# MAXBAND: A Program for Setting Signals on Arteries and Triangular Networks 

JOHN D. C. LITTLE, MARK D. KELSON, AND NATHAN H. GARTNER


#### Abstract

MAXBAND is a portable, off-line, FORTRAN IV computer program for setting arterial signals to achieve maximal bandwidth. Special features of the program include (a) automatically choosing cycle time from a given range, (b) permitting the design speed to vary within given tolerances, (c) selecting the best lead or lag pattern for left-turn phases from a specified set, (d) allowing a queue clearance time for secondary flow accumulated during red, (e) accepting user-specified weights for the green bands in each direction, and (f) handling a simple network in the form of a three-artery triangular loop. Green splits can be provided or, alternatively, flows and capacities can be given and splits calculated by using Webster's theory. The program produces cycle time, offsets, speeds, and order of left-turn phases to maximize the weighted combination of bandwidths. The optimization uses Land and Powell's MPCODE branch and bound algorithm. As many as 12 signals can be handled efficiently. The program is available from the Federal Highway Administration.


Signal-setting methods for fixed-time traffic-control systems separate broadly into two classes. The first, and historically oldest, consists of methods that maximize bandwidth and progression. This group includes, among others, those of Little and Morgan (1), Little (2) and Messer and others (3). The second group contains methods that seek to minimize delay, stops, or other measures of disutility. Examples are Hillier's combination method (4), Traffic Research Corporation's SIGOP (5), Robertson's TRANSYT (6), Gartner, Little, and Gabbay's MITROP (ㄱ), and Lieberman and Woo's SIGOP II (8).

Although aisutility-oriented methous have now been available for some time, many traffic engineers continue to prefer maximal bandwidth settings because they have certain inherent advantages. For one thing, bandwidth methods use relatively little input; the basic requirements are street geometry, traffic speeds, and green splits. In addition, progression systems are operationally robust. Spacetime diagrams let the traffic engineer visualize easily the quality of the results. Through accumulated experience, engineers who have knowledge of the specific streets can spot problems and, if necessary, make modifications to the settings. Furthermore, various studies [for example, the study by Wagner, Gerlough, and Barnes (g)] and much practical experience have shown that bandwidth systems give good results in the field. It may even be that drivers expect signal progression and take it as a measure of setting quality. In any case we take the position here that, if engineers are going to use bandwidth systems, they should have the best.

Morgan and Little (I) first computerized the setting of arterial signals for maximal bandwidth. The widely distributed program of Little, Martin, and Morgan (10) efficiently finds offsets for maximal bandwidth given cycle time, red times, signal distances, and street speed. The total bandwidth attained can be allocated between directions on the basis of flow.

Little (2) subsequently generalized the computation in several ways: The cycle time could be automatically selected from a given range, and so could speed. Networks could be solved. These and several further extensions became possible through a mixedinteger formulation of the problem.

The flexibility thereby introduced has several advantages. For example, maximal bandwidth calculations frequently have a disconcerting feature. On a
long street, the signals that constrict bandwidth may turn out to be far apart. A small change in design speed or cycle time can produce quite different signal settings and bandwidths. Yet drivers do not hold their speeds exactly constant and, as shown by Desrosiers and Leighty (11), they tend to adjust their speeds to the signals. Therefore, it makes sense to let design speed between signals be a variable, at least within certain limits. Similarly, it is helpful to be able to consider a range of possible cycle times automatically and determine which one combines best with the street geometry to yield maximal bandwidth. The mixed-integer formulation permits this.

The approach has not become popular, however, for two principal reasons:

1. A person must invest substantial effort in learning how to formulate and solve problems in this way.
2. At the time the paper by Little (2) appeared, the solving of mixed-integer problems was inefficient and expensive. Since then, however, methods for solving mixed-integer problems have become better and large-scale computations have become cheaper.

Further research described here reveals that the mixed-integer formulation extends to multiphase signals. For example, asymmetric reas that occur when the green is delayed for left turns can easily be introduced into the formulation. The decision whether to put a left-turn arrow at the beginning or end of the green can be assigned to the optimization and resolved in whichever way maximizes total bandwidth.

Messer and others (3) have also developed a program, PASSER II, to consider this last issue. An advantage of the present approach is that the multiphase feature is combined with the flexibilities mentioned earlier into a single formulation that can be pursued to a mathematically proved optimum. A further extension is of potential value when secondary flows are significant. Turn traffic entering the artery at a previous intersection may build a queue that interferes with the progression. In this case, a time advance can be put into the through band to permit the queue to clear the intersection before the through platoon arrives.

A portable FORTRAN IV computer program has been developed that maximizes bandwidth according to methods described by Little (2) and extended here. The program, called MAXBAND, is designed to handle arteries and simple three-artery networks that contain as many as 17 signals. The program has been documented in a series of reports for the Federal Highway Administration (FHWA) (12).

## METHODOLOGY

## Optimization

The optimization formulation draws and generalizes on Little (2). The basic geometry is shown in Figure l. Let
$b(\bar{b})=$ outbound (inbound) bandwidth (cycles);

## Figure 1. Space-time diagram showing green bands.



$$
\begin{aligned}
& r_{i}\left(\bar{r}_{i}\right)= \text { ith signal (i }=\text { outbound (inbound) red time } \\
& \text { at } S_{i} \text { (cycles); } \\
& w_{i}\left(\bar{w}_{i}\right)= \text { time from right (left) side } \\
& \text { of red at } S_{i} \text { to left (right) } \\
& \text { edge of outbound (inbound) green } \\
& \text { band (cycles); } \\
& t(h, i)[\bar{t}(h, i)]= \text { travel time from } S_{h} \text { to } S_{i} \\
& \text { outbound ( } S_{i} \text { to } S_{h} \text { inbound) } \\
& \text { (cycles); } \\
& \phi(h, i)[\bar{\phi}(h, i)]= \text { time from center of an } \\
& \text { outbound (inbound) red at } S_{h} \\
& \text { to the center of a particular } \\
& \text { outbound (inbound) red at } S_{i} \\
& \text { (cycles) fthe two reds are } \\
& \text { chosen so that each is } \\
& \text { immediately to the left (right) } \\
& \text { of the same outbound (inbound) } \\
& \text { green band; } \phi(h, i) \\
& {[\phi(h, i)] \text { is positive if the } } \\
& S_{i} \text { center of red lies to the } \\
& \text { to the right (left) of the } S_{h} \\
& \text { center of red); } \\
& \Delta_{i}= \text { time from center of } \bar{r} \bar{r}_{i} \text { to } \\
& \text { nearest center of } r_{i}(c y c l e s) \\
& \text { (positive if center of } r_{i} \text { is to } \\
&\text { the right of center of } \left.r_{i}\right) ; \text { and } \\
&= \text { queue clearance time, an advance } \\
& \text { of the outbound (inbound) band- } \\
& \text { width upon leaving } S_{i} \text { (cycles) }
\end{aligned}
$$

The fundamental equation for formulating the arterial problem arises from a physical constraint. It is derived with the help of Figure 1 by expressing the difference in time from $A$ to $B$ in two different ways: First, by using outbound-defined quantities, time $A$ to $B=\Delta_{h}+$ integer number of cycles $+\phi(h, i)$. Then, by using inbound-defined quantities, time $A$ to $B=$ integer number of cycles - $\bar{\phi}(h, i)+$ another integer number of cycles $+\Delta_{i}$. Setting these times equal, rearranging, and coalescing the integers into a single variable, m(h,i),
$\phi(\mathrm{h}, \mathrm{i})+\bar{\phi}(\mathrm{h}, \mathrm{i})+\Delta_{\mathrm{h}}-\Delta_{\mathrm{i}}=m(\mathrm{~h}, \mathrm{i})$
We call $m(h, i)$ the loop integer in recognition of the more general case of networks. The terminology applies in the present case because the links $S_{h}$ to $S_{i}$ and $S_{i}$ to $S_{h}$ form a loop and Equation 1 states that the sum of times around the loop is an integer number of cycles.

From Figure l, we also read from C to D:
$\phi(\mathrm{h}, \mathrm{i})+(1 / 2) \mathrm{r}_{\mathrm{i}}+\mathrm{w}_{\mathrm{i}}+\tau_{\mathrm{i}}=(1 / 2) \mathrm{r}_{\mathrm{h}}+\mathrm{w}_{\mathrm{h}}+\mathrm{t}(\mathrm{h}, \mathrm{i})$
and from $\overline{\mathrm{C}}$ to $\overline{\mathrm{D}}$ :
$\bar{\phi}(\mathrm{h}, \mathrm{i})+(1 / 2) \overline{\mathrm{r}}_{\mathrm{i}}+\overline{\mathrm{w}}_{\mathrm{i}}=(1 / 2) \overline{\mathrm{r}}_{\mathrm{h}}+\overline{\mathrm{w}}_{\mathrm{h}}-\bar{\tau}_{\mathrm{h}}+\overline{\mathrm{t}}(\mathrm{h}, \mathrm{i})$
Substituting Equation 2 into Equation 1 to eliminate $\phi$ and $\bar{\phi}$ gives

$$
\begin{align*}
& \mathrm{t}(\mathrm{~h}, \mathrm{i})+\overline{\mathrm{t}}(\mathrm{~h}, \mathrm{i})+(1 / 2)\left(\mathrm{r}_{\mathrm{h}}+\overline{\mathrm{r}}_{\mathrm{h}}\right)+\left(\mathrm{w}_{\mathrm{h}}+\overline{\mathrm{w}}_{\mathrm{h}}\right)-(1 / 2)\left(\mathrm{r}_{\mathrm{i}}+\overline{\mathrm{r}}_{\mathrm{i}}\right) \\
& \quad-\left(\mathrm{w}_{\mathrm{i}}+\bar{w}_{\mathrm{i}}\right)-\left(\tau_{\mathrm{i}}+\bar{\tau}_{\mathrm{h}}\right)+\Delta_{\mathrm{h}}-\Delta_{\mathrm{i}}=\mathrm{m}(\mathrm{~h}, \mathrm{i}) \tag{3}
\end{align*}
$$

So far we have required that $S_{i}$ follow $S_{h}$ in the outbound direction, but this restriction is not necessary. For physical reasons we wish $t($.$) to$ satisfy $t(h, j)=t(h, i)+t(i, j) ;$ by reason of which, setting $h=j$, we shall require $t(i, h)=-t(h, i)$ and, by a similar argument, $\bar{t}(i, h)=-\bar{t}(h, i)$. With these relations, Equations 2 and 3 hold for arbitrary $\mathrm{S}_{\mathrm{h}}$ and $\mathrm{S}_{\mathrm{i}}$, and
$\phi(\mathrm{h}, \mathrm{j})=\phi(\mathrm{h}, \mathrm{i})+\phi(\mathrm{i}, \mathrm{j}) \quad \phi(\mathrm{h}, \mathrm{i})=-\phi(\mathrm{i}, \mathrm{h})$
$m(h, j)=m(h, i)+m(i, j) \quad m(h, i)=-m(i, h)$
along with corresponding expressions for $\bar{\phi}$.
Notation becomes simpler if the signals are numbered sequentially from $l$ to $n$ in the outbound direction. Then define $x_{i}=x(i, i+l)$ for $x=t$, $\overline{\mathrm{t}}, \mathrm{m}, \phi, \bar{\phi}$. Now Equation 3 gives

$$
\begin{align*}
\mathrm{t}_{\mathrm{i}} & +\overline{\mathrm{t}}_{\mathrm{i}}+\left(\mathrm{w}_{\mathrm{i}}+\bar{w}_{\mathrm{i}}\right)-\left(\mathrm{w}_{\mathrm{i}+1}+\bar{w}_{\mathrm{w}+1}\right)+\Delta_{\mathrm{i}}-\Delta_{\mathrm{i}+1}=-(1 / 2)\left(\mathrm{r}_{\mathrm{j}}+\overline{\mathrm{r}}_{\mathrm{i}}\right) \\
& +(1 / 2)\left(\mathrm{r}_{\mathrm{i}+1}+\overline{\mathrm{r}}_{\mathrm{i}+1}\right)+\left(\bar{\tau}_{\mathrm{i}}+\tau_{\mathrm{i}+1}\right)+\mathrm{m}_{\mathrm{i}} \tag{6}
\end{align*}
$$

From Figure 1 we also see that
$w_{i}+b \leqslant 1-r_{i}$
$\bar{w}_{i}+\bar{b} \leqslant 1-\bar{r}_{i}$

If for the moment we also require $b=\vec{b}$, we can collect Equation 6 and Equation 7 into a basic mixed-integer linear program (LPI) for setting arterial signals.

LPl: Find $b, \bar{b}, w_{i}, \bar{w}_{i}$, and $m_{i}$ to max $b$, subject to

$$
\left.\left.\begin{array}{l}
\bar{b}=b \\
w_{i}+b \leqslant 1-r_{i} \\
\bar{w}_{i}+\bar{b} \leqslant 1-\bar{r}_{i}
\end{array}\right\} \quad i=1, \ldots, n\right) \quad \begin{aligned}
& \left(w_{i}+\bar{w}_{i}\right)-\left(w_{i+1}+\bar{w}_{i+1}\right)+\left(t_{i}+\bar{t}_{i}\right)+\Delta_{i}-\Delta_{i+1}=-(1 / 2)\left(r_{i}+\bar{r}_{i}\right) \\
& \quad+(1 / 2)\left(r_{i+1}+\bar{r}_{i+1}\right)+\left(\bar{\tau}_{i}+\tau_{i+1}\right)+m_{i} \quad i=1, \ldots, n-1
\end{aligned}
$$

$\mathrm{m}_{\mathrm{j}}=$ integer
$b, \bar{b}, w_{i}, \bar{w}_{i} \geqslant 0 \quad i=1, \ldots, n$

LPl has $3 n$ constraints, $2 n+2$ continuous variables, and $n-1$ unrestricted integer variables.

In the formulation of LPl , the green band is defined on departure from the signal. Therefore, when queue clearance times are introduced, the jog put into the band will, under some circumstances, cause the tail of the arriving band to hit red. If desired, the green band can be defined so as to require room for both the arriving green band and the queue clearance jog. This can be done by adding $t_{i}$ and $\bar{t}_{i}$, respectively, to the left-hand sides of Equations 7a and 7b (or 13 a and l3b below). This change may reduce the bandwidth somewhat.

We next introduce a generalization that permits the optimization program to decide when the leftturn phase (if one is present) will occur with respect to the through green at any signal. The

Figure 2. The four possible patterns of left-turn phases.

1. Outbound lelt leads; inbound lags

2. Oulbound left lags inbound leads


3 Outbound left leads; inbound leads

4. Oulbound left lags; inbound lags

left-turn green can be picked to lead or lag, whichever gives the most total bandwidth. At the same time, however, the traffic engineer must be able to specify which of the possible combinations of leads or lags will be permitted in a given instance.

Figure 2 shows the four possible patterns of left-turn green phases. Let

$$
\begin{aligned}
g_{i}\left(\bar{g}_{i}\right)= & \text { outbound (inbound) green time for through } \\
& \text { traffic at } s_{i} \text { (cycles), } \\
\ell_{i}\left(\bar{\ell}_{i}\right)= & \text { time allocated for outbound (inbound) } \\
& \text { left-turn green at } s_{i} \text { (cycles), and } \\
R= & \text { common red time in both directions to } \\
& \text { provide for cross-street movements } \\
& \text { (cycles). }
\end{aligned}
$$

Since the time allocated to outbound left-turn green is inbound red time, we have (see Figure_2) $\underline{r}_{i}=$ $R+\bar{l}_{i}, \bar{r}_{i}=R+l_{i}$ and $\bar{r}_{i}+g_{i}=1, \bar{r}_{i}+\bar{g}_{i}=$ 1. Moreover, ${ }^{\text {we }}$ can express $\Delta_{i}$, the time from the center of $\bar{r}_{i}$ to the next center of $r_{i}$, in terms of $\ell_{i}$ and $\bar{x}_{i}$ for each case as follows:

| Pattern | $\Delta_{i}$ |
| :---: | :---: |
| 1 | -(1/2) ( $\left.\ell_{i}+\bar{\ell}_{i}\right)$ |
| 2 | $(1 / 2)\left(\ell_{i}+\ell_{i}\right)$ |
| 3 | -(1/2) ( $\left.x_{i}-\bar{k}_{i}\right)$ |
| 4 | $(1 / 2)\left(l_{i}-\bar{l}_{i}\right)$ |

All of these can be expressed in the following form:
$\Delta_{i}=(1 / 2)\left[\left(2 \delta_{i}-1\right) \ell_{i}-\left(2 \bar{\delta}_{i}-1\right) \bar{\ell}_{i}\right]$
where $\delta_{i}$ and $\bar{\delta}_{i}$ are $0-1$ variables and the previous cases are now picked out by

Pattern

$$
\frac{\delta_{i}}{0} \quad \frac{\bar{\delta}_{i}}{1}
$$

| Pattern |  | $\delta_{\mathbf{i}}$ |  |
| :--- | :--- | :--- | :--- |
|  |  |  | $\bar{\delta}_{\mathbf{i}}$ |
| 3 |  |  | 0 |
| 4 | 1 |  | 1 |

Therefore, we can use the mixed-integer program to select the pattern that will maximize bandwidth. If only certain patterns are to be permitted, restriction can be placed on the $\delta_{i}$ and $\bar{\delta}_{i}$ to enforce the requirements. For example, if only patterns 1 and 2 are permitted, the constraint $\delta_{i}+\delta_{i}=1$ is added.

We often wish to let the user favor one direc-tion--say, by manipulating the ratio of inbound to outbound bandwidth. For example, this ratio might be set to that of the two flows. Such a requirement is easily built into the LP as a constraint. However, in making one green band larger than the other, we can never make it larger than the smallest green in that direction. Once this is achieved, it is foolish to restrict the opposite direction further just to satisfy the ratio. Therefore, we speak of a target ratio.

Let $k=$ target $r$ atio of inbound to outbound bandwidth. For the case of $k<1$ (outbound favored), we can set up the objective function and ratio constraint as follows: max $(b+k b)$, subject to $\bar{b} \geq k b$. The $k>1$ case is also accommodated if we change the formulation to $\max (b+k b)$, subject to
$(1-k) \bar{b} \geqslant(1-k) k b$
For $k=1$, the last inequality must be replaced by $\mathrm{b}=\overline{\mathrm{b}}$.

A further set of generalizations is possible. One of the most important is to let signal period (cycle length) and speed be variables. Each will be constraineu by upper and lower limits. Iñ addition, changes in speed from one street segment to the next can be limited. Let

$$
\begin{aligned}
T= & \text { cycle length (signal } \\
& \text { period) (s); } \\
z= & 1 / T=\text { signal frequency } \\
& \text { (cycles } / \mathrm{s}) ; \\
\mathrm{T}_{1}, \mathrm{~T}_{2}= & \text { lower and upper limits } \\
& \text { on cycle length, i.e., } \\
& \mathrm{T}_{1} \leq \mathrm{T} \leq \mathrm{T}_{2}(\mathrm{~s}) ; \\
\mathrm{a}(\mathrm{~h}, \mathrm{i})[\overline{\mathrm{d}}(\mathrm{~h}, \mathrm{i})]= & \text { distance between } \mathrm{S}_{\mathrm{h}} \text { and } \mathrm{s}_{\mathrm{i}} \\
& \text { outbound (inbound) }(\mathrm{m}) ; \\
\mathrm{d}_{\mathrm{i}}= & \mathrm{d}(\mathrm{i}, \mathrm{i}+1), \mathrm{d}_{\mathrm{i}}=\mathrm{d}(\mathrm{i}, \\
& \mathrm{i}+1) ;
\end{aligned}
$$

We are constraining change in speed by putting upper and lower limits on change in reciprocal speed.
 serves to prevent large, abrupt speed changes. Reciprocal speed is used because it enters linearly in the constraints and can be transformed into $t_{i}$. Thus,
$\mathrm{t}_{\mathrm{i}}=\left(\mathrm{d}_{\mathrm{i}} / \mathrm{v}_{\mathrm{i}}\right) \mathrm{z}$
$\bar{t}_{i}=\left(\bar{d}_{i} / \bar{v}_{i}\right) z$
In the expanded formulation, $t_{i}, \bar{t}_{i}$, and $z$ are decision variables that, once known, determine progression speeds.

We add to LPl all of these generalizations to yield a more versatile_mixed-integer linear program.

LP2: find $b, \bar{b}, z, w_{i}, t_{i}, t_{i}, \delta_{i}, \delta_{i}$, and $m_{i}$ to $\max (b+k \bar{b})$, subject to
$(1-k) \bar{b} \geqslant(1-k) k b$
$1 / \mathrm{T}_{2} \leqslant \mathrm{z} \leqslant 1 / \mathrm{T}_{1}$
$\left.\begin{array}{l}w_{i}+b \leqslant 1-r_{i} \\ \bar{w}_{i}+\bar{b} \leqslant 1-\bar{I}_{i}\end{array}\right\} \quad i=1, \ldots, n$
$\left(w_{i}+\bar{w}_{i}\right)-\left(w_{i+1}+\bar{w}_{i+1}\right)+\left(t_{i}+\bar{t}_{i}\right)+\delta_{i} \ell_{j}-\bar{\delta}_{i} \bar{\ell}_{i}-\delta_{i+1} \ell_{i+1}+\bar{\delta}_{i+1} \bar{\ell}_{i+1}-m_{i}$
$\quad=\left(r_{i+1}-r_{i}\right)+\left(\bar{\tau}_{i}+\tau_{j+1}\right) \quad i=1, \ldots, n-1$
$\left.\left(\mathrm{d}_{\mathrm{i}} / \mathrm{f}_{\mathrm{i}}\right) \mathrm{z} \leqslant \mathrm{t}_{\mathrm{i}} \leqslant\left(\mathrm{d}_{\mathrm{i}} / \mathrm{e}_{\mathrm{i}}\right) \mathrm{z}\right\}$
$\left.\left(\bar{d}_{i} / \bar{f}_{\mathrm{i}}\right) \mathrm{z} \leqslant \overline{\mathrm{t}}_{\mathrm{i}} \leqslant\left(\overline{\mathrm{d}}_{\mathrm{i}} / \overline{\mathrm{e}}_{\mathrm{i}}\right) \mathrm{z}\right\}$
$\mathrm{i}=1, \ldots, \mathrm{n}-1$
$\left.\left(\mathrm{d}_{\mathrm{i}} / \mathrm{h}_{\mathrm{i}}\right) \mathrm{z} \leqslant\left(\mathrm{d}_{\mathrm{i}} / \mathrm{d}_{\mathrm{i}+1}\right) \mathrm{t}_{\mathrm{i}+1}-\mathrm{t}_{\mathrm{i}} \leqslant\left(\mathrm{d}_{\mathrm{i}} / \mathrm{g}_{\mathrm{i}}\right) \mathrm{z}\right\}$
$\left.\left(\overline{\mathrm{d}}_{\mathrm{i}} / \overline{\mathrm{h}}_{\mathrm{j}}\right) \mathrm{z} \leqslant\left(\overline{\mathrm{d}}_{\mathrm{i}} / \mathrm{d}_{\mathrm{i}+1}\right) \bar{t}_{\mathrm{i}+1}-\overline{\mathrm{t}}_{\mathrm{i}} \leqslant\left(\overline{\mathrm{d}}_{\mathrm{i}} / /_{\mathrm{i}}\right) \mathrm{z}\right\}$
$i=1, \ldots, n-2$
$\mathrm{b}, \overline{\mathrm{b}}, \mathrm{z}, \mathrm{w}_{\mathrm{i}}, \bar{w}_{\mathrm{i}}, \mathrm{t}_{\mathrm{i}}, \overline{\mathrm{t}}_{\mathrm{i}} \geqslant 0$
$\mathrm{m}_{\mathrm{i}}=$ integer
$\delta_{\mathbf{i}}, \bar{\delta}_{\mathbf{i}}=0,1$
where, if $k=1$, Equation 11 is replaced by $\bar{b}=b$.
LP2 involves ( $11 \mathrm{n}-10$ ) constraints and $(4 n+1)$ continuous variables, up to $2 n(0-1)$ variables and n - 1 unrestricted integer variables, not counting slack variables. In addition, if the user decides to require or prohibit certain left-turn patterns, constraints on $\dot{o}_{i}$ and $\vec{\delta}_{i}$ are added up to a maximum of 2 n .

LP2 describes how the arterial case is solved. The triangular loop consists of three arteries. Its mathematical program consists of (a) an objective function that is a weighted combination of the objective functions of the individual LP2's the weights being set by the user to express the relative importance of bandwidth on the three arteries), (b) all the constraints of the individual LP2's, and (c) the loop constraint. The loop constraint is
$\sum_{(\mathrm{i}, \mathrm{j}) \in \mathrm{L}} \psi_{\mathrm{ij}}=n_{\mathrm{L}}$
where $\psi_{i j}$ is the offset (in cycles) for link (i,j) in loop $L$ and $n_{L}$ is the loop integer.

## Determining Green Splits

One option in MAXBAND is for the user to supply the green splits. As an alternative, the user can provide traffic volume and capacity information and the program will calculate the splits. This is done essentially by using the theory of Webster (13), who has shown that under certain circumstances total delay at an intersection is minimized by dividing the available cycle time among competing streams of traffic proportional to their volumes divided by their capacities.

In MAXBAND, the user who wishes to use this option provides volume and capacity information for the traffic, classified into four through movements and four left-turn movements for each intersection. Let

TRAT(i) = through-traffic ratio of volume to capacity in direction i,
LRAT(i) $=$ left-turn traffic ratio of volume to capacity in direction i, and
$i=$ OUT, IN, OUTC, INC $=$ outbound main,
inbound main, outbound cross street, and inbound cross street.

The procedure calculates
MAIN $=$ max [TRAT (OUT) + LRAT(IN), TRAT(IN) + LRAT (OUT)] = the larger of through volume/capacity plus opposite left-turn volume/capacity for the two directions on the main street.
CROSS $=$ max $[$ TRAT (OUTC) + LRAT (OUTC), TRAT (INC) + LRAT (OUTC)] $=$ the same quantity for the cross street.

The basic split between main street and cross street is
$M M=$ MAIN/(MAIN + CROSS $)=$ total time allocated to main street (fraction of cycle).
CC $=$ CROSS $/($ MAIN + CROSS $)=$ total time allocated to cross street (fraction of cycle).
Let $L$ (OUT) $[L(I N)]=$ outbound (inbound) left split and $G$ (OUT) [G(IN)] = outbound (inbound) through split. Then,
L (OUT) $=\{\operatorname{LRAT}($ OUT $) /[\operatorname{TRAT}(O U T)+\operatorname{LRAT}($ IN $)] \times \mathrm{MM}$.
$\mathrm{L}(\mathrm{IN})=\{$ LRAT $($ IN $) /[$ TRAT $($ OUT $)+\operatorname{LRAT}($ IN $)]\} \times \mathrm{MM}$.
$G(O U T)=M M-L(I N)$.
$G(I N)=M M-L$ (OUT).
If necessary, these splits are then modified slightly to meet minimum green requirements.

## COMPUTER PROGRAM

The computer program consists of (a) a control section, (b) an input section that accepts the problem data from the user, (c) a matrix generator that transforms the input into a form usable by the mixed-integer linear program, (d) a mathematical programming package, and (e) an output routine that interprets the mathematical programming results and prints them out in a form usable by a traffic engineer. The program contains approximately 11000 lines of FORTRAN IV code, broken down as follows:

| Item | No. of Lines |
| :--- | :---: |
| Control program | 100 |
| Input program | 3200 |
| Matrix generator | 3200 |
| MPCODE | 2500 |
| Output program | 2000 |
| Control Program |  |

The control program manages the overall computation, calling each of the other programs as needed.

## Input

Input to the program is on IBM cards or a corresponding card image file on another medium such as magnetic tape or disk. The basic inputs are as follows:

1. Overall problem information--The overall problem information includes a name for the run, an indicator for whether it is a loop problem or an artery problem, an indicator for whether metric or customary units are used, the acceptable range of cycle lengths, and the target ratio for the bandwidths on each artery and their weights, unless these are to be computed from volume information. Usually, default values are used in the mathematical programming package for certain parameters, such as the maximum number of iterations and reinversions, but, if the user wishes, these can be supplied as part of the overall problem information.
2. Network geometry--The order of signals on each artery is given, along with the distances between them (which may be different in each direction) and the names of their intersections. In the case of a loop, the intersection numbers at artery meetings are specified.
3. Green splits or traffic flows and capaci-ties--The user may specify the green splits at each signal as a fraction of the overall cycle time. Alternatively, traffic flows and capacities can be given for each link, including cross streets and turning movements, and the program will calculate green splits by using Webster's formula.
4. Left-turn patterns--Left-turn phases can occur at the beginning or end of the green in either direction, creating four possible patterns for the through direction at each intersection. The user can specify which of the patterns are acceptable, and the program will choose among them to maximize bandwidth.
5. Queue clearance time--Queues may build up during red time as a result of turning movements onto the artery at previous intersections. Such queues may impede the flow of vehicles in the through band. The user may therefore specify at any intersection in either direction a queue clearance time as a fraction of the cycle length. The program will adjust the through band to arrive at the intersection after the queue has cleared and leave the intersection with the queue included as part of the band. In effect, this puts a jog into the through band, advancing it upon leaving the intersection by an amount equal to the queue clearance time.
6. Range of speed--For each link or, if preferred, for the artery as a whole, the user specifies a design speed and a speed tolerance. The program then chooses speeds for each link from this range so as to maximize bandwidth. In addition, the user can constrain the change in speed from one link to the next. If the user does not set limits on speeds and speed changes, default values of 10 percent are used.

More detailed specification of inputs is found in the MAXBAND User's Manual (12).

## Matrix Generator

The mathematical program LP2 is a special case of the general linear mixed-integer problem: max $c x$, subject to
$\mathrm{Ax} \leq \mathrm{b}$
$x \geqslant 0$
$\mathrm{X}_{\mathrm{j}}$ integer $\mathrm{j} \in \mathrm{J}$
where $x$ is an $n$-vector of variables whose values are sought, $c$ is an $n$-vector of objective functions coefficients, A is an ( $m \times n$ ) matrix of coefficients, $b$ is an m-vector of right-hand-side constants, and $J$ is a set of subscripts identifying the variables required to be integer.

The traffic problem described earlier as LP2 re-
 will change as the traffic problem changes. It is necessary to have a generalized program, called the matrix generator, that will take the traffic input data and convert them into the appropriate vectors and matrices for input to the mathematical programming package.

## Mathematical Programming Package

A key part of the computer program is the routine for solving the mixed-integer linear program. After
several linear programming codes were examined and comparative runs were made on two mixed-integer packages, MPCODE was selected. MPCODE is available in FORTRAN IV source code and is superbly documented in a hard-cover book by Land and Powell (14).

## Output

The output of the program is divided into three parts: input cards, a data summary, and a solution report.

The input-card section is a simple printing of the input cards. The data-summary report contains the following information:

1. MPCODE values used by the Land and Powell system;
2. For an artery, (a) general information such as the name of the artery, the number of signals, limits on cycle length, units, target bandwidth ratio, and bandwidth weights; (b) arterywide values such as design speed, tolerances, and limits on changes between links; (c) intersection values, including splits with an indication of their origin, queue clearance times, minimum greens, and the permitted patterns for left turns; (d) link values as actually used, including length, design speed, and speed tolerances; and (e) volumes and capacities on all approaches, when provided.
3. For a loop, (a) general loop information, including upper and lower limits on cycle time and where the arteries meet, and (b) the same artery information described above but for each of the three arteries.

The solution report presents the following data:

1. An indicator for whether the problem has been solved successfully;
2. MPCODE statistics describing the number of iterations, etc., used by the Land and Powell algorithm to solve the problem ("number of solutions" is the number of integer solutions, including the optimal integer solution, found for the problem);
3. For an artery, (a) general information, including name of artery, number of signals, and type of units; (b) cycle time and bandwidths; (c) left patterns selected as optimal; (d) duration and offsets of splits in both fractions of a cycle and seconds; and (e) traversal times and speeds on links;
4. For a loop, (a) loop information, including chosen cycle time, bandwidths, and objective function; (b) same information as for a single artery for each of the three arteries; and (c) repeat of duration and offsets of splits for signals at artery meetings.

Examples of outputs for several test problems are given elsewhere (12).

## TESTING

The testing of the program has included runs on a wide variety of problems and operation on several computers.

Table 1 gives run statistics for 10 arterial problems and 3 loop problems. The number of variables, constraints, and integer variables (total and free) relate to the mixed-integer program and represent measures of program difficulty. The number of branch and bound iterations is another measure of how much computation the program required. The number of solutions is the number of feasible integer solutions discovered, up to and including the optimal one. Input data for several of the problems can be found in Little and Kelson (12).

Table 1. MAXBAND performance statistics.

| Problem Type | Problem | Problem Characteristics |  |  |  |  | Solution Characteristics |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  | No. of Signals | No. of Variables | No. of Constraints | No. of Integer Variables |  |  |  |  |  |
|  |  |  |  |  |  |  | No. of Iterations ${ }^{\text {a }}$ | No. of Solutions ${ }^{\text {b }}$ | CPU <br> Time <br> (s) | Cost <br> (\$) |
|  |  |  |  |  | Total | Free |  |  |  |  |
| Arterial | Broadway, Cambridge, MA | 5 | 25 | 45 | 4 | 4 | 36 | 1 | 2.17 | 0.73 |
|  | Voorhees scenario $1^{\text {c }}$ | 6 | 30 | 56 | 5 | 5 | 668 | 2 | 5.81 | 1.07 |
|  | Short version, Waltham artery | 6 | 33 | 56 | 8 | 8 | 197 | 3 | 3.23 | 0.76 |
|  | Voorhees scenario $2^{\text {c }}$ | 6 | 36 | 56 | 11 | 11 | 957 | 6 | 7.01 | 1.23 |
|  | Voorhees scenario $2^{\text {c }}$, computed splits | 6 | 36 | 56 | 11 | 11 | 1457 | 5 | 9.74 | 1.52 |
|  | Modified Voorhees scenario $2^{\text {c }}$, computed splits | 6 | 42 | 56 | 17 | 17 | 607 | 7 | 6.26 | 1.35 |
|  | Voorhees scenario 1 ${ }^{\text {c }}$, computed splits | 6 | 42 | 56 | 17 | 17 | 2089 | 6 | 14.08 | 2.03 |
|  | Modified Waltham artery | 11 | 60 | 116 | 15 | 10 | 1296 | 4 | 17.16 | 2.44 |
|  | Waltham artery | 11 | 60 | 111 | 15 | 15 | 3781 | 7 | 44.79 | 5.69 |
|  | Wisconsin Avenue, Washington, D.C. | 17 | 88 | 177 | 19 | 19 | 8700 | 12 | $\sim 210.00$ | $\sim 25.00$ |
| Loops | Modified Attleboro loop | 4 | 36 | 44 | 5 | 5 | 286 | 3 | 3.69 | 0.84 |
|  | Attleboro loop | 4 | 37 | 44 | 5 | 5 | 432 | 4 | 4.44 | 0.92 |
|  | FHWA test network | 15 | 93 | 168 | 19 | 16 | 32885 | 15 | 628.81 | 74.87 |

${ }^{\mathrm{a}}$ Total number of simplex iterations used. $\quad{ }^{\mathrm{b}}$ Total number of feasible integer solutions found. $\quad{ }^{\mathrm{c}}$ Test arterials provided by FHWA.

Figure 3. Eleven-signal test problem: Main Street, Waltham, Massachusetts.


Figure 4. Main Street, Waltham: outbound green band.


The central processing unit (cpu) time is the number of seconds required on the Massachusetts Institute of Technology (MIT) IBM 370-168 and is the primary performance measure. As can be seen, most problems have been solved in a few seconds, although one problem, the 15-signal FHWA test network loop problem, took 10.5 min. The cost shown is the cost charged by the MIT Information Processing Service.

The settings developed by MAXBAND have been put into a traffic simulation program (NETSIM) for several of the test networks with good results. In addition, the MAXBAND output can be used to construct a space-time plot. Figure 3 shows Main Street, Waltham, Massachusetts. Figures 4 and 5

Figure 5. Main Street, Waltham: inbound green band.

show space-time diagrams for its MAXBAND-generated settings. In Figure 4, note the slight changes in slope resulting from 10 percent permitted variation in design speed. In Figure 5, inbound red times differ from outbound because of left-turn phases.

The MAXBAND program has been transmitted to FHWA on magnetic tape and has been operated on computers to which FHWA has access.

## DOCUMENTATION

Little and Kelson (12) provide documentation for MAXBAND in three volumes. Volume 1 , the Summary Report, provides an overview of MAXBAND, including complete input and run data on three test networks. Volume 2, the User's Manual, describes the MAXBAND system and how to use it in detail, including worked-out examples of a basic symmetric artery, a basic asymmetric artery, a general artery, and a loop. Volume 3, the Programmer's Manual, gives computer program documentation: First, an overall description of the program is provided, organized by subroutine, and then a listing is given for each subroutine along with a description of what it does. In addition, major variables used in the subroutine are listed. No attempt is made to document MPCODE, since excellent documentation has already been provided by Land and Powell (14).

## ACKNOWLEDGMENT

The development of the FORTRAN IV MAXBAND program was supported by a contract from FHWA. S.H. Cohen was contract manager. We also wish to acknowledge the assistance of Matthew Steele during several phases of the effort.

## REFERENCES

1. J.T. Morgan and J.D.C. Little. Synchronizing Traffic Signals for Maximal Bandwidth. Operations Research, Vol. 12, 1964, pp. 896-912.
2. J.D.C. Little. The Synchronization of Traffic Signals by Mixed-Integer Linear Programming. Operations Research, Vol. 14, 1966, pp. 568-594.
3. C.J. Messer, R.H. Whitson, C.L. Dudek, and E.J. Romano. A Variable-Sequence Multiphase Progression Optimization Program. HRB, Highway Research Record 445, 1974, pp. 24-33.
4. J.A. Hillier. Appendix to Glasgow's Experiment in Area Traffic Control. Traffic Engineering and Control, Vol. 7, 1966, pp. 569-571.
5. SIGOP: Traffic Signal Optimization Program--A Computer Program to Calculate Optimum Coordination in a Grid Network of Synchronized Traffic Signals. Traffic Research Corp., New York, Sept. 1966. NTIS: PB 17378.
6. D.I. Robertson. TRANSYT: A Traffic Network Study Tool. Road Research Laboratory, Crowthorne, Berkshire, England, Rept. LR 253, 1969.
7. N.H. Gartner, J.D.C. Little, and H. Gabbay. Optimization of Traffic Signal Settings by Mixed-Integer Linear Programming: Parts $I$ and II. Transportation Science, Vol. 9, Nov. 1975, pp. 321-363.
8. E.B. Lieberman and J.L. Woo. SIGOP II: A New Computer Program to Calculate Optimal Signal Timing Patterns. KLD Associates, Inc., Huntington Station, NY, Jan. 1976.
9. F.A. Wagner, D.L. Gerlough, and F.C. Barnes. Improved Criteria for Traffic Signal Systems on Urban Arterials. NCHRP, Rept. 73, 1969.
10. J.D.C. Little, B.V. Martin, and J.T. Morgan. Synchronizing Traffic Signals for Maximum Bandwidth. HRB, Highway Research Record 118, 1966, pp. 21-47.
1l. R.D. Desrosiers and C.H. Leighty. Traffic Flow Responses to Unannounced Increases in Progression Speeds of Signal Systems. Public Roads, Vol. 34, No. 1, April 1966, pp. l-15.
11. J.D.C. Little and M.D. Kelson. Optimal Signal Timing for Arterial Signal Systems: Volume l--Summary Report, Volume 2--User's Manual, Volume 3--Programmer's Manual. Federal Highway Administration, U.S. Department of Transportation, April 1980.
12. F.V. Webster. Traffic Signal Settings. Her Majesty's Stationery Office, London, Road Research Tech. Paper 39, 1958.
13. A. Land and S. Powell. FORTRAN Codes for Mathematical Programming. Wiley, London, 1973.
