REAL-TIME NETWORK DECOMPOSITION AND SUBNETWORK INTERFACING

R. J. Walinchus, TRW Systems Group, Houston

New methods of decomposing networks and interfacing the resulting subnetworks are being developed as part of the continuing research and development effort to improve real-time urban traffic control. The basic concepts and alternatives are introduced in this paper. Network decomposition may be required when the network geometry is not uniform and/or when the traffic characteristics within an area are not uniform. Decomposition is not successful unless two tasks are accomplished: (a) subnetwork determination according to certain criteria and (b) interfacing the various subnetworks. The two general ways in which subnetworks may be defined are fixed definition and real-time subnetwork definition. Fixed subnetworks are defined from geometrical and other considerations. Real-time subnetwork definition is based on traffic dynamics. To define successfully subnetworks in real time, two elements must be present: (a) a criterion by which it can be determined that an intersection is a "candidate" for consideration as part of a separate subnetwork and (b) a procedure by which these candidate intersections can be grouped into a workable subnetwork. The proposed criterion for candidacy is offset error—the difference between a realizable and an idealized offset. Among the grouping procedures are two genuinely real-time methods (rectangular subset and connectible set) and one pseudo real-time method. Interfacing subnetworks can be a difficult task. In the special case where the two areas have the same signal cycle, interfacing is accomplished via a "delta offset." This delta offset is a volume-weighted average of the offset changes desired across the interface. With unequal cycles, a matching and resynchronization technique might be employed, or interfacing can be accomplished through a transition zone where the signals are traffic-actuated rather than operated on any specific cycle.

•THE urban traffic control system (UTCS) of the 1970s will be burdened with increasing demands for more effective traffic control. The Federal Highway Administration (FHWA), as part of its effort to improve urban traffic conditions, is establishing the UTCS Laboratory in Washington, D. C. The UTCS Laboratory will employ a digital computer, other off-the-shelf hardware, and existing stored-pattern software; however, it will also serve as a tool for the development of advanced control techniques.

The FHWA requested proposals (1) for the development of second-generation traffic-control software that could be operational in the Washington Laboratory in 1972. The prospectus recognized the need for subnetwork configurations. This need was identified by the Traffic Systems Office of TRW Systems Group under an earlier FHWA contract (2) and has been pursued by the author through in-house studies. Developments in real-time network decomposition and subnetwork interfacing techniques are introduced in this paper.

Sponsored by Committee on Traffic Control Devices and presented at the 50th Annual Meeting.

Real-time network control via first-generation techniques requires choosing an appropriate stored pattern of signal settings; second-generation network optimization requires detailed computation of the signal settings based on traffic dynamics. An example of a network optimization program is SIGOP (3), but SIGOP is an off-line, steady-state program as opposed to a real-time one.

Most network optimization programs contain an iteration routine or a matrix inversion, or both (e.g., subprogram OPTIMIZ in SIGOP). The computation time required for these operations is nonlinear with respect to the number of variables involved; i.e., doubling the number of variables will more than double the computational time required (quite often, the time increase is nearly exponential). Partitioning the network is a major benefit because solving several small problems will, in general, take less time than solving one large problem. Thus, partitioning of a network into subnetworks (fixed and/or real-time definition) makes optimum use of available time on the digital computer, which is used to periodically compute optimum signal settings.

The analytical basis of network optimization via subnetworks can be likened to manipulating a very large matrix to solve a big problem: It is easier and faster to partition the matrix (network) into submatrices (subnetworks). This partitioning may be required when the network geometry and/or the traffic characteristics are not uniform or when the optimization problem is too large to solve in real time. In addition, partitioning must be judicious; network decomposition and subnetwork interfacing must be done with care so that the optimization of individual subnetworks yield satisfactory results when integrated.

The decomposition of a general urban network into subnetworks will become an integral part of real-time urban traffic control. Network decomposition is not successful unless the following two tasks can be accomplished:

1. Subnetwork determination according to certain criteria, and
2. Interfacing the various subnetworks.

These two tasks are discussed in the following sections.

SUBNETWORK DETERMINATION

Two general ways in which a network may be decomposed into subnetworks are (a) fixed subnetwork definition and (b) real-time subnetwork definition. Fixed subnetworks arise from geometrical and other considerations. Real-time subnetwork definition is based on traffic dynamics.

For a general urban area, a combination of the two methods may be necessary because of the limited instrumentation that may be available and because of certain peculiarities in geometry that may exist in the area.

Fixed Subnetworks

There are a number of criteria that are used to establish fixed subnetworks. Typical decomposition criteria are (a) freeway access/service road; (b) major arterial into urban grid; (c) area of closely spaced intersections; (d) signal-independent flow area; (e) geometrical/political subdivisions; and (f) established traffic patterns. A few comments about each of these will illustrate their applicability for any given urban area.

Where urban streets empty onto freeways or merge into freeway service roads, a subnetwork boundary usually can be defined. Where a major arterial feeds into an urban grid, the arterial and the grid may be considered as different subnetworks with appropriate boundary conditions.

Certain areas are a maze of closely spaced signals. Often, these "dense" areas are surrounded by relatively long blocks with reasonably laminar flow. Decomposition around the middle of the long surrounding blocks is possible.

Certain areas and streets have sufficient feeders and turning movements so that traffic flow is rather consistent and virtually independent of the settings of surrounding signals. (In some cases, this condition is brought about by lack of instrumentation and signal control.) This is an ideal place to perform network decomposition. This criterion is usually related to the following one.

In many cities a river, creek, park, or institution provides for easy decomposition. In other areas, the political subdivisions between communities or counties may introduce mandatory subnetwork boundaries.

Finally, there may be areas where the pattern of behavior of traffic is "established" in some sense. The city traffic engineer may decide on subnetworks because "area A always requires a signal cycle of about X seconds and area B always needs one of about Y seconds." This is one of the less desirable criteria for subnetwork definition because it assumes constancy and will limit flexibility to optimize flow if conditions change.

Regardless of which criteria are used, the result is the same—fixed subnetworks. Some degree of flexibility can be obtained by storing in the computer several network "decomposition maps," any one of which can be called in by an evaluation of traffic conditions, by keyboard entry, by time of day, etc. This is discussed in more detail under the heading "Pseudo Real-Time Definition."

A more meaningful criterion for the definition of subnetworks is traffic dynamics. This implies real-time subnetwork definition, as will be discussed.

Real-Time Subnetwork Definition

In order to define successfully subnetworks in real-time, two elements must be present:

1. A criterion by which it can be determined that an intersection is a candidate for consideration as part of a separate subnetwork, and
2. A procedure by which these candidate intersections can be grouped into a workable subnetwork.

Criteria for Candidacy—Real-time subnetwork definition based on traffic conditions requires establishment of criteria by which an intersection can be judged to be a candidate for separation from the network proper.

One criterion for subnetwork candidacy is "offset error" at the intersection. This parameter is defined as the optimum offset when the intersection is considered as part of the whole network minus the idealized offset of the intersection with respect to adjacent intersections. (The "idealized offset" as used in SIGOP is a constant; in real-time urban traffic control it is dynamic and is based on speeds, queues, etc.) When the offset error is too large (compared to a constant or dynamic threshold value), this implies that the network solution is constraining the intersection to be inefficient on a local basis. This intersection is then a candidate for consideration as part of a separate subnetwork.

A variation of the preceding criterion is "weighted offset error," where the offset error on each approach to the intersection is weighted by the pertinent volume (rather, vehicles requiring service). In this way, an intersection with few vehicles requiring service need not be overemphasized because the offset error may be counterbalanced by spare green time in which the improperly phased traffic can be processed.

Other criteria that might be used to determine candidacy are intersection saturation level, traffic density, etc.

The criterion chosen depends to some extent on the amount of instrumentation and the accuracy of data for a given intersection. It is difficult to determine the candidacy of an intersection for which traffic dynamics data do not exist.

In the discussion that follows, assume that an appropriate criterion or combination of criteria has been chosen; the candidate intersections must be grouped into a workable subnetwork using one of the methods discussed.

Figure 1 will be useful in the following discussions. It shows a pattern of candidate (dark dot) intersections and a street-intersection numbering technique, namely via i and j. The north-south streets are specified by a specific value of i, and the east-west streets by j. An intersection is specified by an i-j pair, (i,j). Among all the intersections in the network, the candidate intersections form a set $\{(i^*,j^*)\}$. In Figure 1, this set of candidate intersections contains

$$\left\{(i^*,j^*)\right\} = \left\{\begin{matrix}(3,3), & (3,4), & (4,3), & (4,4), & (4,5)\\ (5,2), & (5,4), & (5,5), & (6,3), & (6,4)\end{matrix}\right\}$$

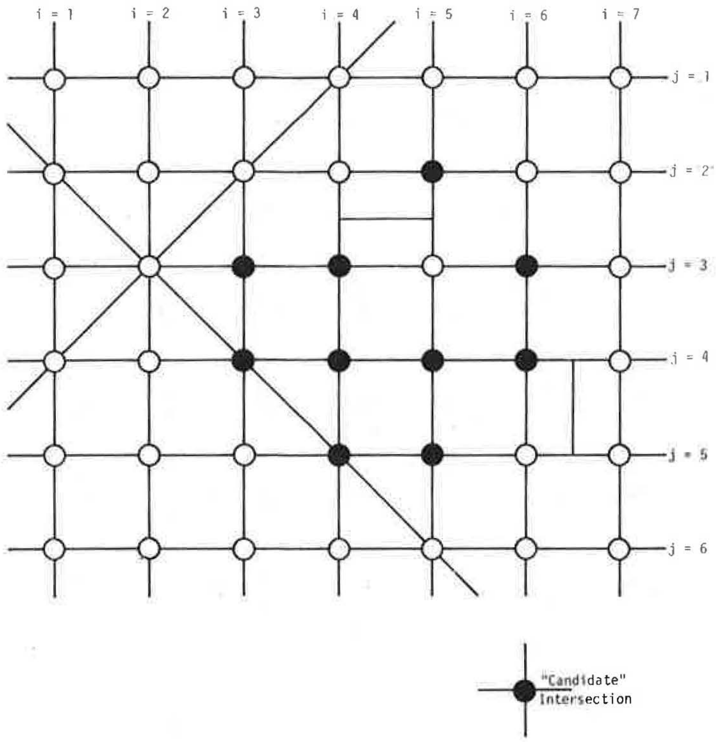

Figure 1. Intersections that are candidates for grouping into a subnetwork.

This street-intersection numbering technique and the set of candidate intersections are particularly useful in the "rectangular subset" method of real-time subnetwork definition.

Grouping Methods—Experimentation with three methods of grouping candidate intersections into workable subnetworks has been pursued. The methods are pseudo real-time, real-time (rectangular subset), and real-time (connectible set). The first two are being programmed and checked out.

Pseudo real-time subnetwork definition makes use of stored decomposition maps; the map that best fits current conditions is used. The two real-time methods, rectangular subset and connectible set, do what their names imply. In the first, a rectangular area containing all (or the highest density) of the candidate intersections is used to define a subnetwork. The second merely searches for the largest grouping of candidate intersections that are connected by optimizable links. More details on all three methods are given in the following.

Pseudo Real-Time Definition—This method uses one of the several stored "decomposition maps" that tell how the network should be partitioned into subnetworks. One version employs an index cross-referencing procedure to sort traffic data according to subnetworks. If a library of these maps is available, it becomes a matter of picking the best map to match current traffic conditions. If the stored maps handle the frequently encountered situations, matching is relatively easy. For example, consider Figure 2.

In the figure, the area within the dashed circle has given trouble in the past, and therefore the corresponding decomposition map has been stored in the library. The current traffic conditions show a high density of candidate (dark dot) intersections in this area and none outside. Therefore, the indicated decomposition map is the one best matched to current conditions, and a subnetwork is defined by the intersections within the dashed circle.

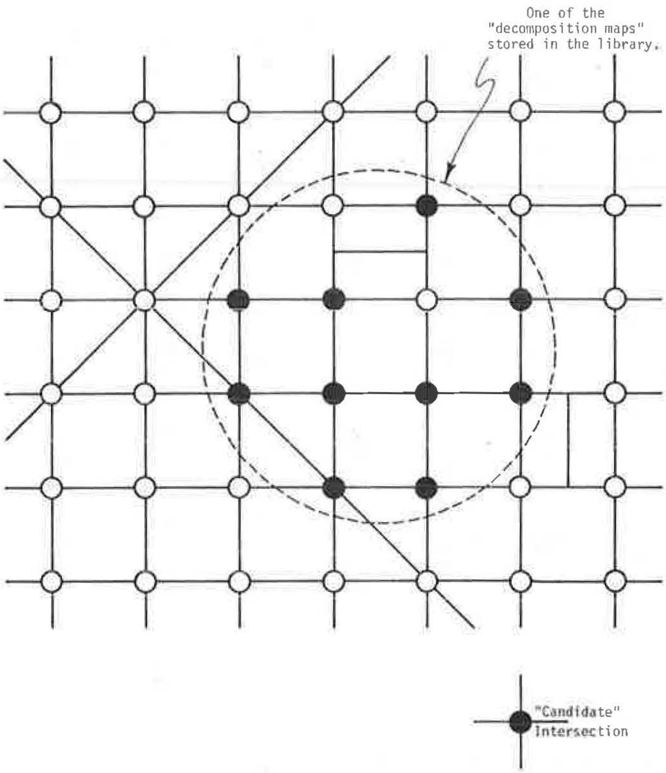

Figure 2. Pseudo real-time subnetwork definition: decomposition map matching.

The name pseudo real-time has been attached to this method because of the possibility of poor map matching (due to variable traffic conditions and/or an incomplete map library). The real-time methods define their own subnetwork boundaries.

Real-Time Definition: Rectangular Subset—This method searches for a rectangular area that includes all (or the highest density) of the candidate intersections.

By using the street-intersection numbering technique shown in Figure 1, it is relatively easy to determine the rectangular area enclosing all of the candidate intersections. All that need be done is to find the maximum and minimum i* and j* in the set $\{(i^*,j^*)\}$. Define the following quantities:

$$I_1 = \max_{\{(i^*,j^*)\}} i^* \quad , \quad I_2 = \min_{\{(i^*,j^*)\}} i^*$$

$$J_1 = \max_{\{(i^*,j^*)\}} j^* \quad , \quad J_2 = \min_{\{(i^*,j^*)\}} j^*$$

In Figure 1, these quantities have the values

$$I_1 = 6 \quad , \quad I_2 = 3$$

$$J_1 = 5 \quad , \quad J_2 = 2$$

The corresponding rectangular subnetwork that includes all of the candidate intersections is within the dashed area shown in Figure 3. This technique has been programmed.

A variation of the preceding technique is to find a smaller rectangular area (within the one above) that contains a higher density of candidate intersections. This is ac-

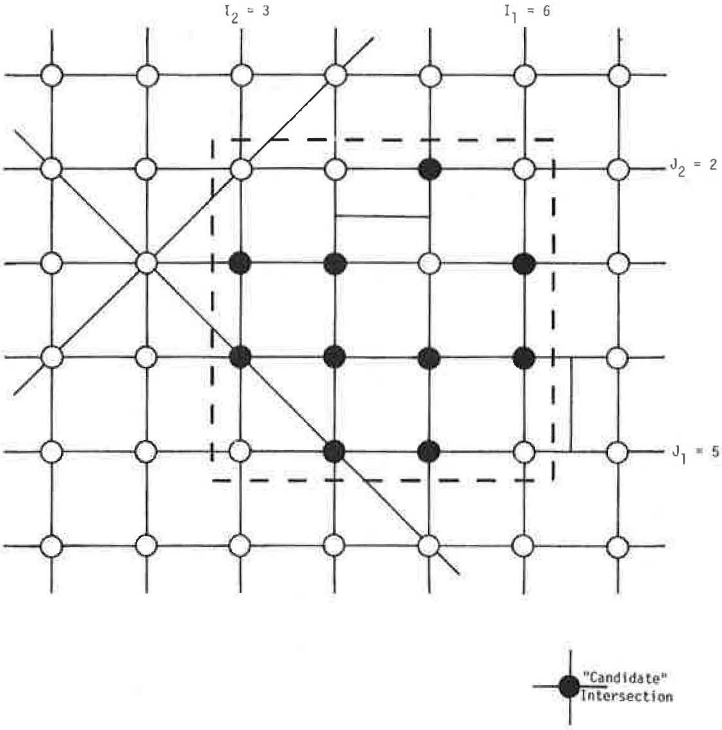

Figure 3. Subnetwork definition via rectangular subset containing all candidates.

complished simply (from a programming standpoint) by eliminating one street at a time from the sides of the full rectangular area; i.e., test

$$I_1' = I_1 - 1 \text{ with } I_2, J_1, J_2 \text{ fixed}$$
$$I_2' = I_2 + 1 \text{ with } I_1, J_1, J_2 \text{ fixed}$$
$$J_1' = J_1 - 1 \text{ with } I_1, I_2, J_2 \text{ fixed}$$
$$J_2' = J_2 + 1 \text{ with } I_1, I_2, J_1 \text{ fixed}$$

and eliminate the one street that leaves the highest density of candidate intersections in the remaining rectangle. (The procedure can be applied, if necessary, more than once, provided the resulting rectangle does not yield a trivial or undesirable case.) Applying the procedure once to the situation in Figure 3 gives the situation shown in Figure 4 where

$$I_1 = 6 \quad , \quad I_2 = 3$$
$$J_1 = 5 \quad , \quad J_2' = 3$$

The "density" of candidate intersections has been increased from 0.625 (in Figure 3) to 0.75 (in Figure 4) while dropping only one candidate intersection. (That one intersection will be no worse off, but the three adjacent noncandidates will benefit by remaining part of the other area.) The corresponding smaller rectangular subnetwork is within the dashed area shown in Figure 4.

Real-Time Definition: Connectible Set—This method is an extension of the higher density concept; it is the most logical from a grouping standpoint, but not necessarily the best when considering the overall optimization problem.

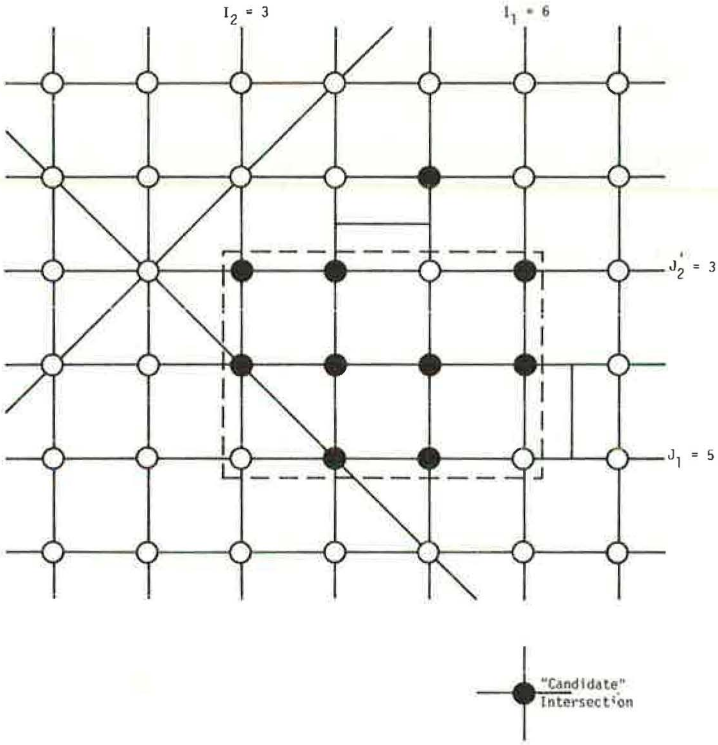

Figure 4. Rectangular subset containing higher density of candidates.

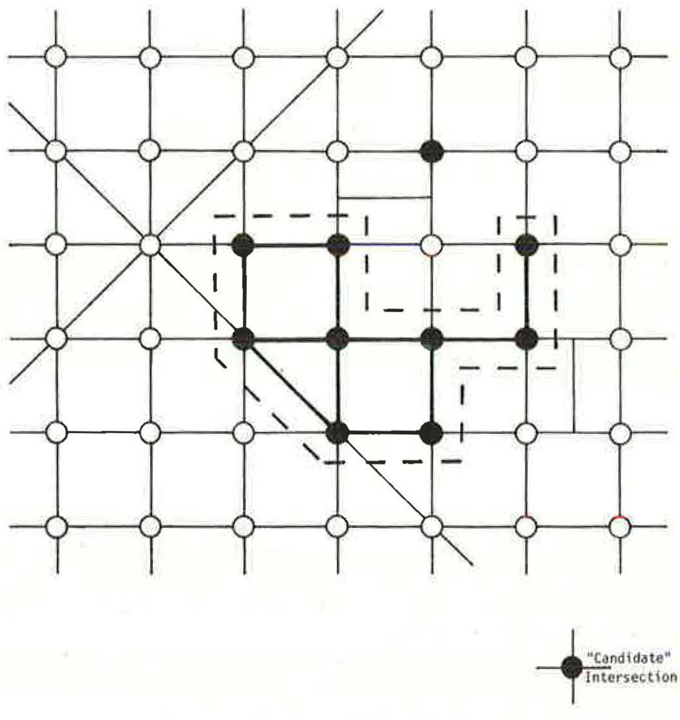

Figure 5. Largest connectible set of candidates.

It requires sorting through the list of candidate intersections and associated links to find the largest connectible set. (In this context, a pair of candidate intersections is "connectible" if there is an optimizable link and no other optimizable intersection between the pair.) Figure 5 shows the results of the method where the pertinent connecting/optimizable links are shown as dark lines. Note that one candidate intersection is not connected to the rest of the group.

The method is largely a sorting problem. The efficiency of the bookkeeping operations may depend a great deal on the numbering scheme for identifying intersections and links.

A drawback of the method is the irregular subnetwork boundary that may result (see dashed line in Figure 5). This will affect the efficiency of merging traffic smoothly at the boundary interface.

INTERFACING SUBNETWORKS

The preceding section dealt with the problem of defining subnetworks. Once the subnetworks are defined, traffic data can be sorted according to subnetwork, and each subnetwork can be optimized. Then comes the problem of interfacing subnetworks with one another and the rest of the network.

If a subnetwork and surrounding area have the same optimum signal cycle, the interfacing task is much simpler. With different cycles there are two approaches that can be taken for interfacing:

1. Initially match the offsets of the two areas for good traffic flow between them; then, as the two signal cycles move too far out of synchronism, disrupt the subnetwork cycle to force resynchronization; and
2. Introduce a "transition zone," at least one intersection wide between the two areas, letting the transition intersections operate almost on a traffic-actuated basis.

Interfacing Under Equal Cycles

When two areas have the same signal cycle, interfacing across their common boundary is relatively simple. It is a matter of introducing a "delta offset" to all the offsets within the subnetwork (or to the subnetwork master signal) to put its main-street green-on times in synchronization with the surrounding area.

The delta offset is computed via the equation

$$\text{Delta offset} = \frac{\displaystyle\sum_{\substack{\text{interfacing}\\ \text{links}}} (\text{Volume})\,(\text{Offset change desired})}{\displaystyle\sum_{\substack{\text{interfacing}\\ \text{links}}} (\text{Volume})}$$

where

Offset change desired = Offset of upstream intersection + Idealized offset between intersections based on free-flow travel time - Offset of downstream intersection, modulo the signal cycle

for each interfacing/boundary link. Thus, the delta offset is simply the volume-weighted average of the offset changes desired across the interface.

Encountering two adjacent areas with the same (or nearly the same) signal cycle introduces some interesting questions:

1. If the two areas have the same signal cycle, should they be combined into and optimized as one area?
2. If the two signal cycles are "nearly" the same, should (a) the two areas be combined into one, or (b) the two areas be kept separate but constrain the two signal cycles to be the same?

The traffic engineer will have to answer these questions before development of the subnetwork definition and interfacing program is completed.

For areas with different signal cycles, one of the two following approaches can be employed.

Interfacing by Matching/Resynchronization

In the matching/resynchronization technique, an initial matching of traffic flow is accomplished using the delta offset previously discussed. However, because the two signal cycles are different, the signal settings move out of phase so that traffic flow between the two areas is not synchronized. It is desirable to resynchronize the settings periodically if it is not too disruptive to traffic flow.

Periodic resynchronization could be performed as new optima are computed. A routine can be developed whereby subnetwork signal phases are appropriately extended or shortened to resynchronize the settings.

This technique might be required in cases where the subnetwork boundary/interface must be so abrupt as to prohibit introduction of a transition zone between areas.

Interfacing Through a Transition Zone

The other interfacing technique calls for a transition zone (one or two intersections deep) to be established between the subnetwork and the surrounding area. The signals in this zone do not operate on any specific cycle.

The phases of the transition-zone intersections are set on the basis of traffic demand. The demand can be determined by actuation (if instrumentation exists) or prediction (of platoons leaving the subnetwork or surrounding area and proceeding toward the transition zone). The prediction method of determining demand probably has to be used because of the limited amount of instrumentation that may be available in a given urban area, and because the instrumentation is not likely to be in any desired transition zone.

CONCLUSIONS

This paper has presented the basic concepts and alternatives for the real-time decomposition of networks into appropriate subnetworks and subsequent interfacing of the subnetworks. Some of these developments will be applied in the Washington laboratory beginning in 1972; others may be in use in cities by the mid-1970s.

ACKNOWLEDGMENTS

The author is principal investigator under a research and development contract between the Federal Highway Administration and the TRW Systems Group. The opinions, findings, and conclusions expressed are those of the author and not necessarily those of the Federal Highway Administration.

REFERENCES

1. Traffic Adaptive Network Signal Timing Program. U. S. Dept. of Transportation, Federal Highway Administration, RFP-121 (24-82), March 1970.
2. Final Report: System Analysis Methodology in Urban Traffic Control Systems. TRW Report No. 11644-HO14-RO-00, 30 June 1969. Available from National Technical Information Service, Springfield, Virginia, as PB 185 422.
3. SIGOP: Traffic Signal Optimization Program. Traffic Research Corp., New York, September 1966. Available from National Technical Information Center as PB 173 738.