

**Simulation-Based Performance Studies of
Utilizing Network Programming Algorithm
to a Network-Wide Traffic Control**

UCI-ITS-TS-WP-03-15

Wann-Ming Wey¹
R. Jayakrishnan²

¹ Graduate School of Architecture and Urban Design
Chaoyang University of Technology; Taiwan, R.O.C. 413
wmwey@mail.cyut.edu.tw

² Department of Civil and Environmental Engineering and
Institute of Transportation Studies; University of California, Irvine
rjayakri@uci.edu

November 2003

Institute of Transportation Studies
University of California, Irvine
Irvine, CA 92697-3600, U.S.A.
<http://www.its.uci.edu>

Paper No.: 04-3222

**Simulation-Based Performance Studies of Utilizing Network Programming
Algorithm to A Network-Wide Traffic Control**
(revised version)

(Submitted to *TRB Committee A1C05 - TRANSPORTATION NETWORK MODELING*
for 2004 TRB Annual Meeting)

Wann-Ming Wey

Associate Professor

Graduate School of Architecture and Urban Design &

Department of Architecture

Chaoyang University of Technology

168 Gifeng E. Rd., Wufeng

Taichung County, Taiwan, R.O.C. 413

886-4-23323000 ext. 7153

Fax 886-4-23742339

E-mail: wmwey@mail.cyut.edu.tw

R. Jayakrishnan

Associate Professor

Department of Civil and Environmental Engineering

Institute of Transportation Studies

University of California at Irvine

Irvine, CA 92697, USA

(949) 824-2172

Fax (949) 824-8385

email: rjayakri@uci.edu

November 2003

*Submitted for Presentation at the 83rd Annual Meeting of the Transportation Research Board,
Washington D.C., Jan 2004 and for Publication in the Transportation Research Record*

Simulation-Based Performance Studies of Utilizing Network Programming Algorithm to A Network-Wide Traffic Control

Wann-Ming Wey*

Associate Professor

Graduate School of Architecture and Urban Design,
Chaoyang University of Technology,
168 Gifeng E. Rd., Wufeng,
Taichung County, Taiwan, R.O.C.

R. Jayakrishnan

Associate Professor

Department of Civil and Environmental Engineering
Institute of Transportation Studies
University of California at Irvine
Irvine, CA 92697, USA

ABSTRACT

In this paper, we present a complete optimal formulation of a network traffic control scheme with embedded traffic flow models (platoon dispersion) in the form of arc-flows in a time-expanded network. The integer-linear network-programming formulation is solved using a modified network simplex and branch and bound scheme. The results of comparing the solutions to other actuated controls are discussed here. The platoon dispersion model used is the well-known Robertson's model, which forms linear constraints. Thus it is a rare example of a traffic simulation being analytically embedded in an optimization formulation. The formulation is an integer-linear program, and does not assume fixed cycle lengths or phase sequences. It assumes full information on external inputs, but can be incorporated in a sensor-based environment, as well as in a feedback control framework. The integer-linear program formulation may not be efficiently solved with standard simplex and branch and bound techniques. We discuss network programming formulations to handle the linear platoon dispersion equations and the integer constraints at the intersections. A special purpose network simplex algorithm for fast solution is also mentioned.

The optimization model takes the form of mixed integer linear programming. The control strategies generated by these optimization models were compared with those derived from conventional signal timing models, using the TRAF-NETSIM microscopic simulation model. It was found that the optimization models successfully produced optimal signal timing plans for the various signalized intersections including simulated and real-world networks. The proposed optimization models consistently outperformed the conventional signal control methods with respect to system delay objective. This conclusion was drawn from the TRAF-NETSIM simulation.

Key Words: Network Programming Formulation, Network Simplex Algorithm, Microscopic Simulation, Rolling Horizon, Traffic Control

* Corresponding author; Tel.: 886-4-23323000 ext. 7153; fax: 886-4-23742339.
E-mail address: wmwey@mail.cyut.edu.tw (Wann-Ming Wey).

1. INTRODUCTION

Conventional signal control strategies based on methods developed during the 1970s and 1980s are not considered uniformly effective under all possible conditions (Tarnoff and Gartner, 1993). Many strategies fail to provide improvements over well-timed first generation systems for congested conditions, and some of these systems exhibit degraded performance during specific sets of under saturated conditions (Boillot et al., 1992). For example, current systems are relatively slow to respond to sudden changes in traffic flow caused by incidents or large fluctuations in demand. Such systems have been designed to implement small changes over time to overcome the problem of frequent transitioning (see Michalopoulos (1992), for an excellent discussion of the deficiencies in the current practice). Real-time stochastic control based on detected traffic is an option which has not been applied in an integrated fashion at the network level due to the lack of complete analytical network-wide optimization formulations. In an earlier paper (Wey and Jayakrishnan, (1997)) we presented such a network formulation, which also includes analytically embedded models for traffic flow between intersections so that such a network-wide optimization can be attempted. The formulation, then presented only with results on the computational efficiency of the network simplex application, was received well, which encouraged us to present results on its application to a real-network traffic control context, which is in this paper. The performance evaluation reported here are based on microscopic simulations, since complete real-world hardware implementation is only planned for the future.

An extensive review of traffic signal control methods can be briefly stated as follows. For real-time control, several algorithms have been proposed. For example, Miller (1965) considered an intersection with heavy traffic and assumed that at time t the signal is green on primary approach. At this time the controller can make a binary decision, i.e., to change the signals immediately, or after an extension of one unit of time. However, Miller did not consider the intersection of adjacent intersections, and thus did not include the downstream delays in determining an optimal extension strategy. Ross et al. (1970), basing their work on a philosophy similar to that of Miller, developed a computer control scheme for traffic responsive control of a critical intersection that not only minimizes the total delay of all users of the intersection, but also minimizes the total delay accumulated at down stream intersections. Moreover, Longley (1968) proposed a control scheme for a two-phase congested intersection employing a "queue balancing" strategy. This strategy seeks to hold a particular linear function of the intersection queues to a value of zero by adjustment of the green time split. Lee et al. (1975) also considered queues rather than delays as the objective of the control and developed another semi-empirical strategy called "Queue Actuated Signal Control". This is a control policy where an approach receives green automatically when the queue on that approach becomes equal to or greater than some predetermined length, regardless of the conditions on the conflicting approaches. The policy assumes that no two conflicting approaches reach the upper bound specified for them simultaneously. Another approach to critical intersection control has been suggested by Gordon (1969). Gordon did not attempt to minimize delay at the intersection but rather to maintain a constant ratio of the queue lengths on opposing approaches. The cycle length is assumed constant and the splits are changed according to the demand so that the ratio of the actual queues to the maximum link storage space on both phases are equal. Finally, Michalopoulos (1977, 1978) proposed an optimal control policy for both pretimed and real time control. His control policy was to minimize total system delay subject to queue length constraints. However, it should be noted that most of the control algorithms mentioned above suffer from complex computational requirements described as above, and from the lack of

intersection-to-intersection traffic flow models.

Besides, the review examines Fixed-time Control schemes such as the original Urban Traffic Control System (UTCS), Optimization techniques such as MAXBAND (Little et al., 1981) and TRANSYT (Robertson, 1969); Traffic-responsive control without optimization, such as the common actuated control; Traffic responsive control with optimization such as SCATS (Lowrie, 1982, 1990; Luk, 1984; Sims, 1979), SCOOT (Hunt et al., 1981; Robertson and Bretherton, 1991), UTOPIA (Donati, et al., 1984), PROLYN (Henry et al., 1983; Henry and Farges, 1989), OPAC (Gartner et al., 1992) and ongoing research on methods such as RHODES (University of Arizona (Larry Head et al., 1997)) under the recent FHWA research initiatives.

The primary difference in the formulation in our research from other existing models is the use of a multiple time-period network programming scheme to handle traffic flow between intersections using the linear nature of platoon dispersion models, and to specify the integrated network level control of multiple intersections in one single framework. The special purpose solution algorithm yields orders of magnitude better solution efficiency, as well as applicability to orders of magnitude of larger problems, compared to standard linear programming. Our research is perhaps the first attempt to apply this to the signal control problem. Perhaps the reason why such a technique has not been used for this problem is that the resulting network program is in a specific non-standard form which is considered to be in a difficult class of network optimization problems. We, however, develop the techniques to handle the non-standard nature without affecting the solution efficiency.

In this paper, we provide results on the application of our formulation to a subnetwork in Irvine, California. Due to the non-availability of standard benchmark solutions showing the "best-possible" optimization results in a real network, we simulate this network under a very large number of control conditions to identify possibly the "best" actuated control scheme for the network, and then compare the results of such control to the results from our control scheme. The comparison is made under the assumption of full knowledge of the traffic demands in the network over the time period. Thus the performance comparison against actuated control are made under the assumption of perfect detection and prediction in both cases. Preliminary work showing a rolling horizon application of the method is also given in this paper; however, the results from real-world applications should be expected to be different, and will be available only in the future.

2. OPTIMIZATION MODEL FORMULATION

We describe our formulation of the network-wide signal optimization problem, and provide the solution algorithm in section 3. The formulation has several concepts borrowed from the above algorithms, and does not consider some constraints included in some of the above algorithms. We mention here that the network programming solution algorithm provides a very intuitive way to incorporate the constraints of the formulation and to add or delete additional constraints that could be considered. Briefly stated, the reason is that the solution uses network paths, and even the most complicated constraints of the flow and signal problem at an intersection become intuitive and simpler to handle using the paths on the time-expanded network. This will become clearer later, but it is useful to remember while reading the formulation given next.

2.1 Introduction

In this research, the traffic flow and dynamic reduction of queues are described by an appropriate linear model with linear capacity constraints for both road links and flows in intersections. It is also assumed that origin-destination relationships are known a-priori. As far as the network traffic flow is concerned, the network control is a superposition of control at individual junctions. Because the control of neighboring intersections determines the arrival process, the entire system is strongly interactive. The optimization of traffic control thus becomes a complex multi-commodity flow problem.

The optimization model proposed here is a dynamic model with multiple-time steps, as it is meant for real-time operations. The dynamic model is formulated not only to obtain minimum delay subject to queue length constraints, but also to include formulas to define relationships of queue transition between time slices. In the model, the green/red lights can change with every time slice. The signal intervals are adjusted toward minimizing delay and permitting queues to build up to a predetermined upper bound.

The optimization model uses mixed integer linear programming (MILP) to mathematically model the above requirements. Some MILP formulations of signal control already exist (Kim, 1990; Messer et al., 1986; Tsay and Lin, 1988), though not for complete networks. An MILP problem could be solved using software packages available for mathematical programming such as GAMS (Brooke et al., 1988) and LINDO (Schrage, 1984), which normally use the Simplex algorithm and branch and bound solution techniques. We however develop much faster algorithms as given in section 3, for use within a standard branch and bound solution scheme.

2.2 Road Traffic Models

Before introducing the model formulation of network traffic signal control, the concept of dynamic road traffic models adapted in this problem formulation will be presented first as follows, followed by some assumptions made for the problem statement.

A well-known platoon dispersion model (Hunt et al., 1982) is used to provide dynamic interaction constraints among individual intersections which have their own set of signalization constraints. This model has been empirically validated in several urban areas around the world (since it is part of the SCOOT control system implemented around the world). Though it is only an empirical model, it is generally considered to represent (interrupted) traffic flow in signalized networks better than other models. The model is presented here to point to its linear structure, which makes it particularly useful in solvable network optimal control formulations.

The dynamic evolution of traffic flow on the j -th road section (approach) can be modeled by the following set of equations:

$$q_{out}^j(k)\Delta T = \min \begin{cases} q_{go}^j(k)\Delta T \\ q_s^j(k)\Delta T + l_j(k-1) \end{cases} \quad (2.1)$$

$$l_j(k) = l_j(k-1) + (q_s^j(k) - q_{out}^j(k))\Delta T \quad (2.2)$$

$$q_{in}^j(k) = \sum_i b_{ij} q_{out}^i(k) \quad (2.3)$$

A discrete time approach with sample time ΔT (the ΔT is selected adaptively, that means, the length of ΔT can be set optionally depending on how adaptive the control system needs.

Here the ΔT is chosen to be 5 seconds) is adopted. The following variables apply for each road section j ($j = 1, \dots, J_s$): $q_{in}^j(k)$ is the volume (veh/sec) entering the j -th road section during $k\Delta T \leq t \leq (k+1)\Delta T$, which depends on the turning fractions b_{ij} from other links, $q_s^j(k)$ is the volume which arrives at the end of the waiting queue or at the stop-line, $q_{out}^j(k)$ is the leaving volume at the downstream end of the j -th road section, $q_{go}^j(k)$ is the capacity flow during green traffic signal and $l_j(k-1)$ is the number of vehicles waiting in the queue. q_s^j and q_{in}^j are connected through the platoon dispersion equations (2.4) stated as follows.

$$q_s^j(k + \tau) = Fq_{in}^j(k) + (1 - F)q_s^j(k + \tau - 1) \quad (2.4)$$

where $F = 1/(1 + \alpha\tau)$, it is a smoothing factor determined by a specific platoon dispersion factor α and the travel time coefficient between the upstream and downstream of an intersection τ ($F = 1$ means no dispersion). And α is a constant parameter (normally around 0.5) which may have a value ranging from 0.2 to 0.5, depending upon whether there is very little or extensive platoon dispersion along the road link. The other parameter τ which is equal to $0.8 \times$ (average travel time in ΔT units). However, one thing needs to be noted is that many studies have focused on the analysis and the calibration of the F or α factor. Most of these investigations have suggested that platoon dispersion should not be generalized with the standard default parameter settings that are suggested for this model, but that instead these parameters should each time be customized to match the unique road condition on each link.

Basic Assumptions of Problem Statement

In order to control traffic during a limited period of interest efficiently, detailed information about the demand structure and the factors determining intersection capacities, which often are the bottlenecks causing the congestion, can be useful. The traffic control model assumes that the traffic assignment is known a-priori, which means that the demand can be specified for individual routes. Rather than using the traditional concept of turning fractions at intersections, we use actual volumes for each movement at the intersection. Note that it is easy to find these by multiplying approach arrivals by turning fractions as well. An additional reason is our path-based approach. Recently proposed path-based assignment algorithms (Jayakrishnan et al., 1994; Sun et al., 1996) have been found to perform much faster than the link-flow based assignment algorithms, which points to the attractiveness of our approach if path-based static or dynamic assignment is used in real time for prediction of path-flows in an ATMS.

Roads entering the network have to be included in the model, because their queue lengths are determined by the signals at the intersections. On the other hand, it is assumed that vehicles can leave the networks as soon as they pass the last intersection on their routes. The entering links (or external approaches) are numbered $1, \dots, E$ while interior links (or internal movement) receive indices $E + 1, \dots, H$. Each exit gets the same index as the associated entrance.

We also assume that:

1. The lost time (yellow and all red) of intersection N is given,
2. The saturation flows $S_i, i \in I_n$, are known,
3. The turning movement fractions are known and may be time variant.

With regard to the turning movement fractions (last assumption above), it should be

noted that they may be estimated in real time by known algorithms (e.g. Cremer, 1991).

2.3 Model Formulation of Network Traffic Signal Control

First, we note that the entire formulation is based on phases (movements) at each intersection. In our presentation, each phase refers to a movement consisting of two links, a from_link and a to_link. At this point we can formally summarize the presentation of the model.

Formulation

The following notations are used in the model formulation:

- N = number of intersections,
- n = index used to refer to one of the N intersections,
- I_n = set of signal phases at the control intersection n ; $n \in N$,
- O_i = set of signal phases at the start node of the from_link of a given phase i that feeds traffic for phase i ,
- i = signal phase at a given node n , $i \in I_n$,
- L_i = from_link of phase i
- T = the time horizon under consideration specified in seconds,
- K = number of discrete ΔT time intervals in the optimization period,
- k = time interval index,
- ΔT = the sample time interval of duration, sec,
- H = number of links in the network, including entrances,
- E = number of entrances and exits,
- $q_{in}^i(k)$ = upstream inflow for phase i over a period $[k\Delta T, (k+1)\Delta T]$, veh/sec,
- $q_s^i(k)$ = the flow which arrives at the end of the waiting queue or at the stop-line, veh/sec,
- $q_{go}^i(k)$ = the capacity flow for green traffic signal, veh/sec,
- S_{ni}^g = saturation flow for green time of phase i at intersection n , veh/sec,
- S_{ni}^y = saturation flow for yellow time of phase i at intersection n , veh/sec,
- $u_{ni}(k)$ = 0 if signal state is green for phase i at intersection n and time step k , and
1 if signal state is red for phase i at intersection n and time step k ,
- $\xi_{ni}(k)$ = 1 if existing signal state of phase i at intersection n is switched at end of time
step k , and
0 otherwise,
- G_{min}^i = minimum green for phase i (seconds)
- G_{max}^i = maximum green for phase i (seconds)
- $U_{ni}(k)$ = green time used by phase i at intersection n , at the end of time step k (seconds)
- $q_{out}^i(k)$ = outflow of phase i at the downstream end over period $[k\Delta T, (k+1)\Delta T]$ veh/sec,
- $l_{ni}(k)$ = number of vehicles of phase i queued up at the end of time interval k at intersection n .
- $d_i(k)$ = side entry flow during phase i on the corresponding approach link, in veh/sec,
- $s_i(k)$ = side exit flow during phase i on the corresponding approach link, in veh/sec,
- β_{i0} = exit rates within phase i ,
- λ_{ip} = fraction of queued vehicles of movement i that uses buffer p ,

- p = the queue buffer number, $p \in B_n$,
 B_n = set of separate queue buffers of node n ,
 Qp = set of phases that share the buffer p ,
 Cp = the storage capacity of the queue buffer p ,
 τ = a travel time coefficient between the upstream and downstream of an intersection,
 F = a dispersion parameter,
 TD = total delay, veh-intervals,
 Z_i = 1 when phase i is oversaturated, and
 0 when phase i is undersaturated,
 M = very large positive value, called Big-M.

We define $q_{in}^i(k)$ and $q_{out}^i(k)$ to be the inflow and outflow respectively of phase i over a period $[k\Delta T, (k+1)\Delta T]$ where ΔT is the sample time interval and $k = 1, 2, \dots$ is a discrete time index. Similarly we define $d_i(k)$ and $s_i(k)$ to be the demand flow and the exit flow, respectively, occurring within phase i .

The control objective of the dynamic model is to minimize total delay in the network. Total delay is the sum of delays on all phases. The full formulation of the mathematical program for minimization of delay is below, and the description of the constraint set follows:

Minimize

$$\text{Delay } TD = \Delta T \sum_{n=1}^N \sum_{i \in I_n} \sum_{k=1}^K l_{ni}(k) \quad (2.5)$$

subject to

$$l_{ni}(k) \geq 0 \quad ; \forall i \in I_n, n \in N, k \in K \quad (2.6)$$

$$l_{ni}(k) \geq l_{ni}(k-1) + (q_s^i(k) - q_{out}^i(k))\Delta T \quad ; \forall i \in I_n, n \in N, k \in K, Li \leq E \quad (2.7)$$

$$l_{ni}(k) \leq MZ_{ni}(k); \forall i \in I_n, n \in N, k \in K, \text{ where } M \text{ is some large number} \quad (2.8)$$

$$l_{ni}(k) - \{l_{ni}(k-1) + (q_s^i(k) - q_{out}^i(k))\Delta T\} \leq M(1 - Z_{ni}(k)) \quad ; \forall i \in I_n, n \in N, k \in K, Li \leq E \quad (2.9)$$

$$l_{ni}(k) \geq l_{ni}(k-1) + \Delta T \{ [F \cdot (1 - \beta_{i0}) \cdot \sum_{j \in O_i} q_{out}^j(k - \tau) + (1 - F) \cdot q_s^i(k-1)] + [d_i(k) - q_{out}^i(k)] \} \quad ; \forall i \in I_n, n \in N, k \in K, E+1 \leq Li \leq H \quad (2.10)$$

$$l_{ni}(k) - \{l_{ni}(k-1) + \Delta T \{ [F \cdot (1 - \beta_{i0}) \cdot \sum_{j \in O_i} q_{out}^j(k - \tau) + (1 - F) \cdot q_s^i(k-1)] + [d_i(k) - q_{out}^i(k)] \} \} \leq M(1 - Z_{ni}(k)) \quad ; \forall i \in I_n, n \in N, k \in K, E+1 \leq Li \leq H \quad (2.11)$$

$$q_{out}^i(k) = (1 - u_{ni}(k)) [S_{ni}^g (1 - \xi_{ni}(k)) + S_{ni}^y \cdot \xi_{ni}(k)] + S_{ni}^g \cdot \xi_{ni}(k) \cdot u_{ni}(k) \quad ; \forall i \in I_n, n \in N, k \in K \quad (2.12)$$

$$u_{ni}(k) = \xi_{ni}(k-1) + u_{ni}(k-1) - 2\xi_{ni}(k-1)u_{ni}(k-1) \quad ; \forall i \in I_n, n \in N, k \in K \quad (2.13)$$

$$U_{ni}(k) = (U_{ni}(k-1) + \Delta T)(1 - \xi_{ni}(k-1)) \quad ; \forall i \in I_n, n \in N, k \in K \quad (2.14)$$

$$G_{\min}^i \leq U_{ni}(k) \leq G_{\max}^i \quad ; \forall i \in \text{In}, n \in \text{N}, k \in \text{K} \quad (2.15)$$

$$\sum_{i \in Q_p} \lambda_{ip} l_{ni}(k) \leq C_p \quad ; \forall i \in \text{In}, n \in \text{N}, k \in \text{K}, P \in \text{Bn} \quad (2.16)$$

$$u_{n1}(k) + u_{n2}(k) \geq 1; u_{n1}(k) + u_{n3}(k) \geq 1; \dots u_{n7}(k) + u_{n8}(k) \geq 1 \quad ; \forall n \in \text{N}, k \in \text{K} \quad (2.17)$$

The number of the vehicles discharged during the green time depends on whether the corresponding phase is oversaturated or not. If the phase is oversaturated, then it is equal to the capacity flow during green. If the phase is undersaturated, then it depends on the sum of the vehicle arrivals at the end of the waiting queue and the existing queue lengths at the end of time interval $k-1$ ($l_i(k-1)$). It should be noted that whether a phase is oversaturated or undersaturated cannot be predetermined. The model automatically determines the state of saturation during the optimization procedure. The above equation also implies that the queue lengths occurring at the end of each time interval must be non-negative. These considerations result in equations 2.6 through 2.11. Note that integer variables, Z_i are introduced for modelling the non-negativity of the queue lengths. Unfortunately, the model now becomes a complex Mixed Integer Linear Programming (MILP) problem because of the integer variables Z_i , and u_i described next.

For an internal link, the queue length $l_{ni}(k)$ is obtained by adding the balance between the new arrivals and the departures to the queue length $l_{ni}(k-1)$ at the end of the previous interval. This gives us equations 2.10 and 2.11, which includes the platoon dispersion variables described next. For the external approaches in the network (i.e., link number $L_i \leq E$), the queue lengths at the end of time interval k , $l_{ni}(k)$, can be represented as the sum of any queues transferred from the previous time interval and the difference between input and output at the current time interval, which results in equations 2.7 and 2.9.

The most important factor to be considered in a network of signals is the movement of vehicles from upstream intersections to downstream intersections, which requires the use of a reliable traffic model that can accurately reflect the movement of vehicles in the network. Research has already been conducted on the applicability of platoon dispersion as a reliable traffic movement model in urban street networks. Most of the research has shown that Robertson's model of platoon dispersion is reliable, accurate, and robust (Castle & Bonnaville, 1985; Axhausen & Korling, 1987). While the arrival patterns for the external approaches of the intersections are derived exogenously, the arrival patterns on the internal movements are obtained from the departures of the upstream intersections using Robertson's platoon dispersion equations. The F variable in equations 2.10 and 2.11 incorporates the platoon dispersion model into the constraints. In the presence of long queues, the assumption of a constant travel time may lead to some inaccuracy with regard to the queue evolution in each time step, but the delay calculation should be sufficiently accurate, as the time spent by the vehicles will still be captured over a number of time steps. Note that for consideration of further control measures such as route guidance and VMS (variable message signs) control, the turning movements are externally specified, through variables β appearing in equations 2.10 and 2.11.

The signal state of any phase i , $u_{ni}(k)$, at time step k , is given by equation 2.13. The first term represents the control decision at the end of time step $k-1$. The second term signifies the signal state of phase i at time step $k-1$. $u_{ni}(k)$ is a decision (binary) variable. If the signal state was red for time step $k-1$, i.e., $u_{ni}(k-1) = 1$, and the control decision at the end of the time step was to switchover, i.e., $\xi_{ni}(k-1) = 1$, then the signal state for time step k must be 0, which corresponds to a green state. The green time already used up by phase i at intersection n , at the

end of time step k , is computed by equation 2.14. The first term denotes the green time by the end of step $k-1$. Based on the control decision variable, $\xi_{ni}(k-1)$, green time is either increased by a duration of ΔT seconds or is increased by none. The standard minimum and maximum green time requirements are reflected in equation 2.15. The maximum queue length (capacity) constraint is fixed in 2.16.

Constraints are also needed to ensure that conflicting phases are not given the green indication during any split. Since u 's are constrained to be binary variables in the formulation, it is easy to ensure that no more than one of any combination of two non-permissible phases has its corresponding u set to 0 (i.e., green time). In this formulation, the standard NEMA numbering convention for the different movements are used to identify the different phases. Thus, the inadmissible combinations of NEMA phases are phases numbered 1 and 2 (EW left and opposing through), 1 and 3 (EW left and SN left), 1 and 4 (EW left and NS through), and so forth, through 7 and 8 (NS left and opposing through). These constraints are shown above in 2.17.

3. DEVELOPMENT OF SOLUTION PROCESS

In this section we present an efficient solution approach, based on a modification of the network simplex algorithm, to solve the model formulation proposed in previous section. Large-scale mixed integer linear-programming (MILP) problems such as above are normally solved with branch and bound techniques with repeated simplex solutions of the LP problem within the MILP. We next explain the efficient network simplex algorithm that can replace the standard simplex for LP problems of network form.

3.1 Network Simplex Algorithm

In the network simplex algorithm, we need not explicitly maintain the matrix representation (known as the simplex tableau) of the linear program and can perform all the computations directly on the network. If the dynamic optimal traffic signal model is solved by a simplex tableau, there are some disadvantages. In the simplex tableau approach, we need to check each possible *pivot* to find out the minimum objective value. As a result, the enormously large number of *pivots* can make the computation prohibitively expensive when the number of time periods and the number of intersections on the network increase significantly.

A specialized network simplex algorithm is used to efficiently operate at any given time interval. It is shown that the algorithm is more efficient in those problems for which its structure is a large-scale network form (for more detailed description about the efficiency of this method and its illustrated examples, refer to the references of Ahuja et al. (1993) and Kennington et al. (1980)).

The network simplex algorithm maintains a feasible spanning tree structure and moves from one spanning tree structure to another until it finds an optimal structure. At each iteration, the algorithm adds one arc to the spanning tree in place of one of its current arcs. The entering arc is a nontree arc violating its optimality condition. The algorithm (1) adds this arc to the spanning tree, creating a negative cycle loop, (2) sends the maximum possible flow in this cycle until the flow on at least one arc in the cycle reaches its lower or upper bound, and (3) drops an arc whose flow has reached its lower or upper bound, giving us a new spanning tree structure. Because of its relationship to the primal simplex algorithm for the linear programming problem,

this operation of moving from one spanning tree structure to another is known as a *pivot operation*.

The network simplex algorithm maintains a feasible basis structure at each iteration and successively modifies the basis structure via pivots until it becomes an optimum basis structure. The special structure of the basis enables the simplex computations to be performed efficiently.

3.2 Proposed Solution Algorithm and the System Control Logic

Graphical Presentation of Network Programming Formulation

The model presented in section 2 provides the coupling constraints in the network optimization formulation. Each intersection has its own constraints in terms of the phase sequences and conflicting movements. The arrival flows in each time period at each intersection are dependent on the departure flows from other intersections through these platoon dispersion coupling constraints. The research has identified a key aspect of the formulation, namely that the linear optimization sub-problem involves extremely sparse matrices, thanks to the network structure. One key reason for this is the nature of the platoon dispersion model which is similar to a multi-period inventory flow model.

Note that the Robertson's platoon dispersion equation means that the traffic flow ($q_s^j(k)$), which arrives during a given time step at the downstream end of a link, is a weighted combination of the arrival pattern at the downstream end of the link during the previous time step ($q_s^j(k-1)$) and the departure pattern from the upstream traffic signal τ seconds ago ($q_m^j(k-\tau)$). However, the recurrence platoon dispersion equation (2.4) can be transformed as the following form:

$$q_s^j(k) = \sum_{i=\tau}^{\infty} F(1-F)^{i-\tau} q_m^j(k-i) \quad (3.1)$$

where F is a dispersion parameter and τ is a travel time coefficient between the upstream and downstream intersection. The summation can be truncated after a reasonable number of terms (say 10 to 15).

The transformed platoon dispersion equations directly translate to links on a time-expanded network, as in Figure 1. A complete graphical representation of the network structure including the constrained turning movements at the intersection for multiple time steps is too complicated to show here; however, the network inventory-flow nature of the linear platoon dispersion equations should be clear from Figure 1. The network size of the whole traffic signal control problem can be reduced after each iteration of computation to a simpler and smaller network. This reduction process simplifies the difficulty of incorporating directly network simplex algorithm to platoon dispersion-based network traffic signal control problem.

<Insert Figure 1>

The optimization algorithms employ the two standard principles of implicit enumeration: network simplex algorithm and branch and bound. Branch and bound is mainly used to obtain integer solutions when combined with some other relaxation method. Alternatively, the network simplex algorithm is mainly applied to solve minimum cost flow problems (i.e., delay minimization); so it appears as subroutines in branch and bound methods. The elements of a

network simplex algorithm are demands, nodes, arcs, costs, and capacities which determine the value of the objective function at each iteration.

Non-standard Network Flow Problem and Special-case Simplex

The network problem presented above poses one difficulty, however, and this is the fixed split required from the upstream node to the platoon dispersion links. The standard network simplex algorithm does not involve such "fixed nodal splits fractions" (F , $F(1-F)$, $F(1-F)^2$ etc., as in Figure 1, are these fractions). This poses a difficulty during the basis update step of network simplex. The reason is that when we attempt to change the flow on any network loop (cycle) involving any of the platoon dispersion arcs, the ratio of flows among the platoon dispersion arcs change. This has to be avoided to retain the split ratios, which means the simplex basis update schemes no longer apply. Thus, the class of network problems with such fixed nodal splits are considered to be a class of problems with much higher difficulty which do not have easy solutions.

We develop a technique to handle this problem and the basic idea is straight forward. We can see that the arcs with costs in the network are only the inventory (queuing) arcs shown vertically in Figure 1. This means that a restricted version of basis update for such a graph that is not based on normal augmenting cycles (loops) can be developed. This essentially involves the selection of a few inventory arcs (i.e., upstream intersections' arcs) and updating the current solution of split fraction arcs in the whole network. Once the inventory arc's flow (note that this is not actual traffic flow, but rather queue length) is updated, the exit flows and the platoon dispersion links can be updated. The algorithm operates in a decomposed fashion, starting from peripheral nodes and moving inwards, updating platoon arcs and inflows to other nodes. Note that the platoon dispersion arc flows (which in turn depend on the signal settings) are adjusted after every iteration, and the network simplex operates essentially independently for each intersection. After about 4 or 5 iterations, all the nodes are normally reached in reasonably sized networks, and the signal settings converge in a few more additional iterations (enough to adjust flows along loops in the network).

This approach of the solution process is implemented in iterations as described in previous section by using network simplex algorithm as the subroutine of branch and bound method. In every iteration, first of all, the branch and bound is performed to prevent conflicting movements. As equation (3.1) shows, we can easily find the flow rates of the initial platoon (i.e., $q_{in}^j(k)$) during time step k by ignoring the split phenomenon on the road link. This is followed by splitting these platoon flow rates as the initial arriving flows of downstream intersection. The algorithm continues by finding the second iteration's flow rates as the initial platoon flow rates of next iteration's under the implementation of branch and bound, and so on.

A platoon dispersion-based problem with flow split constraints is constructed with the introduction of an arrival flow prediction/estimation concept. Based on the Branch & Bound-and-Network-Simplex algorithm, the solution of the first iteration (usually the minimum cost flow computation of upstream intersections in a network) facilitates an initial solution of the second iteration. The solution procedure for each of the iterations can be stated as the following system control strategy.

System Control Logic

The following steps describe the basic control strategy governing the proposed models for network traffic signal control with platoon dispersion constraints. Given the aforementioned

system and all functions of its key elements, the operational procedures are summarized below.

- Step 0. All the external intersections included in the set N are initialized with external entry flows and initial branch and bound signal settings.
- Step 1. For intersection n (in current set N) for each phase i carry out the branch and bound steps. If all intersections are completed, go to step 7.
- Step 2. Check the minimum and maximum green constraints for all time steps k :
- Condition 1:* If green time is less than the minimum green time (i.e., $U_i(k) \leq G_{\min}^i$), then extend the green. For instance, if $\xi_i(k) = 0$, then $U_i(k)$ is updated.
- Condition 2:* If $U_i(k)$, the green time already used up by phase i at time step k , has reached the maximum green time G_{\max}^i (i.e., $U_i(k) \geq G_{\max}^i$), then the green is terminated immediately, i.e., $\xi_i(k)$ is changed to 1.
- If the integer variables for all time step k remain the same as in the previous iteration, update n to the next intersection and go to step 1. Else go to step 3.
- Step 3. Examine and prevent the conflicting movements which happen at intersections using the branch and bound method by setting the constraints $u_i(k) + u_j(k) \geq 1$, where $i, j \in I_n$ and u_i and u_j are binary integers which are either 0 or 1.
- Step 4. Using the Network Simplex based algorithm to get the minimum cost flow solution (i.e., delayed vehicles, TD) of the network for the green extension option evaluated by the branch and bound. (Here Network Simplex is operated at the given intersection and could even conceptually be replaced with vehicle clearance rules if they can guarantee optimality.)
- Step 5. If the performance index is the benefit of giving a green which is compared with that of terminating it by computing the tradeoffs incurred in vehicle delays and it is negative, then the optimal decision is not favorable to the intersection and a switchover decision is possible. Otherwise, the current green can be extended for another ΔT seconds, i.e., $U_i(k+1) = U_i(k) + \Delta T$.
- Step 6. Go to step 2 to ensure of switchover being within the min-max constraints.
- Step 7. Update platoon dispersion arc flows and the corresponding inflows to existing intersections as well as new intersections joining the set N . If the platoon dispersion arc flows are same as in the previous outer iteration (over the last set of intersections N), i.e., no signal settings changed, then STOP; Else, go to step 1.

In the proposed models, the control decision is made every 5 seconds (i.e., duration of a time step) depending on the comparison of benefits between extending green and terminating it. The control strategy makes use of real-time traffic state conditions, instead of pre-stipulated strategies.

3.2 Real-time Applications – Rolling Horizon Approach

The rolling horizon approach, used previously for production-inventory control (Wagner, 1977), and in transportation engineering for online demand-responsive traffic signal control (Gartner, 1983), provides a practical method for addressing the real-time traffic signal control problem. It is especially suited for problems requiring future demand information for the entire planning horizon. The basic idea of the rolling horizon approach is to use currently available information and near-term forecasts with some degree of reliability to solve a problem online

while preserving the effectiveness of the computational procedure in determining "good" control strategies (Peeta et al., 1995).

On the basis of an estimate of the current state of the system, principally the current lengths of the queues on all approaches, an optimal signal plan (or policy) is sought for the duration of the rolling horizon. Only the first part of the plan is implemented before the horizon is rolled forward and the optimal signal plan again sought, with the benefit of the latest detector data. The rolling horizon must therefore be long enough to allow the first part of the optimal plan to be determined uniquely.

In particular, each optimization run is based on an updated initial condition that includes the current values of all states variables, i.e. of all network queues. Furthermore, each optimization run requires availability of historical or real-time forecasts of traffic demand. Therefore, availability of queue measurements or estimates, and of demand forecasts is a major requirement for a real-time implementation of the method (Papageorgiou, 1995). An optimal policy for the entire stage is calculated but is implemented only for the roll period. The projection horizon is then shifted or *rolled* ahead, and the optimization is then reactivated over this stage with new real-time measurements and new demand forecasts.

The implementation aspects represent the major challenge and sensitivity analyses are necessary to address questions regarding the choice of parameter values (for example, roll period, stage length) and their influence on the effectiveness of the control procedure. Also of significance is the deployment of detectors and issues such as whether the upstream output detectors are needed at all. The results reported here for the rolling horizon applications are for demonstrative purposes only, as the effects of the above parameters have not been extensively studied.

4. MODEL EVALUATIONS AND EMPIRICAL APPLICATIONS

To evaluate the performance of the algorithm on a real-world network context using microscopic simulation, the data from Irvine Traffic Management Center (TMC) was collected for a selected field site. The evaluation is based on comparing the solutions to existing actuated signal control of the field site, as well as the "best" possible actuated control.

The selected study network is a subnetwork located along the San Diego Freeway (I-405) and Laguna Freeway (SR-133) corridors through the City of Irvine. The network is located at the triangle between the Santa Ana Freeway (I-5) and San Diego Freeway (I-405) up to Jeffrey. The block length on these links varies from 800 to 2350 feet. Most have two through lanes, with two left turn pockets which have exclusive phases. The Data Sets were collected from the Irvine TMC in September 1996. Figure 2 shows a schematic of the NETSIM representation for the Irvine study network.

<Insert Figure 2>

Identification of Simulated Samples

Performance comparisons of the new algorithm against the existing actuated control settings are not sufficient, since the existing settings may not be the best possible for the given arrival pattern. The approach taken here is to find a large number of possible sample actuated control plans, so that the "Best" possible plan can be determined after all those plans are individually simulated under the same arrival patterns. Five parameters, which are regarded as

important components to design an efficient actuated controller, were selected and randomly varied (within appropriate ranges) across these sample plans. They are:

- (1) minimum green interval for each phase per intersection (secs),
- (2) maximum green interval for each phase per intersection (secs),
- (3) vehicle extension unit for each phase per intersection (secs),
- (4) initial green interval for each phase per intersection (secs), and
- (5) phase reduction for each intersection.

Other factors such as the turning proportions and arrival rates were kept constant in all the cases. Under various combinations of these parameters, 10000 different actuated control settings were found. Each of these control scenarios was then simulated using TRAF-NETSIM. In addition, because the selected study network has only very light traffic flows on some of its external entries, another set of simulations were performed with assumed increased traffic flows ranging from 50% to 200% to these external entries. The performance results from these cases were used to find the "Best" and "Worst" cases of control scenarios (the term "worst" only referring to the worst in the studied scenarios, as even worse signal settings are of course possible!).

For the case of the existing entry flow rates at the low-volume entries, the results showed a "Worst" case scenario with a total delay of 22,247 vehicle-seconds and a "Best" case delay of 16,200 vehicle-seconds. The existing timing plans in Irvine appeared to be reasonably good, at a delay of 17,466 seconds.

The complete time-horizon (full-knowledge) optimization with the new algorithm as well as the rolling horizon approach (described in section 3) were applied to the Irvine subnetwork, to study the performance in comparison to the "best" case benchmark actuated control settings. The rolling horizon approach applied here is based on a roll period of 3 time steps, and a stage length of 5 time steps. The application of rolling horizon here is for demonstrative purposes only, and the results reported here should only be taken in that light. Much more extensive studies with appropriate roll periods and stage lengths need to be conducted to determine the effectiveness of the rolling horizon approach, not to mention the effects of the stochasticity and uncertainty in the detection and prediction of traffic volumes.

4.1 Development and Evaluation of the Control Model

The layout of the experimental network represented as a NETSIM coding form was given in Figure 2. Two studies were conducted for a simulated 36 time-step period (each step is 5 seconds) after an appropriate initialization period for TRAF-NETSIM: one with the current collected traffic arrival rates over the total period, and another with an assumed increase of arrivals ranging from 50% to 200% to the network externals during the total period. The signal operation considered in the optimization model is designed with a 4 phase control, permitted left turns, minimum green of duration 10 seconds, maximum green of duration 80 seconds and an amber of 3 seconds.

NETSIM generates the simulated output data of arrival flows every 5 seconds which is fed into the optimization control model. In turn, the optimization control model generates signal control data which is fed into NETSIM as the traffic control input data. NETSIM provides the summary of traffic measures of effectiveness (MOE's).

From the simulation outputs, the performance was tested based on the vehicle delay that was obtained at the end of every 5 seconds (duration of a time step) for the entire network, for 36

time steps. And, based on the results of preliminary pilot runs, for variance reduction and the NETSIM's inherent variability, five replications of 36 time steps runs using different Random Number Seeds (RNS) were made for each scenario.

4.2 Simulation Evaluation

A computer simulation of traffic operations responding to the proposed control algorithms (including both optimization and rolling horizon), and to the existing actuated signal timing plans indicated appreciable improvement in traffic operations with the proposed timing plans. The simulation results presented here are average values of five simulation runs each of 3 minute duration.

Vehicle delay (vehicle-seconds) was chosen as the measure of effectiveness (MOE) to indicate the relative efficiency of these three strategies (i.e., actuated control, optimization control, and rolling horizon control), because this MOE is generally the most important consideration from the viewpoint of the whole network system and it is clearly defined and easily measurable.

Figure 3 and 4 show the network-level cumulative delay during the 36 time steps for the full-horizon optimization control, rolling horizon control, and actuated signal control logics. Figure 3 shows a comparison of system-wide vehicle delay for the base peak hour external entry volumes and Figure 4 shows the results for the assumed increased peak hour external entry volumes.

<Insert Figure 3 & 4>

As can be seen from the above simulation results, the full-horizon control logic yields better performance (3.05%) than the "Best" actuated control timing plans in the base volume case simulated by NETSIM on the basis of the whole network-wide system. Moreover, the rolling horizon approach shows inferior performance than actuated signal control methods in the base case which is not surprising since actuated control could be more responsive than the 5 second rolling horizon control. In order to investigate in-depth the performance of the optimization control algorithm applied in different and higher traffic demands, another 10,000 NETSIM samples were simulated based on assumed increased traffic volumes as depicted previously and the "Best" actuated signal timing plans was chosen as the benchmark for comparison use. For the advanced traffic case, the overall system delay shows that the optimization control algorithm performs much better than actuated signal control (16.05%). Especially, for most of the links they also show significant improvement compared to actuated control method.

These above simulation results show the implications that actuated signal control runs well in light traffic situations, while the proposed optimization control and rolling horizon control perform well when traffic flows are increasing to moderate or heavy level. Much more detailed field evaluations are required before complete conclusions are derived on the algorithm and its rolling horizon implementation.

Though some recoding to speed up the algorithm is possible, we find that computational performance is good. The above problem of 36 steps (3 minutes) can currently be solved in about a minute on a Pentium processor, and the solutions are even faster under rolling horizon solution due to better starting solution. Distributed processing for the solutions can make it even faster. On the other hand, the reason why this simulation only utilized 36 time steps (i.e., 3 minutes) to evaluate the performance of the control algorithm can be stated as follows:

- (1) First, when the real-time control case (e.g., incorporate the rolling horizon technique into the algorithm) is implemented, there won't need too many time steps' arrival data for every optimization computation. In real time, we may even need only one time step's data for the model and thus the computation time will reduce a lot more.
- (2) Second, there exists some difficulties of manipulating too many time steps' arrival data in this optimization model (i.e., network simplex programming model) by hand. Even take a very simple network case with around 20 time steps as an example, the problem will be such a many orders of magnitude large one and its number of constraints for the optimization network simplex algorithm may exceed ten thousands or even more. Not to mention a 10-node network with 36 time steps simulation period case as proposed in our this research.

5. CONCLUSIONS AND FUTURE RESEARCH

This research presents a systematic approach to network traffic signal control problem. The approach involves formulation and solution of a linear optimization problem for multiple intersections in a network, connected by explicit constraints capturing traffic movement on the links connecting the intersections.

The proposed model formulation is distinct from other models in that the traffic arrival platoon is explicitly incorporated into the network signal control model. The platoon dispersion constraints directly translate to links of a time-expanded network, and thus the problem has the form of a linear multi-commodity network flow problem. Even though, the fixed nodal splits for the platoon constraints render it a non-standard problem, we develop efficient special-purpose version of network simplex to solve the problem, in conjunction with a branch and bound scheme for the integer signal constraints. The results from microscopic simulation studies in a real world network are very promising.

Several operational aspects of the algorithm and its potential application remain to be studied. One option of particular interest is a hierarchical operation where the algorithm provides strategic control for an extended traffic network with an updating period of several seconds and complemented by an inferior, short-term reacting, possibly decentralized direct-control layer, and a superior adaptation layer that provides updated demand forecastings, estimation, and detection information (Papageorgiou, 1984; Stephanedes and Chang, 1993). Another possibility (and perhaps necessity) is to use the algorithm for smaller subnetworks based on real-time computational concerns, and to coordinate multiple subnetworks using a supervisory program.

Some aspects related to this research can be considered to be studied in the future. They include the following issues:

1. Integration of the available demand-forecasting algorithms and the presented signal control modelling framework so as to enable a real-time use of the overall methodology.
2. Comparison of observation of flows to observation of queues based on stochastic simulation. The stochastic feedback and filtering schemes for detected data can be incorporated in the overall methodology.

3. Study of optimization benefits and computational trade-offs in alternate rolling horizon implementations via sensitivity analyses.

REFERENCES

- Ahuja, Ravindra K., Magnanti, Thomas L and Orlin, James B., Network Flows Theory, Algorithms, and Applications. Prentice Hall, New Jersey, 1993.
- Axhausen K. W. and Korling, H. G., Some Measurements of Robertson's Platoon Dispersion Factor, Transportation Research Record 1112, 71-77, Transportation Research Board, Washington D.C., 1987
- Boillot, F, Blosserville, JM, Lesort, JB, Motyka, V, Papageorgiou, M, and Sellam, S, "Optimal Signal Control of Urban Traffic Networks", In International Conference on Road Traffic Monitoring and Control (6th), Institution of Electrical Engineers, London, England, 1992.
- Brooke, A., Kendrick, D., and Meeraus, A., "GAMS: A User's Guide", World Bank, New York, 1988.
- Castle D. E. and Bonnaville, J. W., Platoon Dispersion over Long Road Links, Transportation Research Record 1021, 36-44, Transportation Research Board, Washington DC., 1985
- Cremer M., Origin-destination matrix: Dynamic estimation. In M. Papageorgiou, (Ed.), Concise Encyclopedia of Traffic and Transportation Systems, Pergamon Press, Oxford., 1991, pp. 310-315.
- Donati, F., et al., "A hierarchical decentralized traffic light control system. The first realization", IFAC 9th World Congress, Hungary, Budapest, Vol. 11, 11G/A-1, 1984.
- Gartner, N. H. et al., "Evaluation of Optimized Policies for Adaptive Control Strategy," Transportation Research Record 1324, Transportation Research Board, Washington, D.C., 1992
- Gartner, N.H., OPAC: A demand-responsive strategy for traffic signal control Transportation Research Record, 906, 1983, pp. 75-81.
- Gordon R.L., A technique for control of traffic at critical intersections. Transpn Sci. 4, 279-287, 1969.
- Henry, J.J., et al., "The PRODYN real time traffic algorithm", 4th IFAC-IFIP-IFORS Conference on Control in Transportation Systems, Germany, Baden-Baden 307, 1983
- Henry, J.J., Farges, J.L., PRODYN. Proc.,6th IFAC-IFIP-FORS Symp. Transportation. Pergamon, Oxford, 1989, pp. 505-7.
- Hunt P. B., Robertson D. I., Bretherton R. D. and Royle, M. C., The SCOOT on-line traffic signal optimisation technique. Traffic Engineering & Control 23, 1982, pp. 190-192.
- Hunt, P.B., et al., TRRL Report 1014, 1981.
- Jayakrishnan R., Tsai, W.K., Prashker J.N. and Rajadhyaksha, S., A Faster Path-Based Algorithm for Traffic Assignment, Transportation Research Record, 1443, 1994, 75-83.
- Kennington, Jeff L. and Helgason, Richard V., Algorithms for Network Programming. John Wiley & Sons, New York, 1980.
- Kim, Youngchan, Development of Optimization Models for Signalized Intersections During Oversaturated Conditions. Ph.D. Dissertation, Texas A&M University, 1990.
- Larry Head, K., Mirchandani, Pitu B. and Shelby, Steve, Estimation and Prediction in a Real-Time Traffic Adaptive Signal Control Logic: The RHODES Prototype. INFORMS Meeting, San Diego, May 1997.
- Lee B., Crowley K. and Pigantaro L., Better use of signals under oversaturated flows. Transpn

- Res. Board SR153, 60-72, 1975.
- Little, J., Kelson, M. and Gartner, N., MAXBAND: A Program for Setting Signals on Arteries and Triangular Networks. Transportation Research Record 795. National Research Council, Washington, D.C., 1981, pp. 40-46.
- Longly D., A control strategy for a congested computer controlled traffic network. Transpn Res. 2, 391-408, 1968.
- Lowrie, P. R., "SCATS Principles, methodology, algorithm", IEE Conf. on Road Traffic Signalling, IEE Publication 207, England, London, 1982, pp. 67-70
- Lowrie, P., SCATS - A Traffic Responsive Method of Controlling Urban Traffic. Sales information brochure published by Roads & Traffic Authority, Sydney, Australia, 1990.
- Luk, J., Two traffic-responsive area traffic control methods: SCATS and SCOOT. Traffic Engineering and Control, 1984, pp. 14-20.
- Messer, C.J., Hogg, G.L., Chaudhary, N.A., and Chang, E.C.P., Optimization of Left Turn Phase Sequence in Signalized Networks Using MAXBAND 86 - Volume 2. User's Manual. Report FHWA/RD-84/, FHWA, U.S. Department of Transportation, Washington, D.C., 1986.
- Michalopoulos, Panos G, Stephanopolos George, Optimal control of oversaturated intersections: theoretical and practical considerations, Traffic Engineering & Control, 1978.
- Michalopoulos, Panos G, Stephanopolos George, Oversaturated Signal Systems with Queue Length Constraints-I, Transportation Research. Vol. 11, 413-421, 1977.
- Michalopoulos, PG, Intersection control through video image processing: executive summary, prepared for the Minnesota Dept. of Transportation, Department of Civil and Mineral Engineering, University of Minnesota, St. Paul, Minn, 1992.
- Miller, A.J., A computer control system for traffic networks. In: Almond J. (ed) Proc. 2nd Int. Symp. Theory of Road Traffic Flow. Organisation for Economics Cooperation and Development, Paris, 200-220, 1965.
- Papageorgiou M., An Integrated Control Approach for Traffic Corridors. Transportation Research C, vol 3, No 1, 1995, pp. 19-30.
- Papageorgiou M., Multilayer control system design applied to freeway traffic. IEEE Transactions on Automatic Control 29, 1984, pp. 482-490.
- Peeta, Srinivas and Mahmassani, Hani S., Multiple User Classes Real-Time Traffic Assignment for Online Operations: a Rolling Horizon Solution Framework, Transportation Research C, vol 3, No 2, 1995, pp. 83-98.
- Robertson, D.I. and Bretherton, R.D., "Optimizing Networks of Traffic Signals in Real-Time: The SCOOT Method", IEEE Trans. on Vehicular Technology, 40(1), 1991, pp. 11-15.
- Robertson, D.I., "TRANSYT: A Traffic Network Study Tool", TRRL Report LR 253, London, 1969.
- Ross, D., Sandys R. and Schlaefll T. A computer control scheme for critical intersection control in an urban network. Stanford Research Institute, Menlo Park, California, 1970.
- Schrage, Linus E., "User's Manual for LINDO", 1984.
- Sims, A. The Sydney Coordinated Adaptive Traffic System. Proc. ASCE Engineering Foundation Conference on Research Priorities in Computer Control of Urban Traffic Systems, 1979, pp. 12-27.
- Stephanedes Y.J. and Chang K.-K., Optimal control of freeway corridors. ASCE Journal of Transportation Engineering 119, 1993, pp. 504-514.
- Sun C., R. Jayakrishnan and Tsai, W.K., Computational Study of a Path-Based Algorithm and Its Variants for Static Traffic Assignment, forth coming in Transportation Research Record,

- 1996.
- Tarnoff, PJ and Gartner, N, Real-Time, Traffic Adaptive Signal Control, ATMS Conference, St. Petersburg, Florida, 1993.
- Tsay, H. and Lin, L., New Algorithm for Solving the Maximum Progression Bandwidth. Transportation Research Record 1194, TRB, National Research Council, Washington D.C., 1988
- Wagner H.M., Principles of operations research. Englewood Cliffs, NJ: Prentice-Hall , 1977.
- Wey, Wann-Ming and Jayakrishnan, R., A Network Traffic Signal Optimization Formulation With Embedded Platoon Dispersion Simulation, Paper Number: 971337, Transportation Research Board, Washington, D.C., 1997.

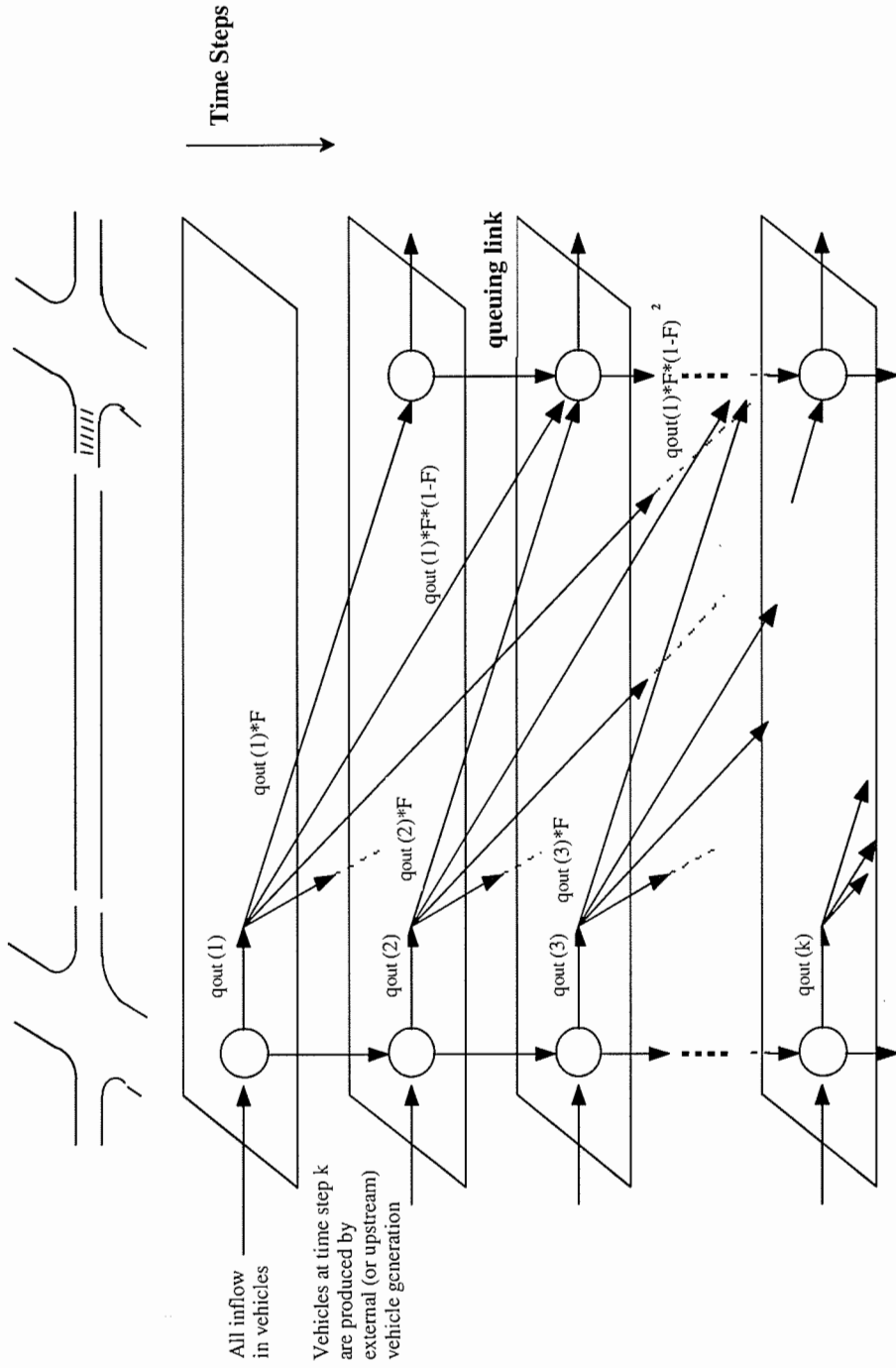


Figure 1. Graphical Representation of Network Programming Formulation

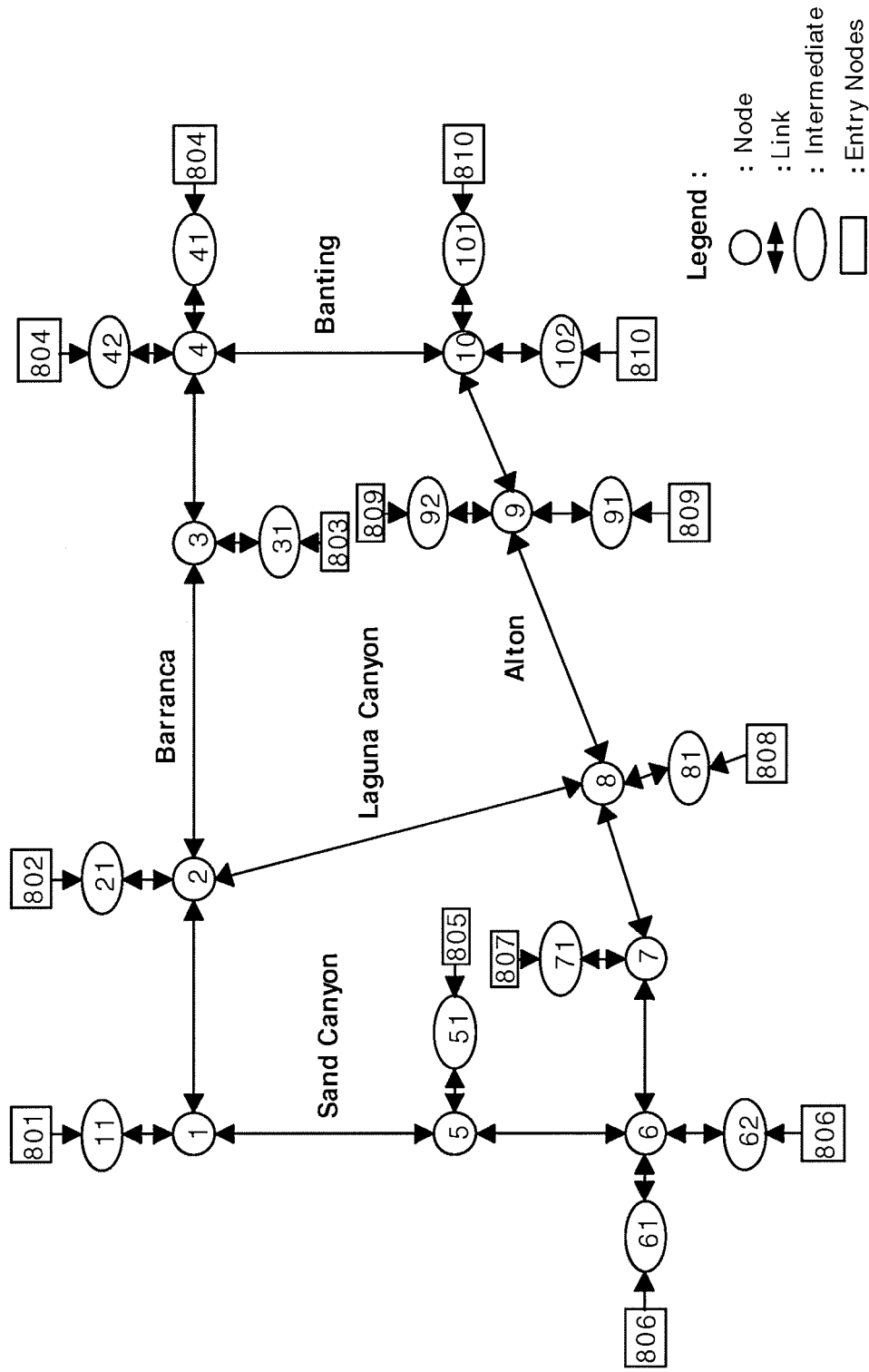


Figure 2. NETSIM representation of the Irvine Study Network

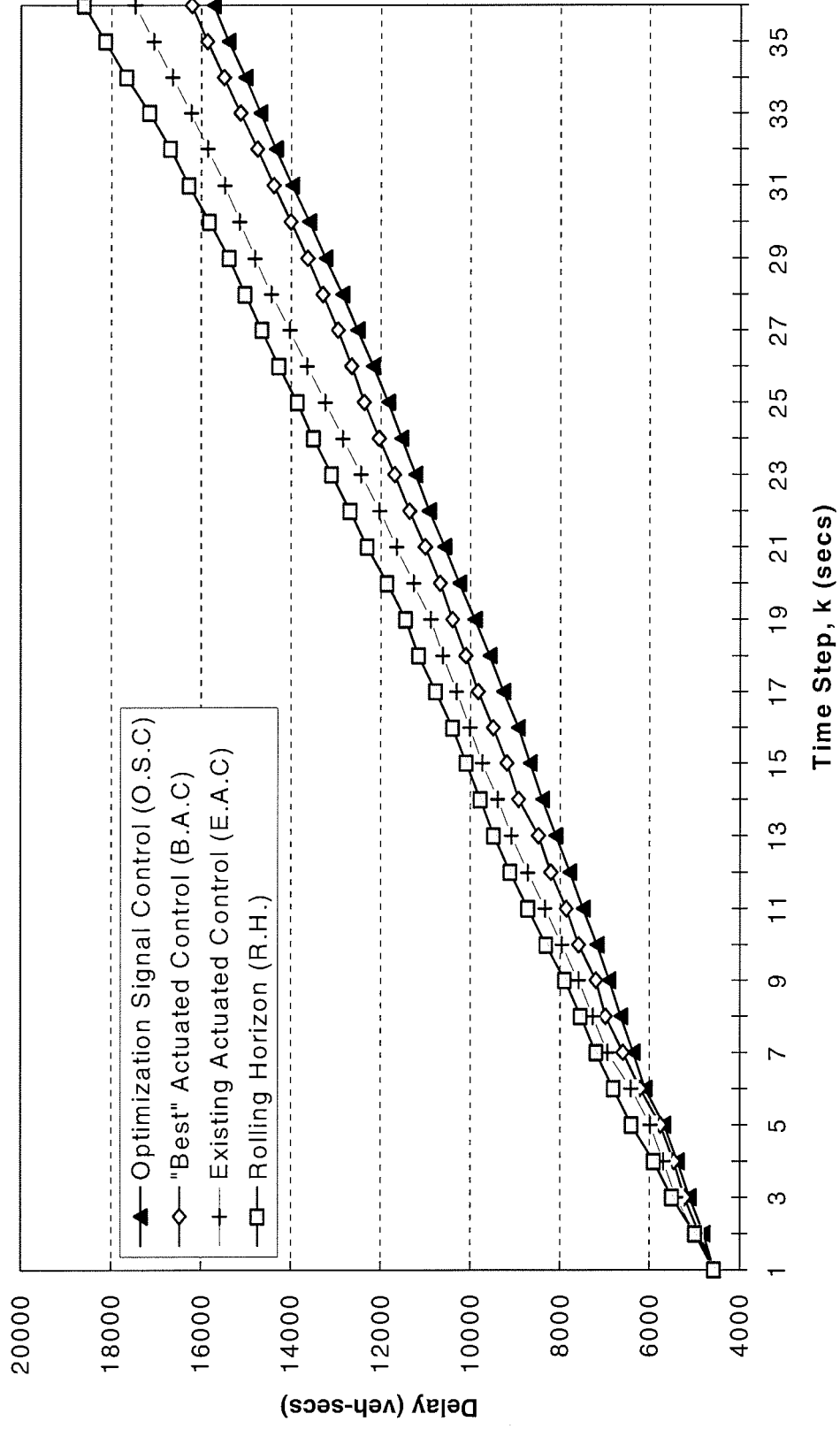


Figure 3. Cumulative System Delay by Different Signal Control Strategies (base volume peak hour case)

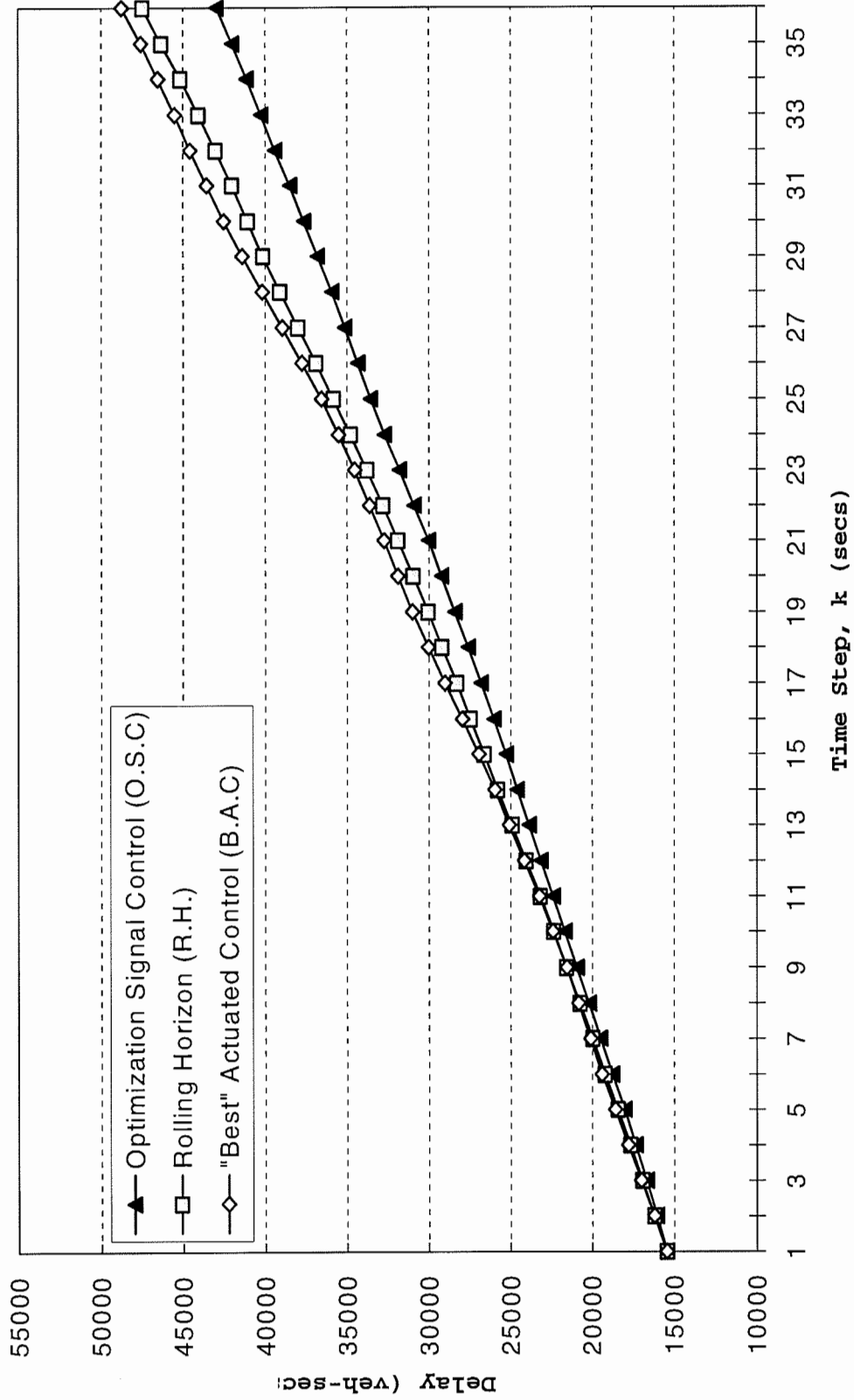


Figure 4. Cumulative System Delay by Different Signal Control Strategies (increased volume peak hour case)