Title
# Effectiveness of Adaptive Traffic Control for Arterial Signal Management



Authors
Gomes, Gabriel
Skabardonis, Alexander

Publication Date
2009-03-01

CALIFORNIA PATH PROGRAM
INSTITUTE OF TRANSPORTATION STUDIES
UNIVERSITY OF CALIFORNIA, BERKELEY

# Effectiveness of Adaptive Traffic Control for Arterial Signal Management

**Gabriel Gomes, Alexander Skabardonis**

**California PATH Working Paper**
**UCB-ITS-PWP-2009-2**

This work was performed as part of the California PATH Program of the University of California, in cooperation with the State of California Business, Transportation, and Housing Agency, Department of Transportation, and the United States Department Transportation, Federal Highway Administration.



Final Report for Task Order 5322

March 2009

ISSN 1055-1417

CALIFORNIA PARTNERS FOR ADVANCED TRANSIT AND HIGHWAYS

# EFFECTIVENESS OF ADAPTIVE TRAFFIC CONTROL FOR ARTERIAL SIGNAL MANAGEMENT

**Gabriel Gomes, Alexander Skabardonis**

**FINAL REPORT for PATH TO 5322**

## ABSTRACT

A number of adaptive control algorithms have been developed in the US and overseas. However, the practical implementation of adaptive control is limited especially in California. There is a need to develop adaptive control algorithms, evaluate their performance through a field test, and develop a deployment plan for possible Statewide application. The objectives of the study are identify and select the most promising of existing adaptive control algorithms, develop improved algorithm(s) as appropriate, conduct field tests on real-world arterials, and develop recommendations for deployment of adaptive control.

This report describes the work performed in the first phase of the research conducted under PATH Task Order 5322. It summarizes the findings from the literature review and presents the study methodology. A section of the Pacific Coast Highway in the city of Lomita was selected as the test site for evaluation of adaptive signal control strategies. A methodology was developed to develop a time-varying origin-destination (OD) matrix from the system loop detector data at the site. The calibrated OD matrix was applied to a microscopic representation of the site created in PARAMICS. Finally, a plug-in for simulating signal control in PARAMICS was written. It simulates non-adaptive strategies such as pretimed, isolated actuated, coordinated actuated, traffic responsive, and critical intersection control, as well as adaptive strategies such as RHODES and TUC.



***Keywords:*** *Traffic signals, adaptive control, simulation, origin-destination matrix estimation*

**ACKNOWLEDGEMENTS**

This work was performed by the California PATH Program at the University of California at Berkeley, in cooperation with the State of California Business, Transportation and Housing Agency, Department of Transportation (Caltrans), Division of Research and Innovation (DRI) (Interagency Agreement #65A0208, Task Order 5322). The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the data presented herein. The contents do not necessarily reflect the official views or policies of the State of California.

The authors thank Hamid Rifaat of Caltrans DRI for his support and advice during the project. Mr. Jorge Fuentes of Caltrans District 7 provided data for the selected test site and much on-site expertise. We also thank the project technical advisory committee members Fred Yazdan, Tam Nguyen, Ahmad Rastegarpour, Steve Hague, and Kai Leung of Caltrans for their comments and suggestions throughout the study. Dr. Muñoz helped to configure the CTNET system, gather the data, and select the final data set. Dr. Andy Chow and Eleni Christofa assisted with the preparation of the report. David Lucas, and Professors Larry Head and Pitu Mirchandani from the University of Arizona assisted with the modeling of the RHODES adaptive signal control strategy.

## EXECUTIVE SUMMARY

A number of adaptive control algorithms have been developed in the US and overseas. However, the practical implementation of adaptive control is limited especially in California. There is a need to develop adaptive control algorithms, evaluate their performance through a field test, and develop a deployment plan for possible Statewide application. The scope of the project is to identify and select the most promising of existing adaptive control algorithms, develop improved algorithm(s) as appropriate, conduct field operational tests on real-world arterial corridors, and develop recommendations for deployment of adaptive control. The report describes the work performed and the finding in the first phase of the project (PATH Task Order 5322).

- A test site with seven signalized intersections was chosen for evaluation of adaptive signal control strategies. The selected test site is a section of the Pacific Coast Highway running through the city of Lomita, managed by Caltrans District 7. Over seventy days of loop detector data were collected for this site using the CTNet software. Additional data on geometrics and signal settings were collected from Caltrans staff and field visits to the site. A model of the test site was constructed in the PARAMICS microscopic simulation model to evaluate the effectiveness of control strategies prior to field implementation.
- A methodology was devised for synthesizing a time-varying OD matrix from the loop detector data. This procedure makes few assumptions on the availability of data, and allows the user to directly affect the resulting turning flows by manipulating the input parameters. The calibrated OD matrix was input to the PARAMICS model to simulate existing operating conditions at the test site.
- A signal control plug-in was written to model control strategies with PARAMICS. It simulates non-adaptive strategies such as pretimed, isolated actuated, coordinated actuated, traffic responsive, and critical intersection control, as well as adaptive strategies. The plug-in can be applied to any PARAMICS network. The plugin is based on a control interface which allows it to communicate with arbitrary "black box" algorithms via standard *hold*, *force-off*, and *omit* messages. When connected to an external control algorithm, the plugin acts in much the same way as a real controller, providing safety and minimum phase timing guarantees.

Ongoing research in the second phase of the project under the PATH Task Order 6322 includes the following tasks:

- Evaluation of the selected existing and proposed adaptive control algorithms through simulation. The best adaptive control algorithm will then be selected for field testing based on the analysis of the simulation results.
- Development of a test plan for field implementation. The test plan will specify the hardware and software requirements and level of effort required for field implementation of the adaptive algorithm. The test plan will also specify the duration of the field experiment, the method of data collection, the duration of the field experiment, and the analysis of the results.
- Development of guidelines for deployment of adaptive control based on the findings from the simulation results. The issues to be addressed in the deployment plan include the system performance, costs (installation, operations and maintenance), system reliability, and integration with other traffic management systems. We also provide an assessment on how well the proposed adaptive system adapts to changing traffic conditions, and how well they, if not stand-alone system, coexist with standard time-of-day signal control systems.

# CONTENTS

## LIST OF FIGURES



## LIST OF TABLES

# CHAPTER 1

# INTRODUCTION

## 1.1 Introduction

Optimal traffic signal control has been recognized as a very cost/effective strategy to improve the efficiency of existing transportation facilities. Synchronizing traffic signals along arterials or in a network, and optimizing the signal settings, result in smoother traffic flows, reducing idling and stopping. This, in turn, reduces fuel use, saves motorists travel time, diminishes wear and tear on vehicles, and cuts vehicular emissions (22). Furthermore, effective arterial control is an important step in developing a systematic integrated control of freeways and arterials for corridor management.

Most traffic control systems today are based on time-of-day schedules where the traffic signal settings (cycle length, green times, offsets) are set by time-of-day based on historical data on traffic demand (e.g., am peak hour turning movement counts). Some systems operate in a traffic responsive mode where the signal timing plans are selected based on observed volumes and occupancies. Traffic responsive systems find the best match between a plan that was developed based on design volume and occupancies values and observed values. There is no guarantee that a traffic responsive system will have a plan for the observed conditions. A truly traffic adaptive system will adjust the settings at traffic signals based on real-time data on traffic conditions, and can respond to unexpected or unplanned events, such as incidents, special events, weather, etc., since they adapt the timings based on observed traffic data. Similarly, adaptive systems will improve performance over time-of-day plans when the traffic patterns have a high degree of variability. Finally, adaptive systems will reduce the adverse effects of offset transition, preemption, and transit priority.

Several adaptive control algorithms have been developed in the US and overseas. However, the practical implementation of adaptive control has been limited especially in California. There is a need to develop adaptive control algorithms, evaluate their performance through a field test, and develop a deployment plan for possible Statewide application. The objective of this study is to develop adaptive signal control strategies along arterials, evaluate their effectiveness, and develop recommendations for possible Statewide deployment. Specific products of the proposed research will be:

- An adaptive control algorithm for arterial streets that explicitly addresses the California Department of Transportation (Caltrans) needs
- Field testing and evaluation of the proposed algorithm in a real-world arterial
- A deployment plan with guidelines for statewide implementation of adaptive control

This report describes the work performed in the first phase of the project, under PATH Task Order 5322. The next section summarizes the findings from the literature review on existing adaptive signal control systems. Chapter 2 presents the study methodology. Chapter 3 describes the selected test site and the data collection and processing. Chapter 4 presents the analytical procedure to develop an origin-destination matrix from detector data for input to the simulation model. Chapter 5 describes a plug-in developed for the PARAMICS model to simulate signal control strategies. Chapter 6 summarizes the study findings and outlines the work to be performed in the second phase of the project under the continuation PATH Task Order 6322.

## 1.2 Existing Adaptive Signal Control Systems

A literature review was undertaken to identify the state-of-the-art adaptive control algorithms that are operational and implemented in real-life systems. Emphasis was placed on the latest information on existing algorithms through published sources and contacts with systems' developers and users. In addition to published sources[1], the Transportation Research Board (TRB) Traffic Signal Systems Committee website includes several presentations on the characteristics of several adaptive traffic signal systems including system architecture, data requirements, communications requirements, local controller and central hardware requirements, installation operation and maintenance costs.[2]

The characteristics of several adaptive signal control systems that have been implemented in the field are summarized in the Table 1.1 below:

**Table 1.1 Summary of Existing Adaptive Signal Control Systems**

| **System** | **Installa-tions** | **Architecture** | **Detection** | **Controller** | **Communications** |
|---|---|---|---|---|---|
| SCOOT | Over 200 worldwide | Centralized | Exit loops | NEMA (EPAC) or special | Once per second for hold, force-off omit and detector data |
| SCATS | Over 50 worldwide | Hierarchical (plan based) | Stop bar loops | 2070 or special | Strategic control from central and local tactical control |
| OPAC | 2 | Decentralized | Exit loops | NEMA (with VS-PLUS firmware) and VME co-processor) | Once per cycle |
| RHODES | 4 | Decentralized | Fully actuated design | 2070 (with NextPhase firmware and VME co-processor) | Peer-to-peer over IP, event based on upsteam detections |
| ATCS (Los Angeles) | 1 | Centralized | Fully actuated with system detectors for VOS | 2070 with LADOT firmware | Once per second |
| TUC *(Chania, Greece)* | 3 | Central | System Loops for VOS | European | Once per second |
| UTOPIA *(Torino, Italy)* | 1 | Distributed | Fully actuated design | European | |

There have been many studies where different adaptive control systems have been evaluated. Most of the field evaluation studies that have been conducted on adaptive systems have concentrated on addressing the ability of these systems to provide benefits in terms of reductions in travel times, intersection delays, and the number of stopped vehicles. These field evaluations have primarily used the "before" and "after" technique. Evaluation of SCOOT deployments has reported reduction in intersection delay in the range of

[1] *Documents describing traffic control strategies and related issues are listed in the BIBLIOGRAPHY section of the report*
[2] http://www.signalsystems.org.vt.edu/documents.html

0% to 53% over the previous (primarily fixed time) control. In some of these studies the signals were first optimized before the "before" data was collected.

SCATS has been deployed in Oakland County Michigan. Reported improvements in travel times were 7%-32% during different times of the day and compared to previous conditions which consisted only time based control with little to no effort in maintaining signal timings.

The Federal Highway Administration (FHWA) sponsored Adaptive Control System (ACS) prototypes - OPAC, RHODES, and RTACL were field tested in Reston, Virginia; Seattle, Washington, and Chicago, Ill, respectively. In general, there were many lessons learned from these deployments. Issues related to installation and operation of detection systems, communications, and controllers were significant challenges in each of the field tests. RTACL performed below expectations. OPAC did not improve network conditions and in some cases delay and travel time actually increased. Simulation tests of OPAC using the CORSIM simulation model were consistent with the field test results. RHODES showed no significant difference in arterial travel times, but the cycle times were significantly reduced in Seattle. No delay study was conducted on RHODES.

In general, traffic engineers perceive traffic adaptive signal control systems to be systems that will improve traffic signal operational performance with decreased staff workload. While this perception indicates the desire of traffic engineers to provide better service to the public within the tightly constrained budgets that are available, the actual benefits and costs are still poorly understood and largely unquantified. Most of the cited benefits are based on limited data and do not relate to the geometric, traffic and control characteristics of the specific project areas (23). Other factors limiting the deployment of adaptive systems include:

- Agency capability and willingness to deploy adaptive systems
- Concerns over actual benefits, dependencies on site specific conditions, and loss of an engineer's control over the timing plans
- Lack of understanding, including complexity of the systems as well as not knowing the difference between adaptive and responsive or other off-line control operations
- Additional initial expense and maintenance costs
- Imaturity of US systems and concern about difficulties associated with using foreign developed systems.

FHWA in cooperation with signal controller manufactures recently developed the Adaptive Control Software *Lite* (ACS Lite) (20), with the goal of providing a "widely deployable" system that automates monitoring of traffic signal performance and adjustment of signal timing. The Lite designation reflects a focus on reducing traditionally high installation and operations costs, which have been the primary impediment limiting the deployment of adaptive systems in the U.S. Simulation evaluation has demonstrated significant benefits in the context of suboptimal settings, and "no harm done" in the context of signal timing that was optimized with perfect knowledge of traffic conditions. Field evaluations were then conducted in four sites showed that signal timing adjustments by ACS Lite provided substantial reductions in vehicle delay, arterial travel time, vehicle stops, and fuel consumption.

# CHAPTER 2
# METHODOLOGY

The research approach consists of theoretical development, simulation modeling, and field implementation and evaluation of the proposed strategies. The primary emphasis on this project is to produce and field test operational adaptive control algorithms. Emphasis therefore is given in the methodology to address the key issues in carrying out this project rather than describing a formulation of a proposed algorithm.

There are several issues to be addressed in developing adaptive control algorithms including but not limited to control philosophy, surveillance requirements, theoretical formulation, computational requirements, controller hardware & software requirements, communications requirements, and interfaces with other systems (e.g., freeway TMCs) for integrated corridor management or other local agencies' TMCs for multi-jurisdictional control. Key issues for the development and evaluation of adaptive control strategies are discussed in the following sections.

## 2.1 Development of Adaptive Control Strategies

The approach for developing an adaptive control algorithm is to select the most promising control strategy from the existing strategies identified in the literature and design and implement new functions and features that are required to meet the needs and requirements of the selected test sites that are representative of major arterials (including freeway-arterial corridors). Some key considerations include:

**Control philosophy:** The common assumption is that adaptive control continually changes the signal settings to match the measured (or predicted) traffic patterns in the network. However, this approach may not be beneficial on congested conditions because it may create spillbacks upstream from the critical intersection(s). Furthermore, coordination with adjacent metered ramps or congested off-ramps may necessitate overriding the blind adaptation to traffic patterns at the arterial. It is critical to understand the difference between "control" and "adapt" and design an algorithm that satisfies the objectives and constraints for the specific project area.

**Compatibility with Caltrans & local agencies signal control practices:** The development of improved control algorithms should take into consideration the existing operation of signal systems on California State highways and local major arterials. Most of the signals are equipped with 170 or 2070 signal controllers and stopline detection. Algorithms that are designed to operate with such hardware framework would be easier to implement than algorithms which require customized controllers or detection location (i.e., not compatible with the California MUTCD guidelines, for example Table 4D-101).

**Surveillance requirements**: The effectiveness of adaptive control depends on the availability and accuracy of the data on traffic conditions, i.e., size, shape and speed of traffic platoons approaching the intersection. The cost of installation and maintenance of conventional surveillance systems (loop detectors) is very high, particularly for systems that require multiple detection points upstream, which make the widespread deployment of such systems very difficult. We will investigate the potential of new detector technologies and how they can work with the adaptive algorithms.

## 2.2 Evaluation of the Proposed Adaptive Control Strategies

The objective of the evaluation is to quantify the benefits of the adaptive control. It is important to differentiate between the benefits of the proposed control (with new hardware and software) against both a) the existing system with existing signal settings and b) existing system with optimal signal settings to estimate the true benefits of adaptive control (24). Figure 2.1 illustrates the proposed methodology.

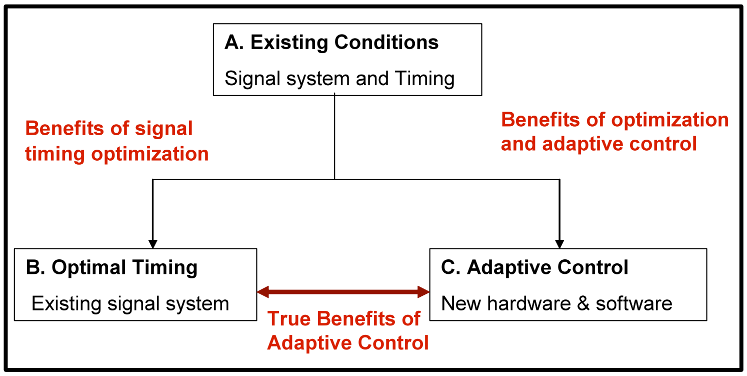


**Figure 2.1 Measuring the Benefits of Adaptive Signal Control**

- Comparison of (a) vs. (b) provides the benefits due to the optimal operation of the system with existing hardware and control strategy (e.g., TOD plans)
- Comparison of (a) vs. (c) provides the total benefits due to adaptive control from existing conditions
- Comparison of (b) vs. (c) provides the true benefits of adaptive control

The measures of effectiveness (MOEs) for the evaluation of the alternative strategies include travel times, delays and number of stops on arterial links and cross-streets. System measures may include VMT (veh-miles traveled), VHT (veh-hrs traveled), and cycle failures. Energy and environmental measures may include excess fuel consumption, and vehicle emissions (CO, HC and NOx). These environmental measures are usually derived from the primary performance measures of travel times, delays and stops.

It is also important to consider the traffic impacts of the proposed strategy(ies) on each system component. For example, it is well known that signal coordination improves traffic performance on the arterial through links at the expense of cross-streets. Therefore, the evaluation plan will analyze the performance measures separately on

- arterial through links
    - travel time (speed)
    - delay
    - number of stops (% vehicles stopped)
- cross-streets
    - queue length
    - delay
- total system
    - VMT
    - VHT
    - cycle failures
    - fuel consumption
    - emissions

## 2.3 Simulation Modeling

The performance of the existing and proposed control strategies will be first evaluated through simulation. Simulation provides several attractive attributes including the ability to evaluate system performance under a wide variety of traffic conditions. Simulation allows to vary many system characteristics including volumes, turning proportions, saturation flow rates (such as might change due to weather conditions), and events such as lane closures, preemptions, etc. However, it is important to realize that simulation generally does not support testing of operational issues such as detector errors. This can be approximated by testing the performance of the algorithm with less than ideal detector data, but still we cannot analyze the effects of *random* data losses. There are several important considerations in the evaluation of control strategies through simulation:

**Model selection**

A number of simulation tools are available to model signal operations on arterials and networks. Existing models can be classified into macroscopic and microscopic. Macroscopic models consider the average rates of flow in the network and use analytical relationships to model traffic flow. Microscopic models in contrast, simulate the movement and interactions of individual vehicles. Microsimulation has been selected for this project because the signal control strategies being considered often react to single vehicle events. For example, actuated signal operation monitors the advance detectors and may extend the green time if a vehicle is registered by an active approach detector. Such effects are difficult to reproduce in macroscopic simulation.

Among the various microscopic models available, PARAMICS was selected as the main tool for modeling control strategies because it has been used in a number of Caltrans sponsored projects, and of the research team's familiarity with its Application Programming Interface (API). The CORSIM model (10) also will be used in the study because it can directly model the RHODES and ACS Lite control strategies.

PARAMICS is a suite of simulation tools developed by Quadstone (19). PARAMICS Modeller is the core network building tool. It enables the user to display a map of the selected test site and overlay and edit the nodes and links that constitute the network model. All of the supply side elements of the network, including traffic signals, lanes, priority and permission rules, are defined in Modeller. PARAMICS Processor is a batch execution tool. It allows the user to define a sequence of simulation runs and execute them at a much faster speed than with Modeller. PARAMICS Programmer is the API; it consists of a set of C-based functions that provide access to many of the internal variables of the PARAMICS model, and it is used to code signal control algorithms.

**Calibration of baseline conditions**

The first step in the simulation modeling of strategies is to calibrate the simulation model against field measurements to ensure that the simulation model replicates field conditions. Model calibration is a complex process given the numerous parameters in the microscopic simulation models. The model calibration will be carried out according to the guidelines developed by FHWA (8), which consist of the following three steps: a) calibration for capacity, b) route choice calibration, and c) overall system performance calibration.

**Coding of the signal control strategies**

The existing and proposed signal control algorithms will be coded in the simulation model. This is generally a complex process because it requires developing an interface through hardware in the loop or a plug-in between the simulator and the control software. Chapter 5 of the report describes the plug-in developed in the project to simulate a number of control strategies utilizing the PARAMICS model API.

**Comparison of alternatives**

This process involves comparisons of the model predicted MOEs under each control scenario to determine if statistically significant improvements are obtained. This in turn requires that several replications of the simulation need to be performed for each scenario and the results statistically analyzed. The number of replications depends on the conditions to be simulated and the performance measures to be analyzed. For example, the stochastic variability in the model results is much higher for link delays under heavy traffic volumes as opposed to system travel times under low volume conditions.

### 2.4 Field Testing

Field testing of the proposed algorithm on the selected sites involves the following steps;

a) "Before" data collection on the selected performance measures
b) Field implementation of the proposed algorithm
c) "After" data collection on the selected performance measures

The methodology for field data collection will depend on the characteristics of the test site and the surveillance system capabilities. For example on systems equipped with cameras at the intersection, we can process the image to obtain estimates of delays and queue lengths through image processing. Floating cars equipped with GPS units may be also utilized to obtain estimates of arterial link travel times, delays and stops. The duration of the field tests will provide sufficient data to determine if statistically significant improvements have been obtained. At a minimum one week of "before" and "after" field data will be collected. This estimate will be revised as appropriate based on the sample measurements of the MOEs. We will carefully monitor the traffic and other operating conditions throughout the field experiment to ensure that the measurements are not masked by external factors.

We will also compare the field results against the simulation results. Comparison of simulation and field test results are often difficult and misleading since simulation studies are generally conducted under well controlled conditions where performance measures are not collected until the system has reach a "steady state". It is not possible to fully control field test conditions so the performance measures may be collected during periods of transition or when random events may be occurring on the network. However, field testing is the only true test of performance, but it should be recognized that field testing may not always produce findings that truly measure the capabilities of an adaptive system.

# CHAPTER 3

# THE TEST SITE

## 3.1 Test Site Selection

This research will field test proposed adaptive control algorithms on a real-world arterial. It is therefore important that suitable test sites are selected for the evaluation of the proposed algorithms. Criteria for test site selection include:

- Traffic volumes and patterns: The selected site should have both heavy traffic volumes and highly variable traffic patterns. There are no significant benefits from adaptive control if the arterial volumes follow a predictable traffic pattern throughout the day. Furthermore, if the traffic volumes are well below capacity then the variability in traffic patterns can be accommodated with conventional fixed-time or actuated control.
- Proximity to freeway: Location of the test arterial close to freeway that is part of a freeway-arterial corridor will permit development and testing of algorithms for integrated corridor control
- Existing signal system capabilities: signal hardware and software and communications that allow central or decentralized monitoring of the signal system and remote implementation of changes in signal settings.
- Data availability: Existing data on intersection geometrics, traffic volumes and patterns, and existing signal settings are required for the development and evaluation of the proposed algorithms through simulation. Also, existing data on traffic performance including travel times, delays and queue lengths
- Cooperation with Caltrans & local agency staff: Staff in charge of the system operation and maintenance willing to participate in the study, review the proposed algorithms and provide cooperation and support throughout the field test

The following sites were investigated based on discussions with Caltrans and local agencies' operations staff:

1) San Pablo Avenue (State Highway 123): San Pablo is located in the San Francisco bay Area and it is a parallel route to I-80 freeway. It is part of the East Bay Smart Corridor.
2) State Highway 17 and Bascom Avenue: This is part of the Silicon Valley Smart Corridor in San Jose.
3) Orange County Test Bed: There are several freeway-arterial corridors in the cities of Irvine and Anaheim
4) Pacific Coast Highway (PCH) in Southern California

The test site selected for the study is a stretch of the PCH running through the city of Lomita in Southern California (Figure 3.1). This stretch runs East-West, is relatively straight and flat, and contains 11 almost evenly spaced intersections, 7 of which are signalized. All of the signalized intersections are operated by 2070 controllers, most of them have protected left turns. The key considerations in choosing this site were its hardware (2070 controllers coordinated by a field master), variability of traffic volumes, and availability of data. The test site is equipped with approach and stopline detectors at every signalized intersection. The measurements from these detectors can be remotely accessed via the Caltrans CTNET system.

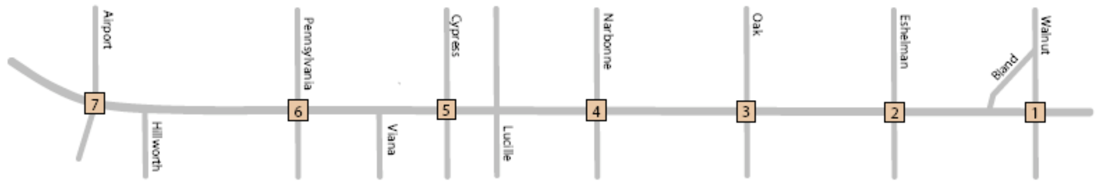


**Figure 3.1 The Selected Test Site (Pacific Coast Highway, Lomita, CA)**

## 3.2 Data Collection and Processing

Information on intersection geometrics, limited manual turning movement counts and signal settings (controller cards) were made available by Mr. Jorge Fuentes of Caltrans District 7. The research team made two field visits to the site to observe operating conditions and to gather supplemental data.

Traffic demands are defined in the PARAMICS model with a time-varying origin-destination (OD) table. To obtain an OD table for the study section, first we obtain detector data using CTNET and then we developed a procedure to estimate the OD table from the detector data (described in Chapter 4).

CTNET (2) is a map-based software developed by Caltrans. It is intended to facilitate the remote management, monitoring, and analysis of signal status and traffic sensor data at signalized intersections. CTNET uses Windows sockets and is based on the server/client paradigm. A server application resides on a computer in the Caltrans offices. It relays information from the field master controller to the various client programs requesting that information. The client side program receives signal and detector information once per second and displays it on a map. The transmitted information includes:

- Cabinet alarm state (preemption and flash).
- Phase (reds, yellows, greens, and PEDs).
- Phase calls (vehicle and PEDs).
- Detector presence.
- Volume and occupancy data (see Figure 3.2).

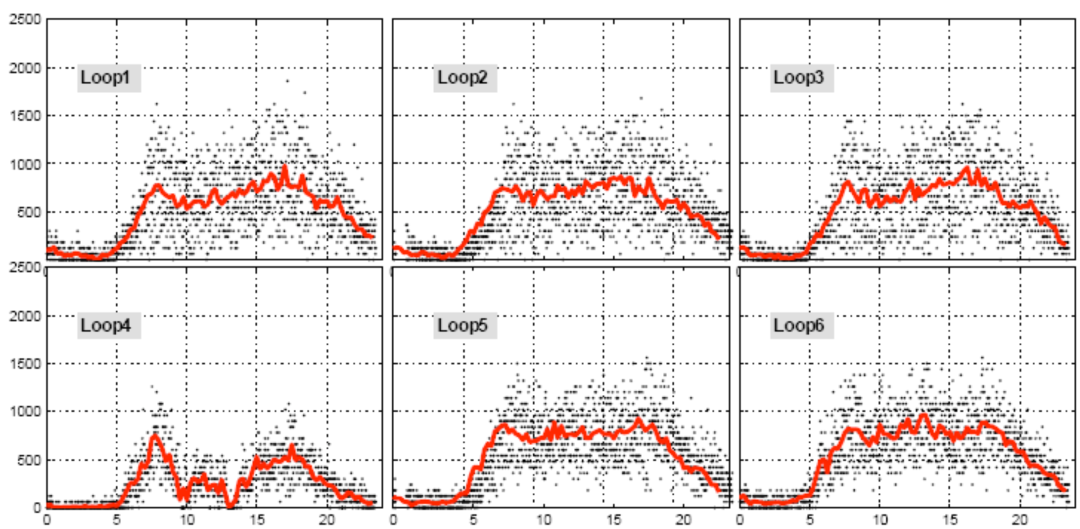


**Figure 3.2 Test Site: Raw and Filtered Traffic Volumes**

CTNET was used to collect 70 weekdays of data between March 2$^{nd}$ and June 29$^{th}$ 2006. The volume and occupancy data were saved in a Microsoft Access database. Each day contained 15-minute flows for each of the 37 mid-block detectors. These Access tables were converted to Excel spreadsheet form with the aid of the *AutoIt* scripting tool (12), a free software that can be used to automate a number of repetitive tasks.

The Excel tables were then loaded into the *Matlab* software and ranked according to the total flow accumulated over 24 hours. Several days were seen to have anomalously low total flows, due presumably to incidents or faulty detectors. About one third of the days were discarded on the basis of low volumes or too few samples (See Figure 3.3). From the remaining two thirds we selected the median day for the volumes to input into the PARAMICS model to simulate existing conditions (See Figure 3.4).

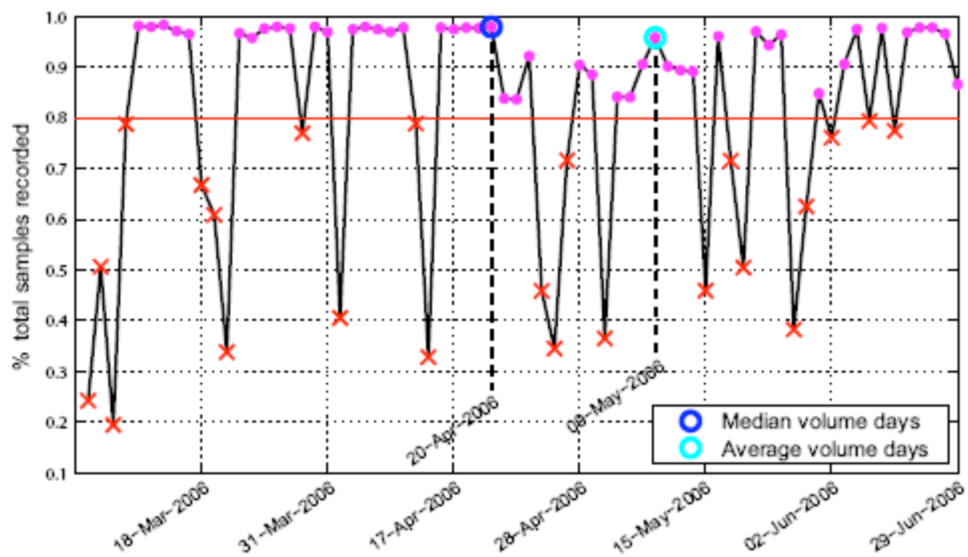


**Figure 3.3 Test Site: Data Samples**

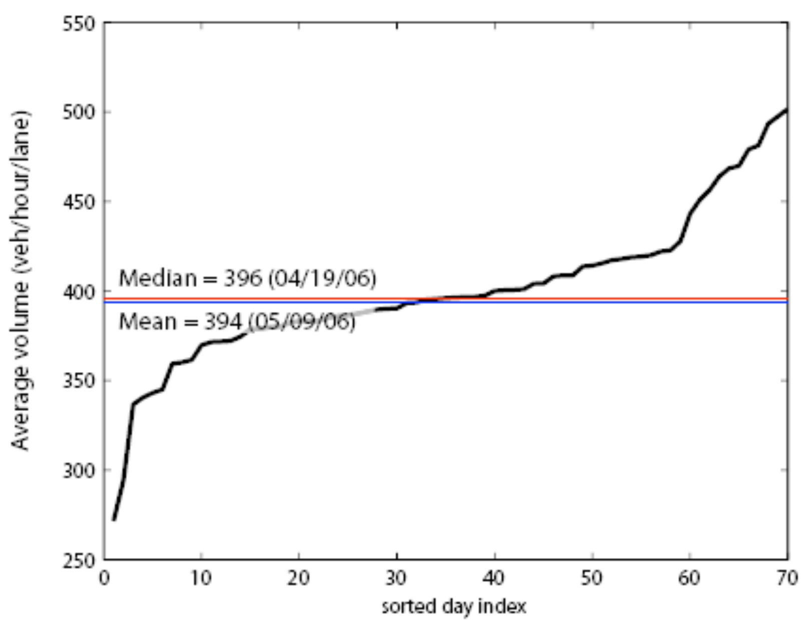


**Figure 3.4 Distribution of Average Volumes**

# CHAPTER 4

# ORIGIN-DESTINATION MATRIX ESTIMATION

Traffic demands are defined in the PARAMICS simulation model with a time varying origin-destination (OD) matrix. This Chapter describes the procedure developed in this study to obtain the OD matrix from detector measurements at the study site[3].

## 4.1 Introduction

Most simulation models, both macroscopic and microscopic, require an OD matrix to define the traffic demand patterns on the study network. However, few simulation software include a procedure for estimating it. The traffic engineer in need of an OD matrix is often confronted with the decision of either using a simulation model that provides OD estimation, such as PARAMICS or implementing one of the algorithms found in the academic literature. Regarding the latter option there is a large number of techniques from which to choose, with varying degrees of sophistication and computational requirements. Most of these employ mathematical programming methods to find the OD matrix. The problem in most cases can be expressed as follows:

> *minimize J(OD, additional data)*
> *such that the measured flows are reproduced, and individual OD flows are non-negative*

The additional data in the objective function is often a target OD matrix, assumed available from previous studies. The problem thus formulated is to find the matrix with non-negative elements which is closest to the target matrix, while reproducing the measurements. Several distance functions $J$ have been considered, including maximum entropy (29), quadratic functions (3), and absolute value norms (21). The method of generalized least squares, applied to OD estimation by Cascetta in (3), has received particular attention due to its versatility and numerical robustness. Bell (1) later improved the method by enforcing non-negativity constraints. Another related line of research has developed algorithms which take explicit account of the stochastic nature of the problem (16, 17, 26).

The approaches mentioned thus far assume that the routing of traffic is known beforehand from a proportional assignment matrix. However several authors have noted that the two problems, OD estimation and traffic assignment, may be interrelated when the network is congested. In this situation, a user equilibrium assumption on the distribution of traffic is more appropriate (28,30).

A more recent development has been the use of time-series data for the estimation of either static or time-varying OD matrices. Cremer and Keller (6) first used sequences of traffic counts to estimate OD information for a single complex intersection. Nihan and Davis (18) then generalized the approach by noting that it falls within a larger family of parameter estimation techniques. Li and De Moor (14) improved upon the accuracy of (31) and (25). These works are of relevance to the present approach because they apply only to a strictly defined network configuration. An important improvement was provided by Li and De Moor in (15) when they allowed for some of the boundary flows to be unmeasured. Another interesting and promising line of research has been the use of automatic vehicle identification techniques for gathering partial vehicle trajectories (7, 13,31).

Aside from the academic approaches, the other alternative is to use a simulation model such as PARAMICS which includes an OD estimation module. The network coding effort notwithstanding, this

---

[3] *This Chapter was originally published in "A simple procedure for estimating origin-destination matrices for arterial corridors", by Gomez and Skabardonis, paper 08-1750, presented at the 87th Annual Meeting of the Transportation Research Board, Washington DC, January 2008.*

approach is attractive because it allows the user to observe the resulting dynamic response of the network. These observations can then be used to adjust the parameters of the estimation algorithm, in an iterative process that may be manual or automated. The general approach adopted by the PARAMICS OD estimator combines an optimization based OD estimator with the microscopic network simulator, which includes a dynamic traffic assignment procedure. At every iteration, the user is allowed to adjust the *confidence weights* assigned to each detector measurement (link or turning flows). These weights in turn influence the distance from the measured values that the estimated values may take.

There is in principle no reason why any of the estimation techniques mentioned above could not be included in an iterative procedure. However it would be difficult for a model user to adjust the parameters of the estimator, since the relation between the inputs and the dynamic behavior is usually complicated. For example, it may not be clear how the entries in the target OD matrix should be adjusted in order to, say, shorten the queue in a particular left-turn pocket. In PARAMICS on the other hand, the user is allowed to adjust not only the target OD matrix, but also target turning and link flows. To shorten the queue, the user would simply increase the target turning flow exiting the queue without increasing the entering target flow. More generally, any adjustment to an aggregate quantity can be achieved through adjustments to the *turning flows*.

### 4.2 Proposed O-D Estimation Procedure

We present a mathematical program with a complete set of consistent turning flows, which are computed previously from the measurements. Because these flows are consistent, they can be imposed strictly upon the OD matrix search. Moreover the target OD matrix, which has traditionally been assumed available from surveys, is no longer needed. To compute the turning flows we describe a framework consisting of several steps. In the first step, the mid-block detector measurements are extended to the intersections. Then the boundary flows on each intersection are used to compute the total number of vehicles entering and exiting the main arterial in either direction. Third, the turning flows are computed from these values, and finally the OD matrix is found with an optimal search. This step-by-step approach is intended to split the larger problem into manageable parts, and thus allow for several levels of verification of the results.

We focus on a specific network topology, consisting of a single two-way arterial, intersected by a number of cross streets. The intersections may have three or four legs, as illustrated in Figure 4.1, and they may or may not be signalized. We will assume that at least some of the intersections (usually the signalized ones) are equipped with approach detectors located some distance upstream of the intersection, on both sides of the arterial. The measurements from these detectors constitute the primary source of data. Any additional counts or measurements, such as cross-street flows or queue length estimates, are considered as supplemental information that can be used to tune the estimation parameters. As a matter of notation, we will assume that the arterial runs in the East-West direction, while the cross streets run North-South. The method is static, so the time index is omitted. The flow measurements are assumed aggregated over a sufficiently long period of time for the static assumption to hold.

The proposed procedure has four consecutive stages. Each stage is described in terms of its input and output data, and equations are suggested to relate the two. However, these equations are not expected to apply to all situations. They are one alternative derived from a set of assumptions. Another set of assumptions, perhaps better suited for a particular scenario, may lead to better equations. Our primary goal is to demonstrate a practical division of the OD estimation problem into smaller parts, where each part maybe solved either by making some assumptions or with additional information.

### 4.2.1 Stage 1: Data completion and extrapolation

This first stage gathers all of the data preprocessing steps. These steps may include data selection, data completion, and aggregation over lanes. The term ‘data selection' refers to the often difficult task of choosing a data set with which to start the OD estimation procedure. This data set will preferably consist of measurements gathered on a single `normal' day, but it may often be necessary to work with a composite data set, constructed from many different days. The selected data set may contain gaps in time, caused perhaps by brief communication failures, or space, due to missing or malfunctioning loops. These can be repaired with data reconstruction techniques such as those described in (4).

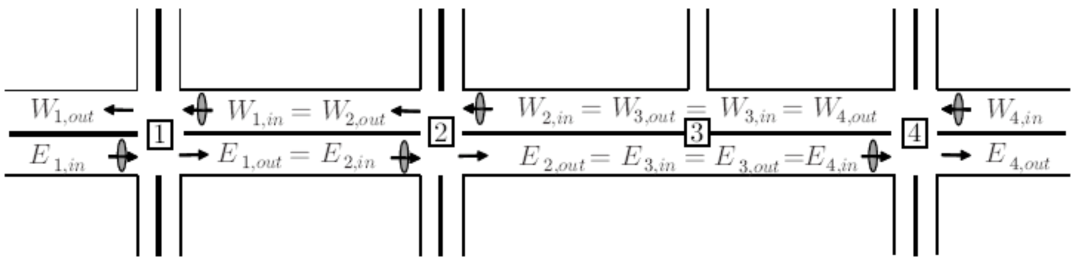


**Figure 4.1 Measured Arterial Flows**

The input to stage 2 is the flow entering and leaving each intersection along the arterial. The flows entering the arterial maybe measured, but the leaving flows are usually not. These maybe assumed equal to the flow entering the next downstream intersection as long as there is no intermediate source or sink, such as an unmeasured intersection or a large parking lot. This is illustrated in Figure 4.1, where $W_{2,out}=W_{1,in}$ and $E_{1,out}=E_{2,in}$. In the case that there is a significant source or sink, any supplemental information about its intensity should be considered in the computation of the exiting flows. Lacking additional information, the intervening unmeasured traffic sources must be considered as making no net contribution to the flow on the arterial. This is also shown in Figure 4.1 where $W_{4,out}$, $W_{3,in}$, and $W_{3,out}$ are all set equal to $W_{2,in}$.

### 4.2.2 Stage 2: Joining and leaving flows

Next we estimate the flow entering and exiting the arterial at each intersection, in each direction. Each intersection i is assigned an intersection weight $n_i$ with larger intersection weights corresponding to larger cross street flows. The ends of the arterial are also assigned weights $n_0$ and $n_{N+1}$, where $N$ is the number of intersections in the network (Figure 4.2).

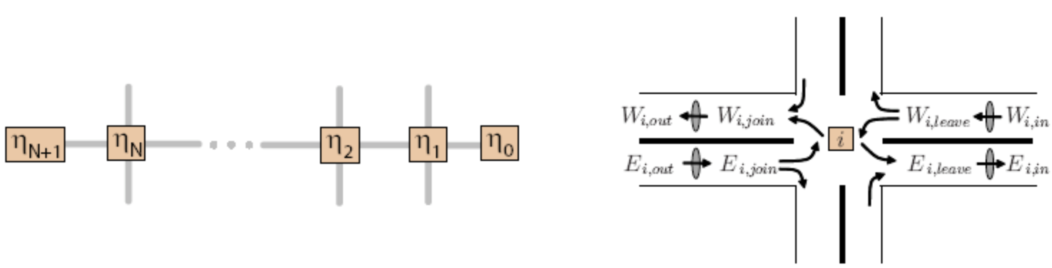


**Figure 4.2 Numbering of Intersection Weights--Joining and Leaving Flows**

The likelihood that a vehicle traveling along the arterial will turn at intersection i is defined as the ratio of the flow entering the intersection to the right and left turning flow. These likelihoods, computed for each intersection in the westbound and eastbound directions, are:

$$\alpha_{i,W} = \frac{W_{i,leave}}{W_{i,in}} \quad , \quad \alpha_{i,E} = \frac{E_{i,leave}}{E_{i,in}} \tag{1}$$

The notation is defined in Figure 5.2. We estimate the likelihoods $\alpha_{i,w}$ and $\alpha_{i,E}$ by taking t f n*i* to the sum of all downstream weights (i.e. all possible destinations).

$$\alpha_{i,W} = \max\left\{ \frac{\eta_i}{\sum_{j=i}^{N+1} \eta_j} \; , \; \frac{W_{i,in}}{W_{i,in} - W_{i,out}} \right\} \tag{2}$$

$$\alpha_{i,E} = \max\left\{ \frac{\eta_i}{\sum_{j=0}^{i} \eta_j} \; , \; \frac{E_{i,in}}{E_{i,in} - E_{i,out}} \right\} \tag{3}$$

The second terms in the max {} are needed to ensure positivity of the flows *Wi,join* a n. The intersection weight *ηi* can be interpreted as a measure of the *attractiveness* of the i n. The proportion of vehicles that turn equals the attractiveness of that intersection relative to tl veness of all downstream destinations.

The equations in the remainder of the paper apply equally to all intersections, so w nit the intersection index in the presentation. Given the measured entering and leaving flows, and (5) express the conservation of vehicles on the arterial.

$$W_{in} - W_{leave} + W_{join} - W_{out} = 0 \tag{4}$$

$$E_{in} - E_{leave} + E_{join} - E_{out} = 0 \tag{5}$$

Combining the above, the leaving and joining flows are given by:

$$W_{leave} = \alpha_W W_{in} \tag{6}$$

$$E_{leave} = \alpha_E E_{in} \tag{7}$$

$$W_{join} = W_{out} - (1 - \alpha_W) W_{in} \tag{8}$$

$$E_{join} = E_{out} - (1 - \alpha_E) E_{in} \tag{9}$$

The task of this stage is to select the intersection weights $\eta_i$. The total number of parameters to tune is $N$+2, where as the number of degrees of freedom is 2$N$ ($W_{leave}$ and $E_{leave}$ for each intersection). With the given assumptions we have therefore reduced the size of the problem from 2N variables per time slice to N+2 constant (but potentially time dependent) weights. This was achieved by imposing a strict relationship between the eastbound and westbound turning likelihoods. It should be noted that the estimated turning rates at each intersection depend on the weights of other intersections. Thus, the tuning of one intersection weight is not independent of the others. Also, the assumption of intersection weights, although very flexible, may not apply to some intersections. That is, their westbound and eastbound turning rates may not be related in the manner implied by Eqs. (2) and (3). In this case it is possible to assign independent weights to the eastbound and westbound directions.

The selection of intersection weights may be based on vehicle counts on the cross street, or, if a simulation model is available, on cross street queue lengths. Lacking any additional measurements from the cross streets, a simple classification into groups such as `small', `medium', and `large', with corresponding weights of, for example,1, 2, and 3, maybe suitable. This is demonstrated in the example of Section 4.3.

### 4.2.3 Stage 3: Turning movements

Once the overall level of cross street flow has been set, the next task is to fix the turning ratios. This is done by disaggregating the joining and leaving flows into north, south, east, and westbound portions. The notation is shown in Figure 4.3. The four left turning flows are directly tuned with four turning coefficients:

$$NW = \varphi_n W_{join} \quad (10)$$

$$SE = \varphi_s E_{join} \quad (11)$$

$$WS = \varphi_w W_{leave} \quad (12)$$

$$EN = \varphi_e E_{leave} \quad (13)$$

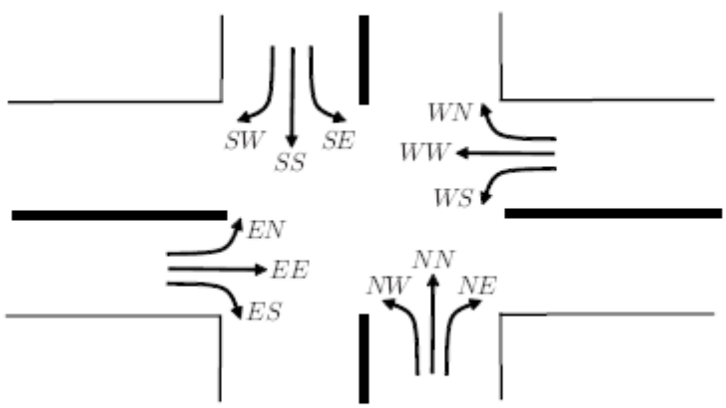


**Figure 4.3 Turning Flows**

Where $\varphi_n$, $\varphi_s$, $\varphi_w$, and $\varphi_e$ range between 0 and 1. Lacking any additional information, all of these parameters should be set to the default value of 0.5. The right-turning flows are then computed directly from flow conservation equations:

$$SW = W_{join} - NW = (1 - \varphi_n)\, W_{join} \quad (14)$$

$$NE = E_{join} - SE = (1 - \varphi_s)\, E_{join} \quad (15)$$

$$WN = W_{leave} - WS = (1 - \varphi_w)\, W_{leave} \quad (16)$$

$$ES = E_{leave} - EN = (1 - \varphi_e)\, E_{leave} \quad (17)$$

Two additional parameters, $\gamma_n$ and $\gamma_s$, are used to set the level of continuing flow on the cross streets:

$$\mathrm{NN} = \gamma_n(\mathrm{NE}+\mathrm{NW}) = \gamma_n(1-\phi_s)E_{join} + \gamma_n\phi_n W_{join} \tag{18}$$

$$\mathrm{SS} = \gamma_s(\mathrm{SE}+\mathrm{SW}) = \gamma_s(1-\phi_n)W_{join} + \gamma_s\phi_s E_{join} \tag{19}$$

Finally, continuing flows on the arterial are determined by $\alpha_W$ and $\alpha_E$, which were fixed ir

$$\mathrm{WW} = W_{in} - W_{leave} = (1-\alpha_W)W_{leave} \tag{20}$$

$$\mathrm{EE} = E_{in} - E_{leave} = (1-\alpha_E)E_{leave} \tag{21}$$

The equations so far described can be gathered into a single matrix equation for each inte

$$M = AU \tag{22}$$

where $U$=[$Win,Wout,Ein,Eout$]$^T$ , $M$=[WN,WW,WS,SW,SS,SE,EN,EE,ES,NE,NN,NW]$^T$

$$A = \begin{bmatrix} \alpha_W(1-\phi_w) & 0 & 0 & 0 \\ \alpha_W(1-\alpha_W) & 0 & 0 & 0 \\ \alpha_W\,\phi_w & 0 & 0 & 0 \\ (1-\phi_n)(\alpha_W-1) & 1-\phi_n & 0 & 0 \\ \gamma_s(1-\phi_n)(\alpha_W-1) & \gamma_s(1-\phi_n) & \gamma_s\phi_s(\alpha_E-1) & \gamma_s\phi_s \\ 0 & 0 & \phi_s(\alpha_E-1) & \phi_s \\ 0 & 0 & \phi_e\alpha_E & 0 \\ 0 & 0 & (1-\alpha_E)\alpha_E & 0 \\ 0 & 0 & (1-\phi_e)\alpha_E & 0 \\ 0 & 0 & (1-\phi_s)(\alpha_E-1) & 1-\phi_s \\ \gamma_n\phi_n(\alpha_W-1) & \gamma_n\phi_n & \gamma_n(1-\phi_s)(\alpha_E-1) & \gamma_n(1-\phi_s) \\ \phi_n(\alpha_W-1) & \phi_n & 0 & 0 \end{bmatrix} \tag{23}$$

We have defined seven tunable parameters for each intersection ($\eta$, $\varphi_n$, $\varphi_s$, $\varphi_w$, $\varphi_e$, $\gamma_n$, se were assumed constant, since the general approach is static. However, different values of the ers may be used for the morning versus the evening period, or even for every time slice. The nume mple of Section 4.3 describes an $\eta$ that changes every 15 minutes.

The turning coefficients for three-legged (T) intersections are fixed to 0 or 1, according to Table 4.1. The only tunable parameter for these intersections is the intersection weight.

**Table 4.1 Fixed Parameters for T Intersections**

| Configuration | $\phi_n$ | $\phi_s$ | $\phi_w$ | $\phi_e$ | $\gamma_n$ | $\gamma_s$ |
|---|---|---|---|---|---|---|
| ⊤ | 1 | 0 | 1 | 0 | 0 | 0 |
| ⊥ | 0 | 1 | 0 | 1 | 0 | 0 |

#### 4.2.4 Stage 4: OD matrix computation

The OD matrix is found by solving an optimization problem. Each turning flow calculated in the previous stage (12 for every 4-branch intersection and 6 for every 3-branch intersection) enters the problem in a separate equality constraint. Hence the total number of equality constraints is at most 12I, where *I* is the number of intersections, whereas the number of OD pairs is (2I+2)(2I+1). The problem is therefore underspecified for all values of *I*>1.

It is important to note that the aggregate behavior of the network has been completely fixed by prescribing the turning flows. That is, all OD matrices in the feasible set have the same aggregate behavior, since the flows entering and exiting every link (and therefore the aggregate link densities) are the same. Any of the feasible OD matrices will therefore suffice, as long as we are only interested in matching aggregate quantities such as the total travel time.

In this context the choice of objective function is not very important. Some popular alternatives found in the literature include maximum entropy, minimum additional information, and least squares. In the numerical example we minimize the variance among the OD flows, defined in Eq.(24).

The optimization problem is stated in Eqs. (24)-(27). Barring U-turns, each OD pair has only a single connecting route. We can therefore define for each intersection m ∈ [1...*I* ] and turning flow *n*∈ [1...12] a set *Umn* of nodes that are upstream and a set *Dmn* of nodes that are downstream of that flow. The sum of all OD pairs *Tij* where i ∈ *Umn* and *j* ∈ *Dmn* must then equal the turning flow *fmn* calculated in the third stage (Eq.(26)). Also, each of the OD flows must be non-negative (Eq. (27)).

$$\text{minimize:} \quad \sum_{ij \text{ pairs}} (T_{ij} - \bar{T})^2 \tag{24}$$

$$\text{subject to:} \quad \bar{T} = \frac{1}{M} \sum_{ij \text{ pairs}} T_{ij} \tag{25}$$

$$\sum_{\substack{i \in U_{mn} \\ j \in D_{mn}}} T_{ij} = f_{mn} \qquad \forall\ mn \text{ pairs} \tag{26}$$

$$T_{ij} \geq 0 \qquad \forall\ ij \text{ pairs} \tag{27}$$

### 4.3 Application on the Selected Test Site

The OD estimation procedure was tested on the selected study section, a stretch of the Pacific Coast Highway in Lomita, California (Figure 3.1). This site includes a total of 11 intersections, seven of which are signalized (Table 4.2). Narbonne Avenue is the only cross street equipped with approach detectors.

Seventy week-days of loop detector data between March and June of 2006 were gathered. Each day contained 15-minute flows for each of the 37 mid-block detectors. The days were ranked according to the total flow accumulated over 24 hours. Many of the days were seen to have anomalously low total flows, due presumably to accidents or faulty detectors. About one third of the days were discarded on this basis. From the remaining two thirds we selected the median day (April 19th, 2006) for the experiment. In addition to the loop data, we made two field visits to gather supplemental information. During these visits we observed traffic at each intersection for about 20 minutes during the afternoon peak period. We recorded approximate queue lengths on the arterial and cross streets, and made visual estimates of the turning coefficients described in Section 4.2.2. These observed turning coefficients provided a starting point for tuning the estimator.

The estimation algorithm was coded in the *Matlab* software and a microscopic model of the site was constructed in PARAMICS. The PARAMICS model was used to test candidate OD matrices and compare the simulated queue lengths with the field observations. Traffic signals control strategies in PARAMICS were simulated using the plug-in described in Chapter 5 of the report.

**Table 4.2: Cross Streets and Tuned Parameter Values**

| Street Name | Index | Branches | Signalized | $\eta$ | $\phi_n$ | $\phi_s$ | $\phi_w$ | $\phi_e$ | $\gamma_n$ | $\gamma_s$ |
|---|---|---|---|---|---|---|---|---|---|---|
| Walnut | 1 | 4 | ✓ | 2.0 | 0.4 | 0.7 | 0.5 | 0.4 | 0.1 | 0.1 |
| Bland | 2 | 3 | × | 0.2 | 0.0 | 1.0 | 0.0 | 1.0 | 0.0 | 0.0 |
| Eshelman | 3 | 4 | ✓ | 2.0 | 0.5 | 0.5 | 0.4 | 0.4 | 0.1 | 0.1 |
| Oak | 4 | 4 | ✓ | 1.0 | 0.5 | 0.5 | 0.5 | 0.5 | 0.0 | 0.0 |
| Narbonne | 5 | 4 | ✓ | $\eta(k)$ | 0.5 | 0.5 | 0.2 | 0.2 | 0.5 | 0.5 |
| Lucille | 6 | 4 | × | 0.2 | 0.5 | 0.5 | 0.5 | 0.5 | 0.0 | 0.0 |
| Cypress | 7 | 4 | ✓ | 1.0 | 0.5 | 0.5 | 0.5 | 0.5 | 0.0 | 0.0 |
| Viana | 8 | 3 | × | 0.2 | 0.0 | 1.0 | 0.0 | 1.0 | 0.0 | 0.0 |
| Pennsylvania | 9 | 4 | ✓ | 1.0 | 0.5 | 0.5 | 0.2 | 0.2 | 0.0 | 0.0 |
| Hillworth | 10 | 3 | × | 0.2 | 0.0 | 1.0 | 0.0 | 1.0 | 0.0 | 0.0 |
| Airport | 11 | 4 | ✓ | 1.0 | 0.5 | 0.5 | 0.5 | 0.5 | 0.0 | 0.0 |

Table 4.2 lists the tuned parameter values. The weights for the ends of the arterial were set to $\eta_0=\eta_{12}=6$. Of the 61 tunable parameters (8 four-branch intersections + 3 three-branch intersection + 2 end weights), 32 were left at their default value, 3 of the parameters for Narbonne Ave. were calculated, while the remaining 26 were tuned by trial-and-error, with the aid of the microscopic simulator. Observing the progress of the microscopic model we were able to identify intersections where excessive queues formed and adjust the relevant turning flows without affecting the rest of the simulation. This is generally not possible with OD estimation techniques that do not directly tune the turning flows.

Narbonne Ave. (Index=5 in Table 4.2) was the only intersection where additional measurements on the cross street were available. The total northbound and southbound flows approaching the intersection on Narbonne during time interval $k$ are denoted $\widehat{N}(k)$ and $\hat{S}(k)$. Using these measurements we were able to explicitly compute a time varying intersection weight ($\eta(k)$) and joining parameters ($\varphi_n$ and $\varphi_s$) such that the computed flows matched the measured flows:

$$\mathrm{NN}(k) + \mathrm{NE}(k) + \mathrm{NW}(k) \;=\; \hat{N}(k) \qquad (28)$$

$$\mathrm{SS}(k) \;+\; \mathrm{SE}(k) \;+\; \mathrm{SW}(k) \;=\; \hat{S}(k) \qquad (29)$$

Replacing the terms on the left with their expressions from Sections 5.1.2 and 5.1.3, xing $\gamma_n = \gamma_s \triangleq \gamma$

we find by adding Eqs.(28) and Eqs.(29) that $\eta(k)$ is a root of the following quadratic pol

$$A(k)\,\eta(k)^2 \;+\; B(k)\,\eta(k) \;+\; C(k) \;=\; 0 \qquad (30)$$

where:

$$A(k) \;=\; W_{in}(k) + E_{in}(k) - F(k) \tag{31}$$

$$B(k) \;=\; W_{in}(k)\,\eta_E \;+\; E_{in}(k)\,\eta_W \;-\; F(k)(\eta_W + \eta_E) \tag{32}$$

$$C(k) \;=\; -F(k)\,\eta_W\,\eta_E \tag{33}$$

$$F(k) \;=\; \frac{\hat{N}(k) + \hat{S}(k)}{1+\gamma} + W_{in}(k) - W_{out}(k) + E_{in}(k) - E_{out}(k) \tag{34}$$

$$\eta_E = \sum_{j=0}^{4} \eta_j = 11.4 \quad , \quad \eta_W = \sum_{j=6}^{12} \eta_j = 9.6$$

The resulting time-varying intersection weight is shown in Figure 4.4. By requiring that the E nd Eqs. (29) hold separately, we find the following expression for $\varphi n(k)$ and $\varphi s(k)$:

$$\phi_n(k) = \frac{\hat{N}(k)}{\hat{N}(k) + \hat{S}(k)} \quad , \quad \phi_s(k) = \frac{\hat{S}(k)}{\hat{N}(k) + \hat{S}(k)} \tag{36}$$

However, instead of allowing $\varphi n$ and $\varphi s$ to change in time, we take these parameters to be con ial to the mean of the values obtained with Eq. (36). The values thus computed are $\varphi_n$=0.51a .9, practically identical to the default values.

Figure 4.4 shows the measured flows on the PCH approaching the Narbonne intersection. C iat during the middle part of the day, between approximately 9:00am and 3:00pm, both $W_{out}$ and $E_{out}$ are larger than $W_{in}$ and $E_{in}$, so there are more vehicles joining the PCH from Narbonne than leaving it during these hours. This is reflected in the result of the second stage, shown in Figure 4.5, where both $W_{join}$ and $E_{join}$ are larger than $W_{leave}$ and $E_{leave}$ between 9:00am and 3:00pm. The plots on the right side of Figure 4.5 compare the flow measured on Narbonne Avenue. With the estimated flows, the two lines in these plots would coincide perfectly had we allowed $\varphi_n$ and $\varphi_s$ to vary with time.

With regard to the optimization problem, several preprocessing steps were found to improve the performance of the solver. First, the problem contains several unused OD pairs which should be removed. These include diagonal elements in the OD matrix and OD pairs for which we expect zero flow. For example, in the Lomita network we do not expect any vehicles departing from the southern source on Walnut St. to use Bland St., since continuing on Walnut is clearly a better option. This is enforced in the example by eliminating the OD pair connecting Walnut South to Bland St. Second, some of the turning flows, which appear on the right hand side of Eq. (26), may be negligibly small. Because these flows represent the sum of a list of non-negative OD flows, it can be concluded that each of the OD flows are also very small and can therefore be discarded. Finally, there may be some OD pairs whose flow is given directly by a turning flow. This is the case for the WN and WS flows at the Walnut intersection. Making these simplifications, the size of the problem was reduced from 441 variables with 112 equality constraints to 222 variables with 50 equality constraints.

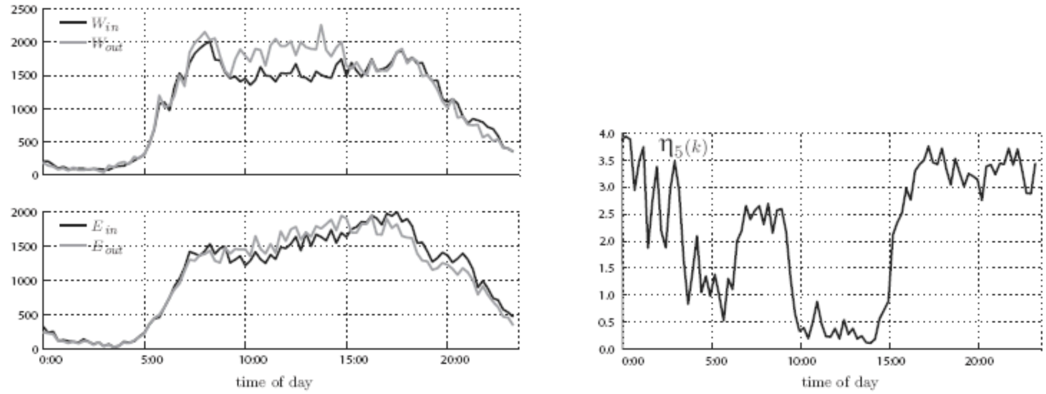


**Figure 4.4 Narbonne Avenue: Arterial Flows (vph) and Intersection Weight**

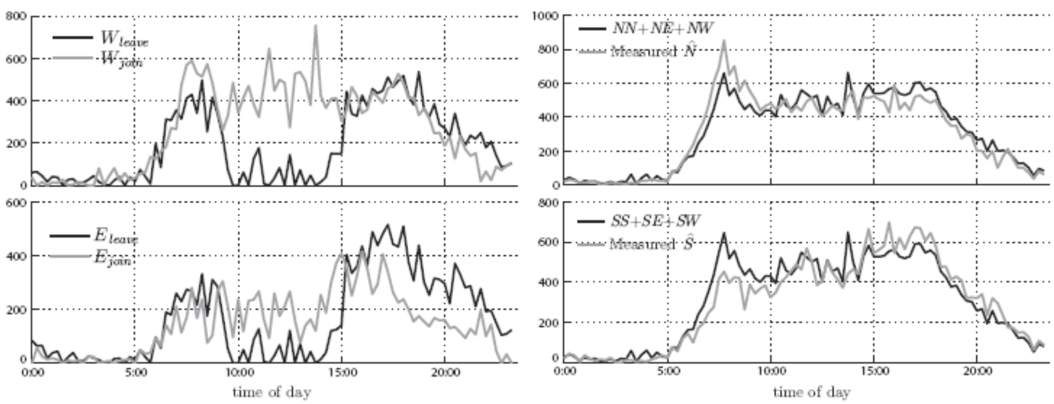


**Figure 4.5 Narbonne Avenue: Estimated Flows (vph)**

# CHAPTER 5

# PARAMICS PLUG-IN FOR SIGNAL CONTROL

In this Chapter we describe a plug-in we developed for the PARAMICS model to explicitly simulate a number of signal control strategies including actuated signal control and traffic responsive control.[4]

The plug-in implements many of the strategies developed by the Urban Traffic Control System (UTCS) program (11,27). The UTCS strategies are classified into three generations, the first generation consisting of methods for selecting timing plans developed offline from a stored library. The second and third generations automatically generate new timing plans based on measured data. The strategies included in the first generation UTCS are pretimed (time-of-day), traffic responsive, and critical intersection control. These were first tested in downtown Washington D.C. in early 1970's and they have since become standard features in many NEMA and 170 type controllers. Some of the basic strategies included in the new plug-in were previously implemented in PARAMICS by researchers at the University of California at Irvine under a previous PATH study (9). Our plug-in can be thought of as extending the existing simulation APIs to include complex adaptive strategies such as RHODES and TUC.

Figure 5.1 shows a typical layout for an eight phase four-legged signalized intersection. Three-legged intersections are also supported. The eight through and left-turn phases are numbered according to the NEMA convention. Each of these can be equipped with a set of *approach* detectors, and/or a set of *stopline* detectors. It is assumed that the single-lane loop detector model of PARAMICS is used.

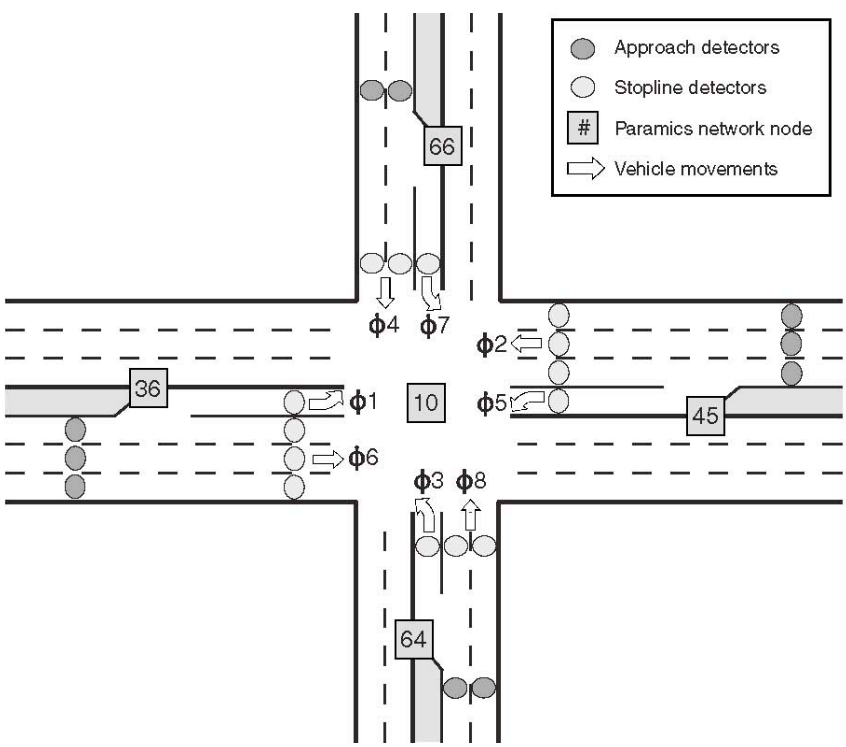


**Figure 5.1 Typical Signalized Intersection Layout**

---

[4] *This Chapter was originally published in "A PARAMICS Plugin for Actuated Signal Control and First Generation UTCS", by G. Gomes and A., Skabardonis, PATH Working paper, UCB-ITS-PWP-2006-8, July 2006.*

The function of the approach detector in vehicle actuated control is to extend the green time and to count vehicles for calculating the initial green time. Approach detection is used in traffic responsive control to generate the traffic patterns, while in critical intersection control it is used to compute green demands. These detectors are typically located about 200 feet upstream from the intersection stopline.

Stopline detectors are usually placed at or within a few feet upstream of the intersection stopline, and are used in actuated signal control to place calls for service on particular phases. Related to the placement of stopline detectors, we have noticed a behavior in PARAMICS that can cause the plug-in to fail. Namely, vehicles wishing to turn left at an intersection which are unable to change lanes into the dedicated left-turn lane before reaching the stopline, will often change lanes from a standstill and move laterally onto the stopline detector. These vehicles, because they do not trigger the PARAMICS detector in the normal way, do not place a call for service. They also prevent other vehicles from going over the detector. The left turn phase is therefore always skipped, and an unboundedly long queue can form. To avoid this problem, we recommend placing the stopline detectors in the model a bit farther upstream.

The main input file for the plug-in is param_main.txt. This file provides geometric, output, and high level control information for the simulation. An example input file is provided in Figure 5.2. Each line in the file begins with a case-sensitive token, followed by information separated by blank spaces or tab characters. Comments can be inserted using the percentage symbol (%).

The first line in the file determines the control strategy. The accepted values for the controller token are:

TOD …. time-of-day
ASC …. actuated signal control
ASC …. traffic responsive control
ASC …. traffic responsive with critical intersection control

The output period token defines the aggregation period for the loop detector output file in seconds. This value does not affect the controller behavior in any way. If omitted, the output file will not be generated.

Traffic responsive and critical intersection control use smoothed measurements, as opposed to raw detector measurements. The smoothed data is generated with a first-order filter:

$$
\begin{aligned}
\mathrm{vol}^s[k] &= (\tau)\,\mathrm{vol}^s[k-1] + (1-\tau)\,\mathrm{vol}^r[k] \\
\mathrm{occ}^s[k] &= (\tau)\,\mathrm{occ}^s[k-1] + (1-\tau)\,\mathrm{occ}^r[k]
\end{aligned}
$$

where the superscripts *s* and *r* indicate smoothed and raw variables. Raw measurements are 1-minute average values. $\tau$ is a filter parameter related to the time constant *Tc* (defined as the time to reduce the filter error by 63%) by $\tau = e^{-\Delta t / T_c}$,where $\Delta t$ is the simulation step size. The time constant *Tc* is defined by the user in seconds with the time constant token.

The rest of the file describes the signalized intersections. Each intersection begins with the node token, followed by the ID number for the PARAMICS network node. This number is the “node name” in PARAMICS Modeller. Although PARAMICS allows character strings for its node name, this plug-in requires a positive integer-valued node name. Also, the plug-in overrides all signal timing and priority settings defined in PARAMICS.

```
controller     ASC
outputperiod   60
timeconstant   30
%--------------------------------------------------------
node 10                              % Walnut St.
phase2nodes 45 36
phase4nodes 66 64

% phase#     1  2  3  4  5  6  7  8
protected    1  1  1  1  1  1  1  1
permissive   0  0  1  0  0  0  1  0

det 1 S n10_1_S_1
det 2 A n10_2_A_1 n10_2_A_2 n10_2_A_3
det 2 S n10_2_S_1 n10_2_S_2 n10_2_S_3
det 4 A n10_4_A_1
det 5 S n10_5_S_1
det 6 S n10_6_S_1 n10_6_S_2 n10_6_S_3
det 6 A n10_6_A_1 n10_6_A_2 n10_6_A_3
det 8 A n10_8_A_1

%--------------------------------------------------------
node 11                              % Eschelman St.
phase2nodes 46 48
phase4nodes 50 47

% phase#     1  2  3  4  5  6  7  8
protected    1  1  0  1  1  1  0  1
permissive   0  0  0  0  0  0  0  0

det 1 S n11_1_S_1
det 2 A n11_2_A_1 n11_2_A_2 n11_2_A_3
det 2 S n11_2_S_1 n11_2_S_2 n11_2_S_3
det 3 S n11_3_S_1
det 4 S n11_4_S_1
det 5 S n11_5_S_1
det 6 P n11_6_S_1 n11_6_S_2 n11_6_S_3
det 6 A n11_6_A_1 n11_6_A_2 n11_6_A_3
det 7 S n11_7_S_1
det 8 P n11_8_S_1
```

**Figure 5.2 PARAMICS Plug-In Main Input File (param_main.txt)**

The phase 2 nodes and phase 4 nodes tokens are used to orient the NEMA numbering scheme with respect to the intersection. The two numbers following phase 2 nodes and phase 4 nodes are respectively the *from* and *to* nodes for phases $\varphi 2$ and $\varphi 4$. To illustrate, the nodes from the sample input file are shown in Figure 5.1. For three-legged intersections, *the from and to* node for either phase 2 nodes or phase 4 nodes should be set to -1.

The protected and permissive tokens define the basic signaling features of the intersection. They are followed by a sequence of eight 0/1 values corresponding to the eight vehicle phases. Protected phases are those that are directly controlled by the traffic signal. Permissive phases are left turns that are allowed to proceed without an exclusive left turn phase (green arrow) when the opposing through phase has the right-of-way. For example, setting $\varphi 1$ to permissive means that vehicles in that phase are allowed to turn left during the green interval of $\varphi 2$. Only left-turn phases can be made permissive.

Loop detectors are assigned to phases using the det token. The syntax for this line is:

```
det [phase number 1-8] [detector type S|A] [list of detector names]
```

The detector names are those defined in PARAMICS Modeller. Phases are not required to have stopline or approach detectors. Furthermore, a single detector maybe associated with more than one phase.

Aggregated raw measurements from each of the loop detectors listed in param_main.txt are exported to output_loop.txt. This file contains a matrix with 7 columns and a row for every detector and aggregation interval. The column headers are given in Table 5.1 below.

**Table 5.1 Column Headers for output_loop.txt**

| Column # | Value |
|---|---|
| 1 | Start time for the aggregation period |
| 2 | Node ID |
| 3 | phase ID |
| 4 | 1=Approach ; 0=Stopline |
| 5 | Lane |
| 6 | Vehicle count [veh] |
| 7 | Occupancy $\in$[0,1] |

The program also produces a log file called output_log.txt, which collects error and warning messages generated during the simulation. Cycle lengths and green times are exported to output_controldata.txt. The column headers for this file are given in Table 5.2

**Table 5.2 Column Headers for output_control.txt**

| Column # | Value |
|---|---|
| 1 | Cycle start time |
| 2 | Intersection node number |
| 3 | Cycle length |
| 4-11 | Green time for phases 1 through 8 |

**5.1 Signal Timing Plans and Time-of-Day Control (TOD)**

Under TOD control, every intersection operates according to a predefined plan with fixed cycle length, green times (splits), offsets and phase sequences. PARAMICS offers an interface for creating preset plans, however we provide another here which is used by the traffic responsive controller described in section 5.4. The input file for TOD control is param_tod.txt (Figure 5.3). A set of plans is defined, each identified by an integer following the plan token. The todstart and todplan tokens are used to set the times for switching between plans. For example, the sample input file will tell the controller to switch from plan1 to plan2 at $t$ = 300 seconds, and back to plan1 at $t$ = 400 seconds.

```
todstart      0   300   400
todplan       1     2     1
transdelay   80
%================================================
plan 1
cyclelength 90
% .............................................
node 10
offset 0
%       phaseA   phaseB   green   yellow   red clear
stage     1        5       15       3         3
stage     2        6       25       4         3
stage     3        7       15       3         3
stage     4        8       17       4         3
% .............................................
node 11
offset 0
%       phaseA   phaseB   green   yellow   red clear
stage     1        5       22       4         3
stage     2        6       30       4         3
stage     4        8       25       4         3
%================================================
plan 2
cyclelength 60
% .............................................
node 10
offset 0
%       phaseA   phaseB   green   yellow   red clear
stage     1        5       10       3         3
stage     2        6       27       4         3
stage     3        7       10       3         3
stage     4        8       25       4         3
% .............................................
node 11
offset 0
%       phaseA   phaseB   green   yellow   red clear
stage     1        5       17       4         3
stage     2        6       25       4         3
stage     4        8       35       4         3
```

**Figure 5.3 Input File for Time-of-Day Control (param_tod.txt)**

Traffic signal controllers do not change timing plans instantaneously. This would result in truncated green and red intervals, which would be dangerous for drivers and pedestrians. Instead, they transition in a gradual manner, with small adjustments to each consecutive cycle. Different controller manufacturers employ different signal transition procedures. This plug-in does not attempt to reproduce those procedures, but instead inserts a delay between the decision to change plans and the realization of the change. A nominal delay value is defined by the user with transdelay. This nominal value is rounded up to an integer multiple of the current cycle length. In the example, plan1 will actually be in effect from $t = 0$ to $t = 390 = 300 +$ one plan1 cycle. Plan2 will then be activated from $t = 390$ to $t = 520 = 400 +$ two plan2 cycles.

Each network plan consists of a number of intersection-specific plans, which contain stage sequences and corresponding green, yellow, and red clearance times. It is left to the user to ensure that the phases combined in each stage are compatible. The start time for the intersection plan is shifted with respect to the master clock by offset seconds. The master clock rotates with cycle length cycle length in seconds. As in param_main.txt, each signalized intersection is identified by node. The sequence of stages is set with the stage token. The syntax for this line is:

```
stage [phase ID] [phase ID] [green sec] [yellow sec] [red clear sec]
```

In the case that a stage involves only one phase, the second phase ID should be set to zero. The sum of the green, yellow, and red clearance times for each intersection must not exceed the cycle length. All signals are set to red during the time remaining after the end of the last stage. Actual traffic controllers usually give this time to the major through movements. To mimic this behavior, it is recommended that an additional stage be added to fill in any leftover time.

## 5.2 Actuated Signal Control (ASC)

The plug-in implements a fully actuated eight-phase dual-ring traffic controller. Under actuated control each intersection operates locally, with no communication to adjacent intersections. The controller architecture has two levels: an upper level represented by a dual-ring controller (section 5.2.1) and a lower level that executes the individual phases (section 5.2.2).

### 5.2.1 The dual-ring controller

The objective of the dual-ring controller is to maintain safe conditions by allowing only compatible vehicle movements to enter the intersection at any time. Compatible movements or phases are those that a) belong to different rings, and b) are on the same side of the barriers (see Figure 5.4). Phases on the opposite sides of a barrier are in conflict, and therefore cannot be combined. For example, the only compatible options for $\varphi 2$ in Figure 5.4 are $\varphi 5$ and $\varphi 6$.

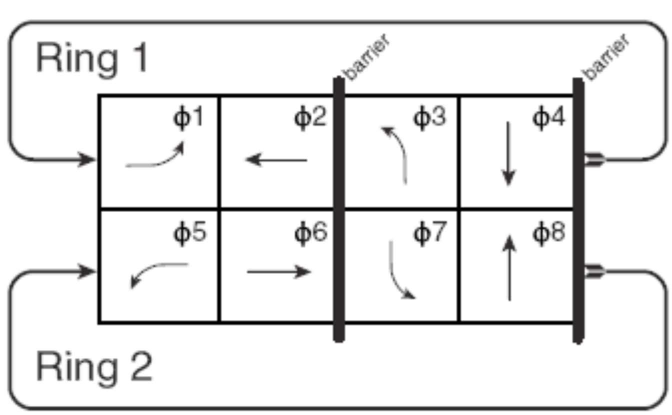


**Figure 5.4 Dual-Ring Controller**

The controller advances by initiating phases and, upon termination, selecting the next one to execute. The upcoming phase is found by searching the ring for the next protected phase which has either registered a vehicle presence or has been designated as a recall phase. Unprotected phases are always skipped. A recall phase cannot be skipped, even if no vehicle is registered by its stopline detectors. Recalls are typically applied to the through phases of major streets. They are also often used on cross streets lacking stopline detection. If no vehicle is registered on a non-recall phase, that phase maybe skipped.

Left turns can be either leading or lagging, depending on their position in the ring with respect to the opposing through phase. All of the left-turn phases in Figure 5.4 are leading. Phase $\varphi 1$ is made a lagging left turn by swapping its position with $\varphi 2$.

All of these features are included in the plug-in. The main task of updating the active phases is implemented in two steps. First, whenever one of the two active phases terminates, the next service able phase in the ring is found with the NextPhase() function. This function is passed an active or inactive phase, and returns the next protected phase with a vehicle call or are call status. It returns the input phase if it is still active. It also returns the input phase if none of the other three phases in its ring require service. The pair of phases found with NextPhase() may not be compatible. The second step is to find the number of barriers crossed in each ring in the transition from the current phase to the NextPhase(). This number—0, 1, or 2—may not be the same for both rings. The following logic is then applied to adjust the selected pair of phases so that they remain compatible:

```
if( both rings jump the same number of barriers )
      - transition to NextPhase() on both rings
else
      if( one ring jumps zero barriers )
            - that ring transitions to NextPhase()
            - the other remains in its current phase
      else      /* the only remaining case is one jumps one barrier, the other two */
            - the one that jumps one barrier goes to NextPhase().
            - the other goes to the compatible through phase.
```

### 5.3.2 Interval timing

The execution of a signal phase is illustrated in Figure 5.5. The dual-ring controller initiates the phase at "START". This point is synchronized with the transition from green to yellow of the previous phase ("END"). The initial wait period is equal in length to the yellow and red clearance intervals of the previous phase. Thus, the transition from wait to green is simultaneous with the transition from red clearance to idle of the outgoing phase. The green interval is divided into two portions: the initial green interval and the extension interval. The duration of the initial green interval is calculated based on the maximum number of vehicles registered by the approach loops during the preceding red interval. Each detected vehicle increases the green period by add per vehicle. The result is limited by min green and max initial:

$$\text{initial green} = \min\Big\{ \max\big\{ (\text{largest count}) \times \texttt{addpervehicle}\,,\, \texttt{mingreen} \big\}\,,\, \texttt{maxinitial} \Big\}$$

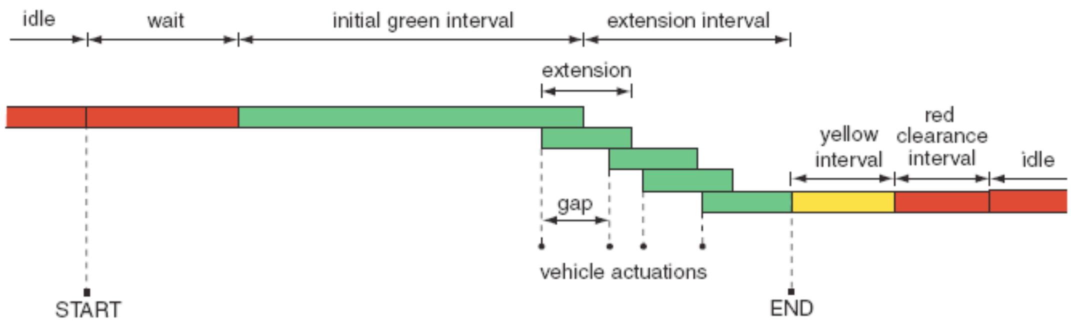


**Figure 5.5 Interval Timing**

During the green interval, vehicles detected by an approach loop are given extension seconds of green time to move through the intersection, without exceeding the maximum green duration of maxgreen. The green interval ends when the maximum green time is reached (max out), or when the time gap between consecutive vehicle actuations exceeds the largest permitted gap (gap out).The permitted gap is a function of time, as plotted in Figure 5.6. It starts at the *maxgap* value and begins to decrease after a vehicle is detected on a conflicting phase. The gap is reduced according to the controller parameters "reduce gap by" and "reduce every", until it reaches its minimum value, *mingap*. The yellow and red clearance intervals have fixed durations of yellow time and red clearance time.

The parameters involved in the ASC algorithm are entered by the user in param_asc.txt. A sample input file is shown in Figure 5.7. The recall token is used to set the recall status of the phases to on (1) or off (0). Entries in the lag left line indicate whether left turns are lagging (0) or leading (1). The remaining parameters are defined in seconds.

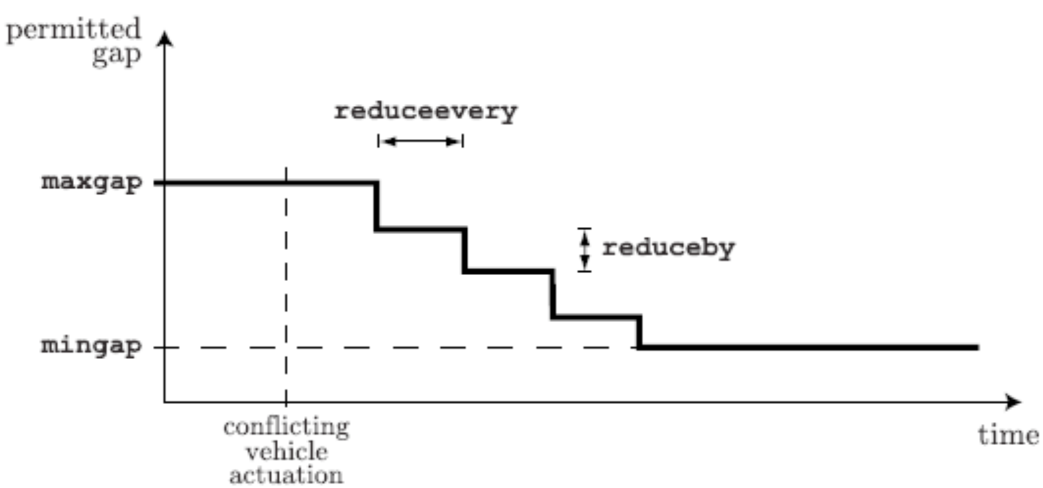


**Figure 5.6 Permitted Gap Function**

```
node 10
% phase#       1    2    3    4    5    6    7    8
recall         0    1    0    0    0    1    0    0
lagleft        1    0    0    0    1    0    0    0
mingreen       10   20   10   40   10   20   10   40
addpervehicle  2.1  2.1  2.1  2.1  2.1  2.1  2.1  2.1
maxinitial     20   30   20   50   20   30   20   50
maxgreen       30   40   30   60   30   40   30   60
extension      2.5  3.5  2.5  3.5  2.5  3.5  2.5  3.5
maxgap         2.5  5.0  2.5  2.5  2.5  5.0  2.5  2.5
mingap         2.5  2.5  2.5  2.5  2.5  2.5  2.5  2.5
reducegapby    0.1  0.1  0.1  0.1  0.1  0.1  0.1  0.1
reduceevery    2.0  2.0  2.0  2.0  2.0  2.0  2.0  2.0
yellowtime     3.0  3.5  3.0  3.5  3.0  3.5  3.0  3.5
redcleartime   1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0

node 11
% phase#       1    2    3    4    5    6    7    8
recall         0    1    0    0    0    1    0    0
lagleft        0    0    0    0    0    0    0    0
mingreen       10   20   0    40   10   20   0    40
addpervehicle  2.1  2.1  0    2.1  2.1  2.1  0    2.1
maxinitial     20   30   0    50   20   30   0    50
maxgreen       30   40   0    60   30   40   0    60
extension      2.5  3.5  0    3.5  2.5  3.5  0    3.5
maxgap         2.5  5.0  0    2.5  2.5  5.0  0    2.5
mingap         2.5  2.5  0    2.5  2.5  2.5  0    2.5
reducegapby    0.1  0.1  0    0.1  0.1  0.1  0    0.1
reduceevery    2.0  2.0  0    2.0  2.0  2.0  0    2.0
yellowtime     3.0  3.5  0    3.5  3.0  3.5  0    3.5
redcleartime   1.0  1.0  0    1.0  1.0  1.0  0    1.0
```

**Figure 5.7 Input File for Actuated Signal Control (param_asc.txt)**

### 5.3 Traffic Responsive Control (TRSP)

The two coordinated modes included in the first generation UTCS are time-of-day and traffic responsive control (TSRP). TOD is often sufficient for systems with predictable traffic volumes; TRSP is preferred if there are significant day-to-day fluctuations in demand. This is because TRSP can automatically respond to variations by selecting an appropriate signal timing plan for the measured traffic conditions.

Under TRSP control, flow and occupancy measurements from *system loops* are used to calculate a characteristic `vpko' (volume plus K occupancy) value for each loop $l$:

$$\mathrm{vpko}(l) \;=\; \mathrm{vol}^s(l) \;+\; \mathrm{K} \times \mathrm{occ}^s(l)$$

Here, $\mathrm{vol}^s(l)$ and $\mathrm{occ}^s(l)$ are the smoothed volume and occupancy measurements for system loop $l$, and K is a system constant. The current state of the network is represented by the array of all vpko($l$) values. System loops are typically the approach loops on the main arterial.

The controller also stores a number of signal timing plans with associated vpko signatures. The signature for the $p$-th plan consists of vpko values for each of the system loops vpko($p,l$). The controller selects a timing plan from its library by comparing the stored signatures to the measured traffic signature, and finding the best match in terms of a weighted 1-norm:

$$\Delta(p) = \sum_{l} \mathrm{W}(l) \left| \mathrm{vpko}(l) - \overline{\mathrm{vpko}}(p,l) \right| \qquad \text{for each plan } p$$

W(*l*) are loop-specific weighting factors. The plan with the smallest $\Delta(p)$ value is considered the best option for the current traffic condition. A transition to this plan will be initiated if the $\Delta(p)$ for the currently active plan is larger than a prescribed threshold. As with TOD control, the transition process is approximated with a user-defined delay interval.

Figure 5.8 shows the plug-in input file for TRSP control. The traffic responsive calculation is performed every update time seconds. kweight is the value of K. The system loops and their respective weighting factors W(*l*) are listed with the det token. The timing plans are given in param_tod.txt. The characteristic signatures for each of these plans are defined using plan and sig. The syntax for the sig line is:

*sig [loop name] [signature flow [vph]] [signature occupancy ∈ (0,1)]*

```
updatetime  900        % 15 minutes
minchange   1.0
kweight     20

% System loops ..............................
%---   Loop name   W
det    n10_2_A_1   3.0
det    n10_2_A_2   3.0
det    n10_6_A_1   3.0
det    n10_6_A_2   3.0
det    n11_2_A_1   3.0
det    n11_2_A_2   3.0
det    n11_6_A_1   3.0
det    n11_6_A_2   3.0

% Signatures ................................
plan 1
%---   Loop name   Flow   Occ
sig    n10_2_A_1   500    0.7
sig    n10_2_A_2   500    0.7
sig    n10_6_A_1   100    0.1
sig    n10_6_A_2   100    0.1
sig    n11_2_A_1   500    0.7
sig    n11_2_A_2   500    0.7
sig    n11_6_A_1   100    0.1
sig    n11_6_A_2   100    0.1

plan 2
%---   Loop name   Flow   Occ
sig    n10_2_A_1   100    0.1
sig    n10_2_A_2   100    0.1
sig    n10_6_A_1   500    0.7
sig    n10_6_A_2   500    0.7
sig    n11_2_A_1   100    0.1
sig    n11_2_A_2   100    0.1
sig    n11_6_A_1   500    0.7
sig    n11_6_A_2   500    0.7
```

**Figure 5.8 Input File for Traffic Responsive Control (param_trsp.txt)**

### 5.4 Critical Intersection Control (CIC)

Critical intersection control is a feature of many implementations of the UTCS software which complements the TRSP strategy. This feature enables the controller to respond more quickly to short-term variations in demand, while preserving coordination on the major street, by automatically adjusting the green times of *critical intersections* based on local measurements. CIC requires that all of the approaches to the critical intersections be equipped with approach loops.

CIC calculates the total *green demand* for all phases in a critical intersection using volume and occupancy measurements from the approach loops. This amount is reduced by the minimum green duration to obtain the excess green demand for each stage. The actual green times are then calculated by distributing the available excess green time among all the stages, in proportion to their excess green demand.

The plug-in provides two options for computing the green demands. The first is used by Los Angeles Department of Transportation ATSAC system (25):

$$\mathrm{gd}(l) \;=\; A\left(\mathrm{vol}^s(l)\right)^B \;+\; C\left(\mathrm{occ}^s(l)\right)^D$$

gd($l$) is the green demand measured by approach loop $l$. $A$, $B$, $C$, and $D$ are user-defined coefficients. $\mathrm{vol}^s(l)$ and $\mathrm{occ}^s(l)$ are smoothed volumes and occupancies. The second is the standard UTCS formula appearing in the Traffic Control Systems Handbook (9):

$$\mathrm{gd}(l) \;=\; K_1\,\mathrm{occ}^s(l) \;+\; K_2\,\mathrm{vol}^s(l) \;+\; K_3\,\mathrm{vol}^s(l)\,\mathrm{occ}^s(l)$$

$K1$, $K2$, and $K3$ are user-defined coefficients. The excess green demand for the stage $\mathrm{gd}_e(s)$ is calculated as the largest of the green demands for the associated approach loops, reduced by the minimum green interval, and limited below by zero:

$$\mathrm{gd}_e(s) \;=\; \max\left\{\;\max_l\left\{\mathrm{gd}(l)\right\} - \mathtt{mingreen}(s)\;\;;\;\;0\right\}$$

The actual green times g($s$) are found by apportioning the available excess green time $G_e$ to the stages:

$$\mathrm{g}(s) \;=\; \frac{\mathrm{gd}_e(s)}{\sum \mathrm{gd}_e(s)} \times \mathrm{G}_e \;+\; \mathtt{mingreen}(s)$$

$$\text{with}\qquad \mathrm{G}_e \;=\; \mathtt{cyclelength} - \sum_s\left(\mathtt{mingreen}(s) + \mathtt{yellowtime}(s) + \mathtt{redcleartime}(s)\right)$$

The parameters used by CIC are defined in param_cic.txt. A sample input file is shown in Figure 5.9. Update cycles defines the CIC update period as an integer multiple of the cycle length. The green demand formula is selected by setting gdfunction to ATSAC or UTCS. The coefficients in the green demand function are defined with gdcoef using the following format:

gdcoef [$A$] [$B$] [$C$] [$D$]    if gdfunction = ATSAC
gdcoef [$K_1$] [$K_2$] [$K_3$]    if gdfunction = UTCS

Critical lists the node numbers for the critical intersections. Min green is an optional input, which can be used to redefine the minimum green time used by CIC. If set to zero, the minimum green time will be that of param_tod.txt. Otherwise min green($s$) will be set to the largest of the entries for the phases combined in stage $s$. The cycle length, offset, yellow time, and red clear time are taken from param_tod.txt.

```
updatecycles  1
gdfunction    ATSAC
gdcoef        7.5 0.5 0.33 1.0
critical      10 11

%-------  node  1     2     3     4     5     6     7     8
mingreen  10    0     12    0     7     0     12    0     7
```

**Figure 5.9 Input File for Critical Intersection Control (param_cic.txt)**

### 5.5 Extensions—Adaptive Signal Control

This section of the report focuses on some of the architectural aspects of the plugin that allow it to be connected to external ``black box" control algorithms, such as RHODES and TUC. This architecture was designed to mimic the basic operation of a 170-type controller and its interaction with external control algorithms via *force-off*, *hold*, and *omit* messages.

Figure 5.10 is a schematic representation of the program structure assigned to each signalized intersection. The central component is the *signal manager*. The signal manager coordinates and passes information among the peripheral objects, which are the *lights*, the *detectors*, and the *controller*. The figure shows the information that is passed between the signal manager and each of the peripheral objects. An external signal control algorithm such as RHODES (whether isolated or coordinated), appears to the signal manager as a *controller* that translates detector and signal states ("all info" in the figure) into transition and flush requests. Transition requests are hold, force-off, and omit for each of the 8 phases. The controller can also request to flush or reset any of the vehicle counts maintained by the signal manager. The transition requests made by the controller are put through a series of consistency and safety checks (e.g. dual ring conditions, minimum green times) and either discarded or passed along to the lights as transition commands. The lights or phases in turn communicate their status to the signal manager, which decides based on detector information, controller requests, and signal states, which phases should be terminated.

This architecture captures the basic behavior of real 170-type controllers. Therefore, any control algorithm that is designed to operate using standard hold, force-off, and omit messages (such as RHODES) can be easily connected to the plugin and simulated in Paramics. In the case of RHODES, the connection was made using the Windows socket mechanism. In the case of TUC, we were provided by the developers with source code.

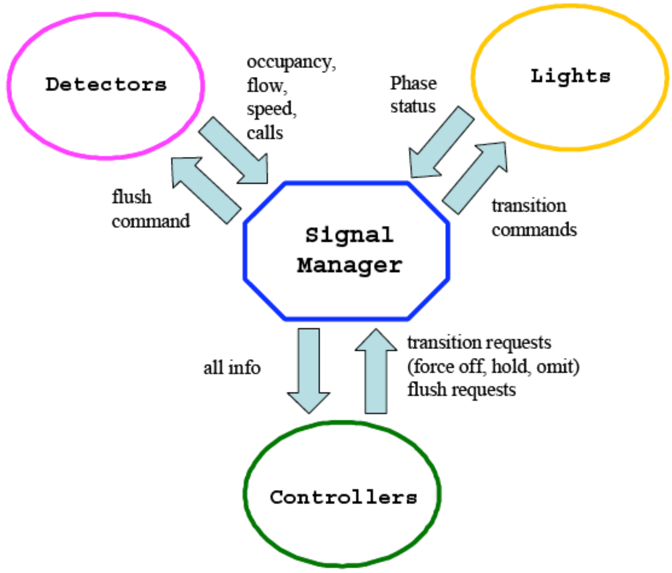


**Figure 5.10 Plugin Architecture**

# CHAPTER 6
# CONCLUSIONS

The scope of the project is to identify and select the most promising of existing adaptive control algorithms, develop improved algorithm(s) as appropriate, conduct field operational tests on real-world arterial corridors, and develop recommendations for deployment of adaptive control. The report describes the work performed in the first phase of the project (PATH Task Order 5322).

- A test site was chosen for evaluation of adaptive signal control strategies. The selected test site is a section of the Pacific Coast Highway running through the city of Lomita, managed by Caltrans District 7. It includes seven signalized intersections. Over seventy days of loop detector data were collected for this site using the CTNet software. Additional data on geometrics and signal settings were collected from Caltrans staff and field visits to the site.

- A methodology was devised for synthesizing a time-varying OD matrix from the loop detector data. The calibrated OD matrix was input to the PARAMICS model to simulate existing operating conditions at the test site.

- A signal control plug-in was written to model control strategies with PARAMICS. It simulates non-adaptive strategies such as pretimed, isolated actuated, coordinated actuated, traffic responsive, and critical intersection control, as well as adaptive strategies such as RHODES and TUC. The plug-in is not limited to Lomita, it can be applied to any PARAMICS network.

Research in the second phase of the project (PATH Task Order 6322) includes the following tasks:

- Evaluation of the selected existing and proposed adaptive control algorithms through simulation. The adaptive strategies to be tested will include the RHODES and TUC strategy, plus the ACS Lite strategy (subject to the availability of the source code). Sensitivity analyses will be also performed by changing model input data (e.g., traffic volumes) and parameters (e.g., saturation flows) to test the performance of the proposed algorithm under a wide range of operating conditions. The best adaptive control algorithm will then be selected for field testing based on the analysis of the simulation results.
- Selection of the most promising adaptive control algorithms. Design and develop new functions and features to enhance the selected adaptive control algorithm(s). The end product of this Task will be detailed documentation and software code of the proposed algorithms.

- Development of a test plan for field implementation. The test plan will specify the hardware and software requirements and level of effort required for field implementation of the adaptive algorithm. The test plan will also specify the duration of the field experiment, the method of data collection, the duration of the field experiment, and the analysis of the results.

- Development of guidelines for deployment of adaptive control based on the findings from the simulation results. The issues to be addressed in the deployment plan include the system performance, costs (installation, operations and maintenance), system reliability, and integration with other traffic management systems. We also provide an assessment on how well the proposed adaptive system adapts to changing traffic conditions, and how well they, if not stand-alone system, coexist with standard time-of-day signal control systems.

It is expected that baseline simulations of the existing adaptive control strategies will be completed by March 31, 2009. The testing of alternative strategies though simulation will be performed in April and May 2009. The final report for Phase 2 of the study (TO 6322) will be completed in June of 2009 and will include the following:

- Findings from the testing of the alternative strategies through simulation
- A detailed test plan for field implementation and testing of the most promising strategy selected based on the simulation results. The test plan will address data collection requirements and sources (e.g., CTNET or ATCS), hardware & software requirements, performance measures and duration of "before" and "after" studies
- Development of guidelines for deployment of adaptive systems based on the simulation results

Phase 3 of the study consists of field testing of the most promising strategy on the Lomita Avenue test site. It is expected to begin in the Fall of 2009, subject to the approval of the research proposal.

## REFERENCES


1. Bell M. The estimation of origin-destination matrices by constrained generalized least squares. *Transportation Research Part B*., 25(1), 1991.
2. California Department of Transportation (Caltrans), CTnet User Manual. Technical Report 1.43,
3. Cascetta E. Estimation of trip matrices from traffic counts and survey data: A generalized least squares estimator. *Transportation Research Part B*., 18(4/5), 1984.
4. Chen C., Kwon J., Skabardonis A., and Varaiya P. Detecting errors and imputing missing data for single loop surveillance systems. *Transportation Research Record*, 1855, 2003.
5. Chu, L., H., Liu, M. McNally and W. Recker, "Development of the capability –enhanced PARAMICS simulation environment," *California PATH Research Report, UCB-ITS-PRR-2005-12,* April 2005.
6. Cremer M. and Keller H. Dynamic identification of flows from traffic counts at complex intersections. 8th *International Symposium on Transportation and Traffic Theory*. 1981.
7. Dixon M. and L. Rilett. Population origin-destination estimation using automatic vehicle identification and volume data. *Journal of Transportation Engineering*, 131(2), 2005.
8. Dowling, R., and A. Skabardonis, "Guidelines for Calibration of Traffic Simulation Models: Framework and Applications," Transportation Research Record # 1876, 2004.
9. Federal Highway Administration. *Traffic control systems handbook*. February 1996.
10. Federal Highway Administration, CORSIM Version 6.0 Users Manual, prepared by ITT Systems, US Department of Transportation, Washington DC, 2006.
11. Henry R.D., R.A. Ferlis, and J.L. Kay. Evaluation of UTCS control strategies. Technical Report FHWA-RD-76-150, JHK & Associates / Peat, Marwick, Mitchell & Co., August 1976.
12. http://www.autoitscript.com/autoit3
13. Kwon J. and Varaiya P. Real-time estimation of origin-destination matrices with partial trajectories from electronic toll collection tag data. *Transportation Research Record*.No.1923, 2005.
14. Li B. and De Moor B. Recursive estimation based on the equality-constrained optimization for intersection origin-destination matrices. *Transportation Research Part B*., 33: 203-214,1999.
15. Li B. and De Moor B. Dynamic identification of origin-destination matrices in the presence of incomplete observations. *Transportation Research Part B*.,36: 37-57, 2002.
16. Lo H.P., Zhang N., and Lam W.H.K. Estimation of an origin-destination matrix with random link choice proportions: A statistical approach. *Transportation Research Part B*., 30(4), 1996.
17. Maher M.J. Inferences on trip matrices from observations on link volumes: A Bayesian statistical approach. *Transportation Research Part B*., 20:435-447,1983.
18. Nihan N. and Davis G. Recursive estimation of origin-destination matrices from input/output counts. *Transportation Research Part B*., 21(2),1987.

19. Quadstone Ltd, "PARAMICS microscopic traffic simulation software. http://Paramics-online.com.

20. Shelby, et al, "Overview and Performance Evaluation of ACS Lite: Low-Cost Adaptive Signal Control System, "paper 08-0334, presented at the 87th Annual Meeting of the Transportation Research Board, Washington DC, January 2008.

21. Sherali H. A linear programming approach for synthesizing origin-destination trip tables from link traffic volumes. *Transportation Research Part B*., 28(3),1994.

22. Skabardonis A., "Evaluation of the Fuel Efficient Traffic Signal Management (FETSIM) Program: 1983-1993," Research Report UCB-ITS-RR-94-11, University of California, Berkeley, 1994.

23. Skabardonis A., "Benefits of Advanced Traffic Signal Systems," in *Measuring the Performance of ITS in Transportation Services*," David Gillen Editor, Kluwer Academic Publishers, 2003.

24. Skabardonis, A., "Assessing The Benefits Of Signal Hardware Improvements," Transportation Research Record No. 1554, 1996

25. Skabardonis, B. Gallaggher, and K. Patel. Determining the capacity benefits of real-time signal control at an intersection. *California PATH Working Paper UCB-ITS-PWP-2000-25*, 2000.

26. Spiess H. A maximum likelihood model for estimating origin-destination matrices. *Transportation Research Part B*.,2 1(5),1987.

27. UTCS-functional description-enhanced first generation software. Technical Report FHWA-TS-79-228, Honeywell, Inc., August 1979.

28. Wu J. A real-time origin-destination matrix updating algorithm for on-line applications. *Transportation Research Part B*., 31(5), 1997.

29. Van Zuylen H. and Willumsen L. The most likely trip matrix estimated from traffic counts. *Transportation Research Part B*.,14,1980.

30. Yang H. Heuristic algorithms for the bilevel origin destination matrix estimation problem. *Transportation Research Part B*., 29(4),1995.

31. Zhou X. and Mahmassani H. Dynamic origin-destination demand estimation using automatic vehicle identification data. *IEEE Transactions on Intelligent Transportation Systems*, 7(1),2006.

# BIBLIOGRAPHY


1. Bell, M.C., and R. D. Bretherton. 1986. Ageing of fixed-time traffic signal plans. Paper presented at the IEE 2nd International Conference on Road Traffic Control.
2. Martin, P.T., and L. M. Hockaday. 1995. SCOOT - an update. ITE Journal 65. p.44-48.
3. Hunt, P.B., D. I. Robertson, R. D. Bretherton, and R. I. Winton. 1981. SCOOT - a traffic responsive method of coordinating signals. Laboratory Report 1014. Transport and Road Research Lab, Crowthorne, Berkshire, U.K.
4. Hansen, B.G., P. T. Martin, and H. J. Perrin. 2000. SCOOT real-time adaptive control in a CORSIM simulation environment. Transportation Research Record 1727. National Research Council, Washington, D.C. p.27-31.
5. Lowrie, P.R. 1982. SCATS: The Sydney Co-ordinated Adaptive Traffic System. Paper presented at the IEE International conference on road traffic signaling at London, U.K.
6. Hu, K. 2000. LADOT's adaptive traffic control system. Workshop on Adaptive Signal Control Systems, 79th Annual Meeting, Transportation Research Board, Washington D. C..
7. Khoudour, L., J.B. Lesort, and J.L. Farges. 1991. PRODYN - Three years of trials in the ZELT Experimental Zone. Recherche - Transports - Securite, English Issue, no. 6. Special Traffic Management. p.89-98.
8. Mauro, V., and D. Di Taranto. 1990. UTOPIA. Paper presented at the 6th IFAC/IFIP/IFORS Symposium on Control and Communication in Transportation at Paris, France.
9. Head, K.L., P. B. Mirchandani, and D. Shepherd. 1992. A hierarchical framework for real-time traffic control. Transportation Research Record 1360. p.82-88.
10. Boillot, F., J. M. Blossville, J. B. Lesory, V. Motyka, M. Papageorgiou, and S. Sellam. 1992. Optimal signal control of urban traffic networks. Paper presented at the IEE Conference on Road Traffic Monitoring and Control at Stevenage, England. p.75-79.
11. Robertson, D.I. 1969. TRANSYT: A traffic network study tool. Laboratory Report 1014. Transport and Road Research Lab, Crowthorne, Berkshire, U.K.
12. Mazzamatti, M.V., D.V.V.F. Netto, L.M. Vilanova and S.H. Ming. 1998. Benefits gained by responsive and adaptive systems in São Paolo. Paper presented at the IEE Road Transport Information and Control, Conference Publication No. 454.
13. Taale, H., W. C. M. Francis and J. Dibbits. 1998. The second assessment of the SCOOT system in Nijmegen. Paper presented at the IEE Road Transport Information and Control, Conference Publication No. 454.
14. Quan, B.Y., J.C. Greenough, and W.L. Kelman. 1993. The Metropolitan Toronto SCOOT demonstration project. 72nd TRB Annual Meeting, Washington, D.C.
15. Peck, C., P.T.W. Gorton, D. Liren. 1990. The application of SCOOT in developing countries. 3rd International IEE Conference on Road Traffic Control, London, p.104-109.
16. Department of Transport, Hereford and Worcester County Council, and Transportation Planning Associates. 1986. Evaluation of Urban Traffic Control in Worcester. Available from TCC division D Tp, Tollgate House, Bristol. May.
17. Chandler, M.J.H., and D.J. Cook. 1985. Traffic control studies in London: SCOOT and bus detection, PTRC Annual Summer Meeting at Seminar M, P269. p.111-128.

18. Powell, R.J. 1985. SCOOT in Southampton. Paper presented at the PTRC Annual Summer Meeting at Seminar M, P269. p.97-110.

19. Coeymans-avaria, J.E., et. al. 1995. SCOOT in Santiago. Paper presented at the 23rd European Transport Forum. PTRC September 1995. P394. p.183-195.

20. Gartner, N.H. 1983. OPAC: A demand responsive strategy for traffic signal control. Transportation Research Record 906. National Research Council, Washington, D.C.

21. Head, K.L. 1995. An event-based short-term traffic flow prediction model. Transportation Research Record 1510. National Research Council, Washington, D.C. p.45-52.

22. Sen, S., and K. L. Head. 1997. Controlled Optimization of Phases at an Intersection. Transportation Science 31. p.5-17.

23. Dell'Olmo, P., and P. B. Mirchandani. 1996. A model for real-time traffic co-ordination using simulation based optimization. Advanced Methods in Transportation Analysis. (ed. L. Bianco and P. Toth) Springer-Verlag, Germany. p.525-546.

24. Dell'Olmo, P., and P. B. Mirchandani. 1995. REALBAND: An approach for real-time co-ordination of traffic flows on networks. Transportation Research Record 1494.

25. Mirchandani, P.B., K. L. Head, A. Knyazyan, and W. Wu. 2000. An approach towards the integration of bus priority, traffic adaptive signal control and bus information/scheduling system. 8th International Conference on Computer-Aided Scheduling of Public Transportation at Berlin, Germany.

26. Dr. William C. Taylor and Sorawit Narupiti, Incident Management FAST-TRAC Phase IIB Deliverables #20. The Model Analysis Report on the Benefits of SCATS in Alleviating the Impacts of Incidents EECS-ITSLAB-FT96- 112, 1995.

27. William C. Taylor and Ahmed S. Abdel-Rahim,*Incident Management Under SCAT Adaptive Control System FAST-TRAC Phase III Deliverable #11.* Final Report On Incident Management Under SCATS Adaptive Control System EECS-ITS LAB-FT98-084, 1998.

28. Hunt P.B. and Holland, T.R., *The effect of an incident in a SCOOT system - In simulation and on street.* Traffic Engineering and Control., February1985. Volume 26. (No.2.).

29. Homes G. N. and D.R. Leonard, *A Study to identify the frequency and importance of incidents in causing congestion.* Proc. 20th PTRC Summer annual meeting: Traffic Management and Operations, September 1992: p. 153-156.

30. P.B. Hunt, D.I.R. and R.D. Bretherton., *The SCOOT on-line traffic signal optimization technique.* Traffic Engineering and Control, April 1982: p. 190-192.

31. Bretherton, R.D., and G.T. Bowen. 1990. Recent enhancements to SCOOT - SCOOT version 2.4. Paper presented at the 3rd IEE International Conference on Toad Traffic Control.

32. Bretherton, R.D., and G.T. Bowen. 1998. SCOOT Version 4. Paper presented at the IEE Conference on Road Transport Information and Control at London.

33. Bretherton, R.D. Current developments in SCOOT: Version 3. Transportation Research Record 1554. National Research Council, Washington, D.C.

34. Bowen, G.T., and R.D. Bretherton. 1996. Latest developments in SCOOT - Version 3.1, 8th IEE International Conference on Road Traffic Monitoring and Control.

35. Hansen, B.G., and P. T. Martin. 1999. Online Testing of SCOOT using CORSIM: Development of a CORSIM Run-Time Extension. Working Paper, University of Utah.

36. Hounsell, N., Intelligent Bus Priority in London: Evaluation and Exploitation in INCOME, the 6th World Congress Conference, Turin, Italy, November 1999.

37. Siemens Traffic Controls Ltd., Peek Traffic Ltd. and TRL Ltd. Bus Priority in SCOOT, 2001, http://www.scoot-utc.com/SCOOTFacilities/busprior.htm, accessed June 12, 2002.

38. Robertson D.I., Research on the TRANSYT and SCOOT Methods of Signal Coordination. ITE Journal, Vol. 56, No. 1 1986, pp. 36 - 40.

39. Mirchandani P., Head L., Knyazyan A. and Wu W., An Approach Towards The Integration of Bus Priority, Traffic Adaptive Signal Control, and Bus Information/Scheduling System. the 8th International conference, computer-aided scheduling of public transport, Berlin, Germany, June 2000.

40. Head, K.L., Mirchandani P.B. and Shelby S., The RHODES Prototype: A Description and Some Results. Paper presented at the 77th Transportation Research Board Annual Meeting (Paper No. 981399), Washington D.C., 1998.

41. Stallard C., and L.E. Owen, Evaluating Adaptive Signal Control Using CORSIM, the proceedings of the 30th conference on winter simulation, Washington D.C., December 1998.

42. Fellendorf. M., Public Transport Priority Within SCATS - a Simulation Case Study in Dublin, 67th Annual Meeting of the institute of transportation engineers, Boston 1997.