITTLE, AND H. GABBAY, "Optimization of Traffic Signal Settings in Networks by Mixed-Integer Linear Programming," Technical Report No. 91, Operations Research Center, M.I.T., Cambridge, Mass., March 1974.
26. G. F. NEWELL, "Traffic Signal Synchronization for High Flows on a Two-Way Street," *Proc. Fourth Intern. Symp. on the Theory of Traffic Flow,* University of Karlsruhe, 87–92 (1968).
27. H. INOSE, H. FUJISAKI, AND T. HAMADA, "Theory of Road-Traffic Control Based on Macroscopic Traffic Model," *Electronics Letters* **3,** 385–386 (1967).
28. D. C. GAZIS, "Traffic Control, Time-Space Diagrams, and Networks," *Traffic Control—Theory and Instrumentation* (T. R. Horton, ed.), Plenum Press, New York, 1965.
29. M. J. EISNER, "Synchronizing Traffic Signals in a Saturated Network: A Deterministic Optimization Model," Technical Report No. 155, Dept. of Operations Research, Cornell University, Ithaca, N. Y., August 1972.
30. M. SPRINGER, "Signal Offset and Split Optimization by Decomposition Methods," Report No. KLD TR-12, KLD Associates, Huntington, N. Y., October 1973.
31. N. GARTNER, "Constraining Relations Among Offsets in Synchronized Signal Networks," *Trans. Sci.* **6,** 88–93 (1972).
32. J. T. WELSCH, "A Mechanical Analysis of the Cyclic Structure of Undirected Linear Graphs," *J. ACM* **13,** 205–210, 1966.

(Received, March 1975; revised, June 1975)