

Modeling of Traffic Signal Control and Transit Signal Priority Strategies in a Microscopic Simulation Laboratory

by

Angus P. Davol

Sc.B. in Civil Engineering (1997)
Brown University, Providence, RI

Submitted to the Department of Civil and Environmental Engineering
in partial fulfillment of the requirements for the degree of

Master of Science in Transportation

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2001

© 2001 Massachusetts Institute of Technology. All rights reserved.

Signature of Author
Department of Civil and Environmental Engineering
August 17, 2001

Certified by
Moshe E. Ben-Akiva
Edmund K. Turner Professor of Civil and Environmental Engineering
Thesis Supervisor

Certified by
Haris N. Koutsopoulos
Operations Research Analyst
Volpe National Transportation Systems Center
Thesis Supervisor

Accepted by
Oral Buyukozturk
Chairman, Departmental Committee on Graduate Studies

Modeling of Traffic Signal Control and Transit Signal Priority Strategies in a Microscopic Simulation Laboratory

by

Angus P. Davol

Submitted to the Department of Civil and Environmental Engineering
on August 17, 2001 in partial fulfillment of the requirements
for the degree of Master of Science in Transportation.

Abstract

This thesis describes the modeling and implementation of an advanced traffic signal controller within a microscopic simulation environment, thus creating a laboratory for the evaluation of advanced traffic control strategies, including transit signal priority. The simulation tool used for this research is MITSIMLab, a microscopic traffic simulation laboratory developed for ITS design and evaluation.

The controller is designed with a generic and flexible logic that allows it to simulate a wide range of traffic signal control types and strategies. These control strategies include both isolated and coordinated intersection control, with fixed-time and demand-responsive logic. The controller is also designed with a modular structure that allows specialized features of advanced control strategies to be implemented within the controller framework. This framework is used to implement transit signal priority in MITSIMLab, allowing the simulation of both passive and active signal priority strategies.

The capabilities of the controller are illustrated through a case study in which MITSIMLab is applied to an urban arterial network in Stockholm, Sweden, where an existing signal priority strategy is implemented. An evaluation of the currently implemented system is performed, and recommendations for improvement and further study are offered.

Thesis Supervisor: **Moshe E. Ben-Akiva**
Edmund K. Turner Professor of Civil and Environmental Engineering

Thesis Supervisor: **Haris N. Koutsopoulos**
Operations Research Analyst
Volpe National Transportation Systems Center

Acknowledgements

I know it will be impossible to acknowledge all the people who have helped make these past two years the rewarding experience it has been, but I'll do my best...

First, my advisors, Moshe Ben-Akiva and Haris Koutsopoulos, with whom it has been my privilege to work. I thank Moshe for the friendship, support, and trust he has given me in my time here. His warmth, humor, and laid-back style make working with him truly a pleasure. I thank Haris for his friendship, his guidance, and most of all the selfless commitment that he has shown to me and to the rest of his students. I hope he knows the depth of appreciation and respect that I hold for him – feelings that I know are shared by all of us here at the ITS Lab.

I thank my fellow students at the lab, especially Tomer Toledo, who has served as my unofficial third advisor over the past two years and whose stable presence makes him always the voice of reason. Thanks also to Margaret Cortes—my office-mate, gossip-buddy, and partner in crime—for her constant support and encouragement and for always keeping me happy, entertained, and well fed during our countless hours at the lab together.

I thank all my sources of technical support: Tobias Johansson and Jan Björck at *Gatu-och Fastighetskontoret* in Stockholm for their assistance with the network data and the signal and PRIBUSS logic; Margaret Cortes and Isaac Moses for their work on the buses in MITSIMLab; and the distinguished line of DynaMIT RAs who have served as de facto network administrators—Bruno Fernandez Ruiz, Manish Mehta, Josef Brandriss, and Rama Balakrishna—for keeping the system up and running and for continually taking pity on a bewildered Macintosh user.

Tack så mycket to our colleagues in Stockholm—Christer Lundin and Tobias Johansson of GFK, and Ingmar Andréasson and Wilco Burghout of *Kungl Tekniska Högskolan*—for their hospitality during our visits there.

I thank the faculty and staff of CTS for their dedication and kindness. Special thanks to Leanne Russell, whose presence always made the trek to Building 1 worthwhile.

I thank MIT and GFK for their financial support of my studies.

I thank all my friends: my CTS friends for making me realize that it's okay to get excited about things like public transportation; my MITSO friends for letting me relive crazy college days that I never actually had the first time around; and my non-MIT friends for making it impossible to forget that life does exist off-campus.

Finally, I thank my family for their unconditional love and support. I realize how lucky I am to have them.

Contents

1	Introduction	15
1.1	Objectives.....	17
1.2	Thesis Outline	18
2	Background and Literature Review	19
2.1	Traffic Signal Control	19
2.1.1	Terminology.....	19
2.1.2	Control Types.....	22
2.2	Transit Priority Strategies.....	27
2.2.1	Passive Priority Strategies.....	28
2.2.2	Active Priority Strategies	30
2.2.3	Evaluation of Priority Strategies	35
2.3	Summary	36
3	Generic Traffic Signal Controller: Design and Implementation	39
3.1	MITSIMLab	39
3.2	Controller Design	41
3.2.1	Basic Logic Elements.....	41
3.2.2	Structure	42
3.2.3	Logic.....	43
3.2.4	Conditions	46
3.3	Control Logic Capabilities	49
3.3.1	Pretimed Control	49
3.3.2	Actuated Control	51
3.3.3	Adaptive Control.....	53

3.4	Supported Controller Types	53
3.4.1	NEMA/Model 170.....	54
3.4.2	European Standards.....	56
3.5	Transit Priority Capabilities	57
3.6	Implementation of Advanced Control Strategies.....	59
4	Implementation of Transit Signal Priority	61
4.1	PRIBUSS.....	61
4.1.1	General Concepts	61
4.1.2	Priority Actions	64
4.2	Implementation in MITSIMLab.....	70
4.2.1	Generic Controller Enhancements	70
4.2.2	Logic Specification	72
4.3	Strategy Implementation	77
5	Case Study	79
5.1	Study Network.....	79
5.2	Simulation Setup	83
5.3	Experimental Design	86
5.4	Results	88
5.4.1	Travel Time Comparisons.....	88
5.4.2	Travel Time Variability.....	93
5.4.3	Effects of Increased Demand	95
5.4.4	Parameter Sensitivity.....	97
5.5	Recommendations	99
6	Conclusions	101
6.1	Summary	101
6.2	Findings.....	102
6.3	Future Research.....	104
A	Logic Conditions for Generic Traffic Signal Controller	105

B Example Signal Input File: Phase-Based Specification	109
C Example Signal Input Files: Signal Group Specification	113
C.1 Pretimed-Coordinated Control	113
C.2 Pretimed-Coordinated Control with Bus Priority.....	115
Bibliography	117

List of Figures

2-1	Example intersection.....	20
2-2	Example signal phase diagram.....	20
2-3	Example signal group diagram.....	21
2-4	Relation between phase and signal group specifications.....	22
2-5	Types of signal control logic.....	22
2-6	Green interval extension of an actuated phase.....	23
2-7	Example of a gap-reduction function.....	24
2-8	Progressive traffic flow under signal coordination.....	26
2-9	Bi-directional progressive flow under signal coordination.....	26
2-10	Vehicle trajectory without signal priority.....	28
2-11	Vehicle trajectory with transit phase extension.....	31
2-12	Vehicle trajectory with early start to transit phase.....	31
2-13	Vehicle trajectory with insertion of extra transit phase.....	32
3-1	Elements of MITSIMLab and their interactions.....	40
3-2	Overall logic of the generic controller.....	44
3-3	Condition evaluation logic within signal group of generic controller.....	45
3-4	Four-phase controller diagram.....	54
3-5	Eight-phase (dual-ring) controller diagram.....	55
3-6	Phase order for dual-ring controller.....	56
3-7	Signal group diagram.....	57
4-1	Time windows for priority calls.....	63
4-2	Execution of green extension.....	65
4-3	Execution of shortening of current phase.....	67
4-4	Execution of insertion of extra phase.....	68

4-5	Execution of green restart.	69
5-1	Hornstull network location.....	80
5-2	Schematic of Hornstull network.	81
5-3	Bus facilities on Hornstull network.....	82
5-4	Lane configuration at Hornstull.	83
5-5	Traffic count locations.	84
5-6	Graphical representation of peak-hour O-D matrix for Hornstull network.	85
5-7	Average bus travel times by priority implementation (peak demand).....	90
5-8	Comparison of bus trajectories showing benefit from priority.	91
5-9	Comparison of bus trajectories showing no benefit from priority.	93
5-10	Average travel time comparison under different demand scenarios.....	97
B-1	Phase and timing diagram for eight-phase controller.	110
C-1	Signal group diagram (Hornsgatan-Varvsgatan intersection).....	113

List of Tables

3.1	Pretimed-isolated logic specification (by signal groups).....	50
3.2	Pretimed-isolated logic specification (by phases).....	50
3.3	Pretimed-coordinated logic specification.....	51
3.4	Pretimed-coordinated logic specification (alternate).....	51
3.5	Actuated-isolated logic specification (basic).....	51
3.6	Actuated-isolated logic specification (advanced).....	52
3.7	Actuated-coordinated logic specification.....	53
3.8	Simple transit priority specification – phase extension.....	58
3.9	Simple transit priority specification – phase shortening.....	58
4.1	Basic pretimed-coordinated logic specification.....	73
4.2	“Green Extension” logic specification – priority signal group.....	73
4.3	“Phase Shortening” logic specification – priority signal group.....	74
4.4	“Phase Shortening” logic specification – shortened signal group.....	74
4.5	“Phase Insertion” logic specification – priority signal group.....	75
4.6	“Phase Insertion” logic specification – preceding signal group.....	75
4.7	“Phase Insertion” logic specification – following signal group.....	76
4.8	“Green Restart” logic specification – priority signal group.....	77
5.1	Average travel times for different priority implementations (peak demand).....	88
5.2	Change in average travel times for different priority implementations (peak demand).....	89
5.3	Standard deviations of travel time for different priority implementations (peak demand).....	94
5.4	Change in standard deviations of travel time for different priority implementations (peak demand).....	94
5.5	Average travel times for different priority implementations (high demand).....	95

5.6	Change in average travel times for different priority implementations (high demand).....	96
5.7	Average travel times for different priority implementations and demand levels.....	98
5.8	Change in average travel times for different priority implementations and demand levels.....	98
A.1	Basic logic conditions (general and change conditions).....	106
A.2	Basic logic conditions (hold and skip conditions).	107
A.3	PRIBUSS logic conditions.....	108
B.1	Signal timing data for eight-phase controller.....	109
C.1	PRIBUSS parameters (Hornsgatan-Varvsgatan intersection).....	115

Chapter 1

Introduction

Intelligent Transportation Systems (ITS), which apply advanced technologies to surface transportation systems, are widely viewed as the solution to the transportation problems that our society faces. In many areas, a steadily increasing demand for mobility is confronting economic, social, and physical constraints on transportation infrastructure. These constraints include reduced funding for transportation projects, social and environmental concerns about infrastructure expansion, and, in urbanized areas, a lack of physical space to devote to such projects. ITS applications, in which technology is used to increase the operating efficiency and capacity of transportation infrastructure, can supplement or even replace infrastructure development, providing more effective mobility solutions at less of a cost to society.

Urban traffic control is a major area in which ITS can be applied. At the local level, traffic signals are designed to manage vehicle conflicts at intersections, allocating time among the conflicting traffic streams which must share the use of the intersection. The logic by which the signal controller allocates usage of the intersection can range from basic fixed-time methods to intelligent strategies that detect and respond to traffic conditions in real time. At a higher level, however, traffic signals can part of a broader control strategy. In this case, signal controllers are used as tools for managing traffic flow, either along a corridor or throughout a network, to provide a more efficient use of the urban street network.

ITS applications for transit, or Advanced Public Transportation Systems (APTS), have the same goals, namely improvements of efficiency without the need for major infrastructure enhancements. One such application is Bus Rapid Transit (BRT), a transit

concept that uses buses to provide a high level of service usually associated with rail transit. The reason that rail transit can provide such a high level of service, however, is that it operates on a right-of-way that is fixed and exclusive. This is typically not the case for city buses, which instead operate on a shared right-of-way in an open and more chaotic system. In such an environment, buses face delays caused by interactions with other vehicles and by the presence of traffic signals at intersections. These two factors can have a significant negative impact on operations.

One method of addressing these operational challenges is by the use of infrastructure solutions such as exclusive bus lanes. While often effective in reducing delays due to congestion, these solutions can be prohibitively expensive or, in many urban areas, infeasible due to inadequate street space. Another method is the use of control strategies, which use the existing traffic signal control system to give priority to transit vehicles. This convergence of APTS and urban traffic control is known as transit signal priority.

Transit signal priority strategies can be categorized into two basic types: passive and active. Passive priority strategies are those that use static signal settings to favor streets with transit routes. These rely on signal timing plans that are prepared off-line and are designed to impede transit vehicles as little as possible. Active priority measures are those which employ dynamic detection and response to transit vehicles, altering signal settings in real-time in order to reduce delay.

Implementing transit signal priority can offer many challenges. One major concern is how to implement transit priority within the existing signal control system. Another is determining what impacts the priority implementation will have on other traffic. Most fundamental, however, is the question of what benefits the priority implementation offers and whether these benefits outweigh the costs.

Because passive priority strategies require no equipment other than the existing traffic controller hardware, these strategies can be implemented and tested relatively easily in the field. Implementation of active priority strategies, however, requires a significant hardware investment, including specialized detectors for transit vehicles and, in some cases, more advanced signal controllers. Field testing of active strategies, therefore, is often too costly to justify, especially when the benefits may be uncertain. In these cases, simulation can be used to evaluate a proposed strategy before it is implemented

determining whether field implementation will have beneficial results. Microscopic traffic simulation is an ideal tool for these evaluations, as it simulates vehicle movements at a detailed level, modeling interactions with other vehicles and response to traffic control devices.

1.1 Objectives

The objective of this research is to design and implement an advanced traffic signal controller within a microscopic simulation environment, thus creating a laboratory for the evaluation of advanced traffic control strategies, including transit signal priority. The simulation tool used for this research is MITSIMLab, a microscopic traffic simulation laboratory developed for ITS design and evaluation. The simulation in MITSIMLab is divided into two components, a microscopic traffic simulator (MITSIM) and a traffic management simulator (TMS). MITSIM simulates the road network and its vehicles, modeling vehicle movements by means of detailed driving behavioral models. TMS simulates the traffic control system of the network, modeling signal controls and advanced features such as route guidance and traveler information systems.

As part of this research, a new simulated traffic signal controller is implemented within TMS. The controller is designed with a generic and flexible logic that allows it to simulate a wide range of traffic signal control types and strategies. These control strategies include both isolated and coordinated intersection control, with fixed-time and demand-responsive logic. The controller is also designed with a modular structure that allows specialized features of advanced control strategies to be implemented within the controller framework. This framework is used to implement transit signal priority in MITSIMLab, allowing the simulation of both passive and active signal priority strategies.

The capabilities of the controller are illustrated through a case study in which MITSIMLab is applied to an urban arterial network in Stockholm, Sweden, where an existing signal priority strategy is implemented. An evaluation of the currently implemented system is performed, and recommendations for improvement and further study are offered.

1.2 Thesis Outline

The remainder of this thesis is structured as follows: Chapter 2 gives an overview of traffic signal control and transit signal priority concepts and reviews recent developments in the literature in the design and evaluation of signal priority strategies. Chapter 3 details the design and implementation of a generic traffic signal controller in MITSIMLab with the capability of simulating advanced signal control and priority strategies. Chapter 4 details the modeling of the PRIBUSS signal priority strategy within the framework of the generic controller. Chapter 5 presents a case study in which MITSIMLab is used to evaluate an existing PRIBUSS implementation in Stockholm. Finally, Chapter 6 presents conclusions from the research and further research directions.

Chapter 2

Background and Literature Review

2.1 Traffic Signal Control

Traffic signal controls are implemented for the purpose of reducing or eliminating conflicts at intersections. These conflicts exist because an intersection is an area shared among multiple traffic streams, and the role of the signal system is to manage the shared usage of the area. Signals accomplish this by controlling access to the intersection, allocating usage time among the various users. The logic for this allocation can vary from simple time-based methods to complex algorithms which calculate the allocation in real time based on traffic demand. This section gives an overview of traffic signal control concepts and defines terminology and basic control types and strategies.

2.1.1 Terminology

Because definitions of signal control terms can vary across different countries and different controller types, this section will establish a consistent terminology that will be followed throughout the thesis. There are essentially two distinct methods of specifying basic signal control logic. The method that is standard in the United States is based on “phases,” while the method standard in much of Europe is based on “signal groups.” (FHWA, 1996; EB Traffic, 1990).

In traffic signal operation, specified combinations of movements receive right-of-way simultaneously. A “phase” is the portion of the signal timing cycle that is allocated to one of these sets of movements. Each phase is divided into “intervals,” which are

durations in which all signal indications remain unchanged. In the U.S., a phase is typically made up of three intervals: green, yellow, and all red. A phase will progress through all its intervals before moving to the next phase in the cycle.

These definitions are illustrated using the example intersection shown in Figure 2-1. The intersection has three approaches and six possible movements, which are numbered as shown in the figure.

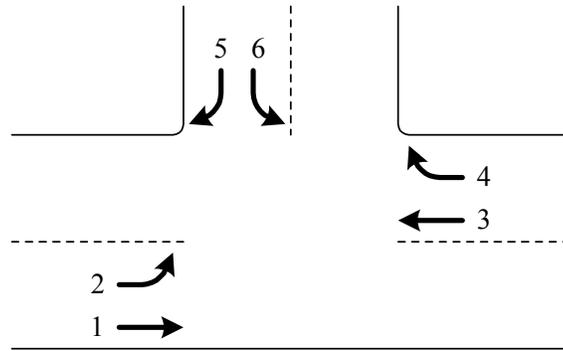


Figure 2-1: Example intersection.

A potential phase diagram for this intersection is shown in Figure 2-2. In this example, the cycle is divided into three phases. Movements 1, 3, and 4 are active in phase 1; movements 1 and 2 are active in phase 2; and movements 5 and 6 are active in phase 3. Each phase represents a distinct time period within the cycle, and in operation the controller moves from one phase to another in the specified order. The timing for the signal is defined by specifying the phase “splits,” which are the percentages of the cycle length allocated to each phase. This split time is further divided among the intervals of each phase, resulting in a specified duration for every interval in every phase.

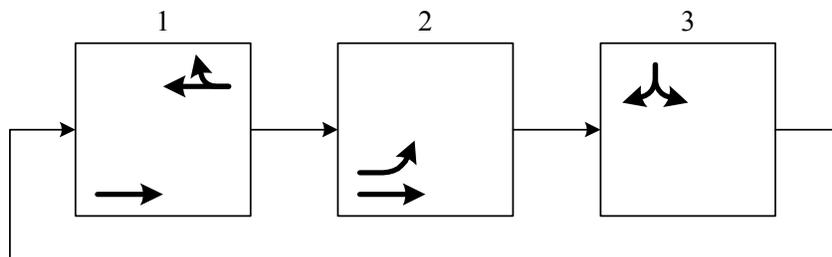


Figure 2-2: Example signal phase diagram.

The alternate specification is based on the concept of a “signal group,” which is defined as a set of signals that must always show identical indications. A signal group controls one or more traffic streams that are always given right-of-way simultaneously. The timing for a signal group is specified by “periods,” which are the durations in which the indication of that signal group does not change.

As an example, the same control logic shown in Figure 2-2 can be expressed in terms of signal groups, as shown in Figure 2-3. Although there are six intersection movements, only four signal groups are needed to represent the logic, because movements that always obtain right-of-way simultaneously can be controlled by a single signal group. Therefore, while movements 1 and 2 must be controlled by two separate groups, movements 3 and 4 can be controlled by a single group because they are never active independently of each other. The same applies for movements 5 and 6, which can also be controlled by a single group.

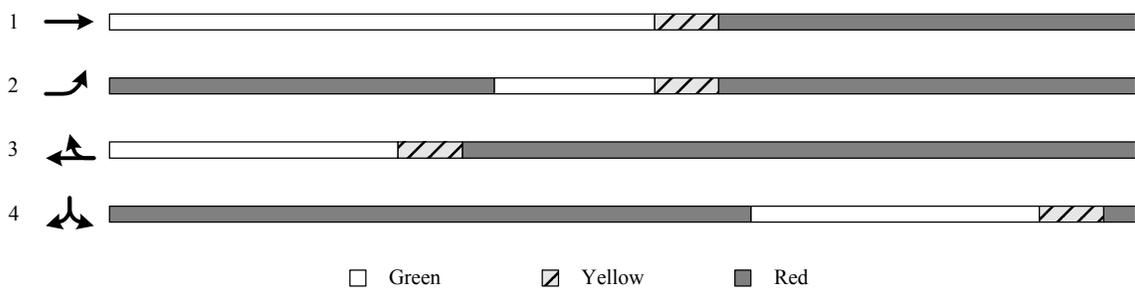


Figure 2-3: Example signal group diagram.

The timing of each signal group is represented by a horizontal bar whose length represents the cycle length. Each bar is divided into different segments that represent the different periods for each signal group. In this example, each signal group has three periods: green, yellow, and red. In operation, these signal groups advance in time independently, each group changing indication when it reaches a new period.

Although signal phases are not explicit in the signal group diagram, phasing can be inferred by reading the diagram vertically. The start of every green period corresponds to the start of a phase, and the time in which all signal groups remain in a single period corresponds to an interval. The correspondence between the two specifications for the above example is demonstrated in Figure 2-4.

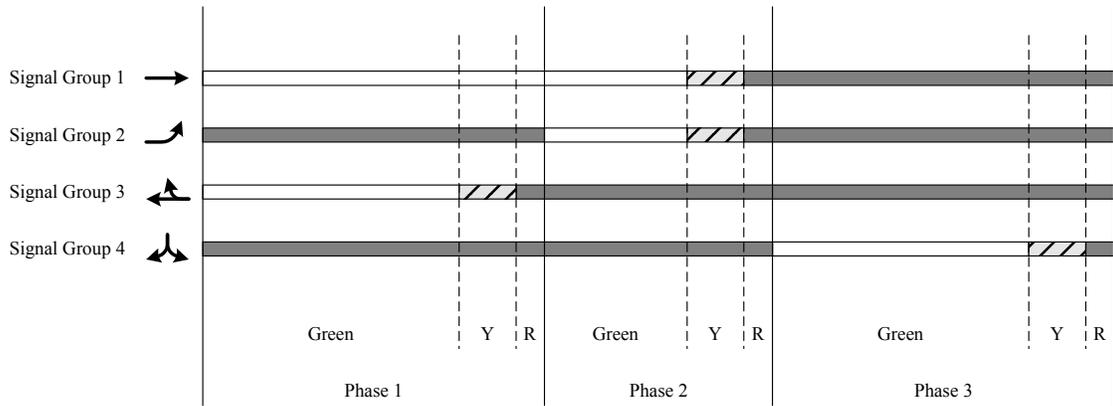


Figure 2-4: Relation between phase and signal group specifications.

2.1.2 Control Types

There is a wide range of logic by which signal phasing and timings can be controlled. Logic types can be categorized along two axes (FHWA, 1996). The first is the type of control logic, specifically how the controller responds to local traffic conditions. This logic can be pretimed, actuated, or adaptive. The second is the scope of the control strategy, i.e. over what area the strategy is applied. Possible strategies are isolated intersection control, arterial control, and network control. The diagram in Figure 2-5 shows the matrix of possible control types.

		Control Scope		
		Isolated Intersection	Arterial Coordination	Network Control
Control Logic	Pretimed	•	•	•
	Actuated	•	•	•
	Adaptive	•	•	•

Figure 2-5: Types of signal control logic.

Control Logic

Pretimed control is the most basic type of control logic that can be implemented. In pretimed control, the cycle length and the phase splits are set at fixed values, as are the

durations of each interval within each phase. Historical flow data is typically used to determine appropriate values for these parameters. The key attribute of pretimed control is that the logic is not demand-responsive, meaning that the signals operate without regard to fluctuations in traffic demand.

Actuated control uses demand-responsive logic to control signal timings, with phase durations set based on traffic demand as registered by detectors on the intersection approaches. The most common feature of actuated control is the ability to extend the length of the green interval for a particular phase. The interval might be extended, for example, when a vehicle is approaching a signal that is about to change to yellow, allowing that vehicle to pass through the intersection without stopping.

Figure 2-6 demonstrates how the green interval of a phase can be extended by vehicle actuation (Kell and Fullerton, 1982). Three parameters are required: the minimum green time, the extension time, and the maximum green time. Regardless of demand, green is retained for at least the specified minimum duration. If a vehicle is detected and less than the extension time remains in the interval, the interval is extended from the time of actuation by the length of the extension time. This can occur repeatedly, as shown in the figure, with the end of the interval delayed by the extension time from the time of each actuation. The interval will be terminated either when no additional actuation occurs during the latest extension time or when the specified maximum interval length is reached. The extension time is often referred to as the “gap time,” because the interval will be extended if a vehicle has a time gap (headway) from the vehicle in front that is less than this value.

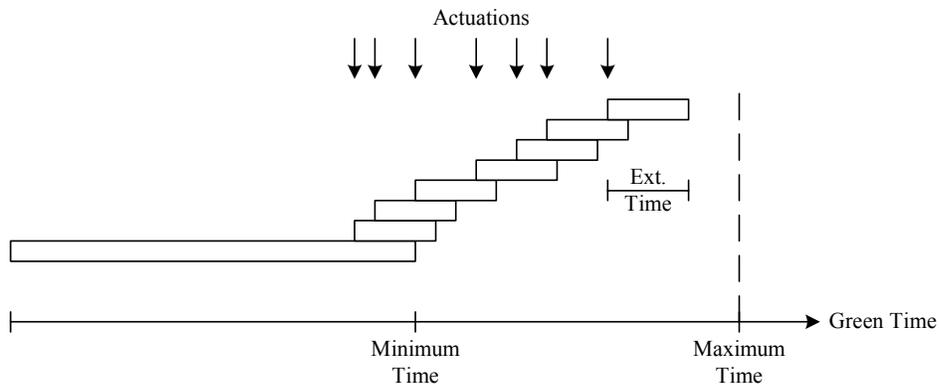


Figure 2-6: Green interval extension of an actuated phase.

The extension time is usually set to be the travel time from the point of detection to the intersection, as this will extend the interval for just enough time for a detected vehicle to be able to cross the intersection. However, the extension time can also be set to vary as a function of the elapsed green time, usually reducing the extension time as the maximum time is neared. A variable extension length is often used when detectors are located a long distance from the intersection, because a long extension time is desirable at the start of the phase to ensure that vehicles can cross the intersection, while a shorter extension is desired near the end of the phase so that the phase is not extended unnecessarily (McShane et al., 1990). A typical “gap-reduction” function is shown in Figure 2-7.

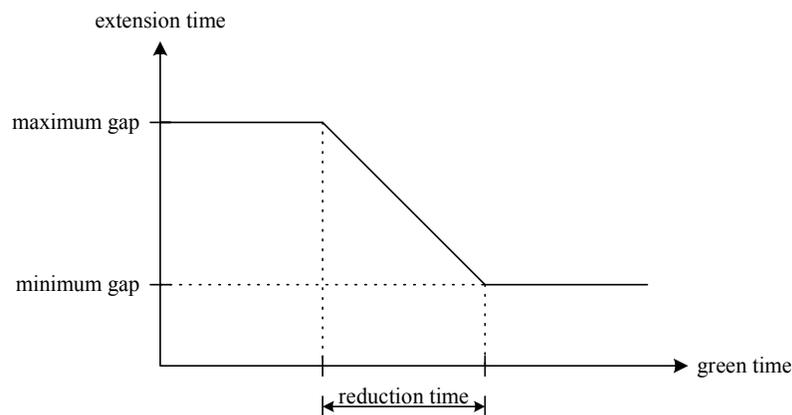


Figure 2-7: Example of a gap-reduction function.

Another common feature of actuated control is the ability to skip a phase if no demand for that phase is present. If there are no vehicles waiting for any movements of a certain phase (as determined by the detectors at the stop lines), the controller can skip over that phase and move directly to the next phase in the sequence.

Adaptive control, like actuated control, responds to traffic demand in real time, but its logic can change more parameters than just interval length. The most common adjustments made are to the cycle time and to the phase splits, which determine the allocation of the cycle time to the various phases. These strategies rely on traffic data collected for each approach upstream of the intersection, and this data is used by the controller to estimate conditions at the intersections and to respond to them in real-time. This logic is often optimization-based, allocating green time to maximize measures such

as vehicle throughput or to minimize measures such as vehicle delays or stops. Adaptive logic can also be predictive, projecting future conditions based on detector inputs and historical trends and adjusting signal settings accordingly.

Adaptive traffic control systems are becoming more widespread, both in application and in development. Urban traffic control systems such as SCOOT are implemented widely (Bretherton, 1996), and applications of systems such as OPAC and UTOPIA are also becoming more prevalent (Gartner et al., 1991; Peek Traffic, 2000).

Control Scope

Isolated intersection control is a control strategy in which the signals for one intersection are operated without consideration of any adjacent signals. In such a case, each intersection will have signal timings that are most appropriate for that single intersection. The local control logic can be pretimed, actuated, or adaptive; but in the case of demand-responsive logic, the controller will only consider local conditions immediately upstream of the intersection.

Arterial coordination is a strategy in which the interaction between adjacent signals is considered. The goal of such strategies is most often to provide “progression” through multiple intersections, allowing vehicles to move through successive signals without encountering a red signal. Figure 2-8 shows an example of how this can be achieved (Homburger and Kell, 1988). In this time-space diagram, the horizontal axis represents distance along an arterial, and the vertical axis represents time. The vertical bands represent three signals along the arterial with their indications displayed over time.

As shown in the diagram, the timing of the signals can be set such that a vehicle travelling at a certain constant speed can obtain green lights at each intersection. The green times at the signals create a “green band,” and vehicles whose trajectories fall within this band will be unimpeded by the signals. This result is achieved by setting each signal at a different “offset,” defined as the time difference between the start of the signal’s green interval and a system reference time. Setting the offset difference between adjacent intersections to equal the travel time between those intersections will establish progression. Arterial coordination can also be used to provide progression to both directions of traffic, as shown in Figure 2-9.

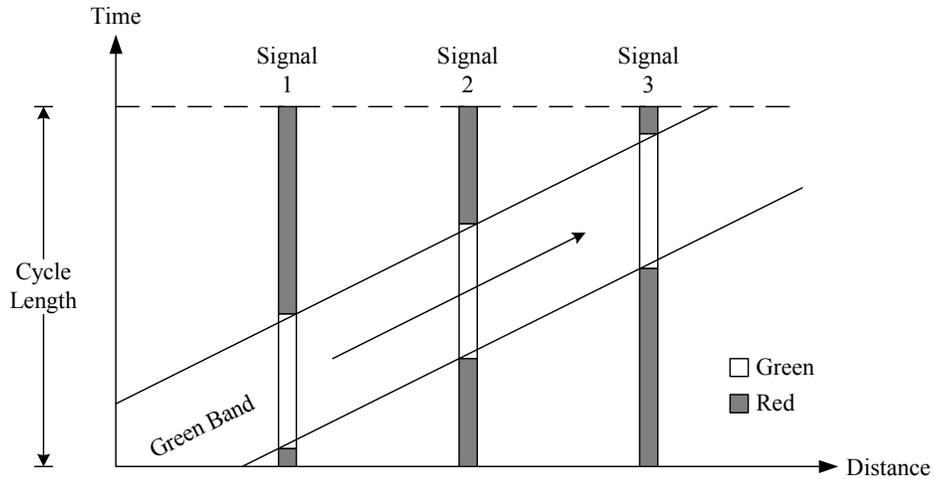


Figure 2-8: Progressive traffic flow under signal coordination

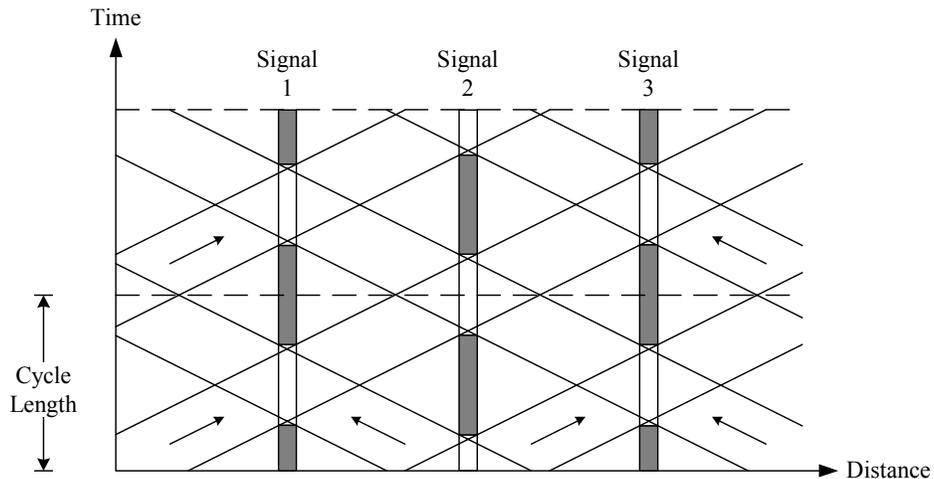


Figure 2-9: Bi-directional progressive flow under signal coordination

With pretimed signals, arterial coordination is established by using the same cycle length for all signals and by defining an appropriate offset for the green interval at each signal. The offsets define the green band, and the common cycle length ensures that the signals remain synchronized and that the green band will be present in each cycle.

Arterial coordination under actuated control operates on a similar principle, with fixed cycle lengths and offsets for the coordinated green intervals. However, the actuated control logic allows added flexibility, as the non-coordinated phases (such as those for cross streets or for left turns from the arterial) can be skipped or extended based on

demand. The coordinated phases, however, must always be green at a fixed time for a specified duration during each cycle in order to maintain the green band for progression.

Under adaptive control logic, arterial coordination can be implemented by optimizing measures such as travel time or stops on a corridor-wide level rather than on a single intersection level. By using inputs and measures from the entire corridor, a more efficient control strategy can be realized. For example, an adaptive control strategy might anticipate demand at one intersection based on the signal operation at an upstream intersection, predicting the arrival of a platoon of vehicles that has been released by the upstream signal.

Network control has the broadest scope of the control strategies, as it considers the performance of a network as a whole in the implementation of signal control. Most often, network control is an extension of arterial coordination that considers progression for all traffic in all directions of travel. An example of a system where network control can be effective is a urban grid network, in which often no direction of travel may be dominant. In this environment, both pretimed and actuated control can be easily used to provide limited progression in multiple directions. With adaptive control, however, consideration of network performance may exponentially increase the size of the optimization problem, and solving this in real time may be too computationally intensive. For this reason, adaptive network control algorithms and strategies are still very much under development (FHWA, 1996).

2.2 Transit Priority Strategies

The objective of transit signal priority strategies is the reduction of delay for transit vehicles at signalized intersections. The rationale for this special consideration of transit vehicles has its basis in the high capacity of the vehicles. Typically, traffic signals are timed so as to minimize the total delay to all vehicles at an intersection. However, minimizing vehicle delay may not be optimal if the passenger load of the vehicles is considered. For example, a 30-second delay to a crowded bus is clearly not equivalent to a 30-second delay to a single-occupancy vehicle. For this reason, a better metric to use may be total person delay instead of total vehicle delay, as this more accurately represents the impacts imposed on the users of the transportation network. Granting

priority to transit vehicles, therefore, is more likely to minimize total delay per person and maximize total person throughput.

Figure 2-10 illustrates how delay to a transit vehicle can be caused by a traffic signal in the absence of transit priority. The figure shows the trajectory of a transit vehicle plotted on a time-space diagram, and the horizontal band represents a traffic signal with its indication displayed over time. If the vehicle trajectory encounters a red traffic signal, delay accumulates until the light turns green and the vehicle can proceed. Signal priority strategies attempt to reduce delay in two ways: by reducing the probability of a transit vehicle encountering a red signal, and, if this does occur, by reducing the wait time until the green signal.

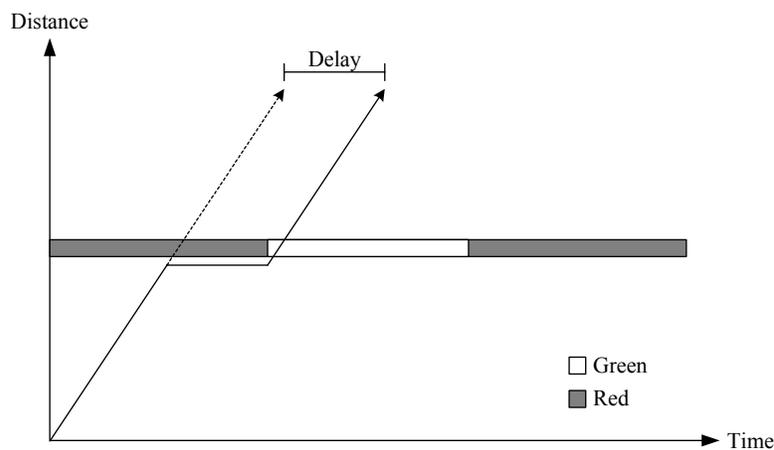


Figure 2-10: Vehicle trajectory without signal priority.

The literature on traffic signal priority falls into two general categories: description and development of signal priority strategies, and evaluation methodologies and results.

2.2.1 Passive Priority Strategies

Passive priority is defined as the use of static signal settings to reduce delay for transit vehicles. Such strategies can be as simple as allocating more green time to the street with the transit route by increasing the split for the phase in which the transit vehicle has right-of way. Because this reduces the percentage of the cycle during which the transit phase has a red signal, both the probability of a transit vehicle arriving during red and the average wait time if it does will be decreased (Garrow and Machemehl, 1998).

Another common passive strategy is the use of a shorter cycle length, which can reduce delay by shortening the wait time until the next green phase. However, this comes at the expense of reduced capacity for the intersection overall due to the increase in lost time, the time in each cycle during which no vehicle movements occur. Lost time typically results from the all-red safety clearance intervals between conflicting movements and from the vehicle startup delay at the beginning of each phase. Since lost time in each cycle is independent of the cycle length, a shorter cycle will mean that a higher percentage of the cycle is wasted as lost time. If an intersection is near saturation, this strategy may actually increase delays; but if excess capacity exists, this strategy can reduce delays to individual vehicles.

Split phasing is a related strategy in which the green phase for the transit corridor occurs twice within the same cycle. The cycle length can remain unchanged if each of the two green phases is half the length of the original phase. As when the cycle length is reduced, lost time is increased, but the increase will be smaller in this case because only one additional phase transition is added per cycle. This strategy benefits transit vehicles by reducing the amount of time between green phases, thus reducing the wait for vehicles encountering a red signal.

Signal coordination is another strategy that can be used to benefit transit vehicles. Arterial progression, for example, can be designed to favor transit vehicles by timing the green band at the average transit vehicle speed instead of the average automobile speed, which is typically faster. Although this strategy increases the travel time for automobiles, it helps ensure that transit vehicles can keep pace with the signal progression. However, progression for city transit vehicles may be difficult to maintain due to the presence of transit stops, which prevent those vehicles from moving at a constant rate through the network. Because the dwell time at transit stops is variable, static signal settings can not ensure proper progression.

A general problem with passive priority strategies is that they typically make the intersection operate less efficiently overall, especially if transit frequency is not very high. This is because the signal settings will be sub-optimal when transit vehicles are not present, which will be the case the large majority of the time. For this reason, such strategies may not always be feasible, especially under highly saturated conditions. In

such cases, using a shorter cycle length or larger transit splits may lead to over-saturation of the intersection, leading to long queues and delays.

Although passive priority strategies have definite limitations, in many cases they are the only viable options, especially when cost considerations require the use of existing traffic controller hardware. However, the amount of recent research into passive strategies is minimal and mostly general in nature (Skabardonis, 2000), reflecting the limited value of such strategies.

2.2.2 Active Priority Strategies

Active strategies address these limitations of passive strategies by altering signal settings dynamically and only when necessary, making adjustments in real-time to the signal timing in order to minimize delay to an approaching transit vehicle. This is more infrastructure intensive than passive priority, requiring devices to detect transit vehicles upstream of the intersection and advanced controllers to employ strategies for granting priority.

There are three basic actions that a controller can perform in response to the detection of a transit vehicle: extension of the green interval in the current phase, ending another phase early to give an early green to the vehicle, and inserting an extra phase to allow the vehicle to pass before returning to the regular timing. The response used will depend on when in the cycle the vehicle is detected.

If the vehicle is approaching the intersection near the end of the green interval for its approach, the current interval can be extended until the vehicle has passed through the intersection, as shown in Figure 2-11. Without extension, the vehicle would have to wait for green in the next cycle, leading to significant delay.

If the transit vehicle is approaching a red signal, the two other actions can be used. In the case where the vehicle will normally get a green in the next phase, the current phase can be ended early to allow the vehicle to get an early green. This is possible if the vehicle will arrive at the signal near the end of the red period for its approach, as shown in Figure 2-12.

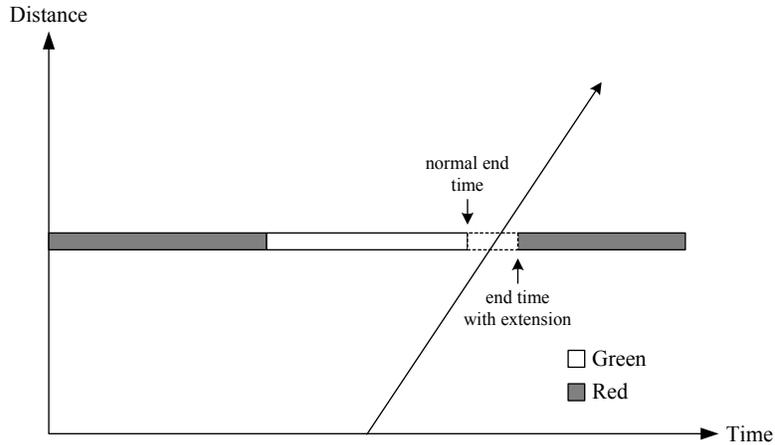


Figure 2-11: Vehicle trajectory with transit phase extension.

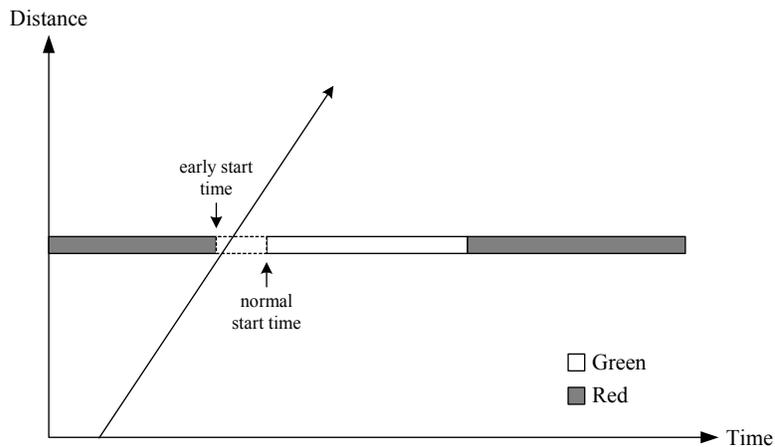


Figure 2-12: Vehicle trajectory with early start to transit phase.

If other phases need to be served before the normal return to green, a short phase for the transit approach can be inserted, with the controller returning to normal operation once the vehicle has passed. Such a case is shown in Figure 2-13, where the controller breaks from its normal plan to serve the transit phase before returning to the regular signal timings.

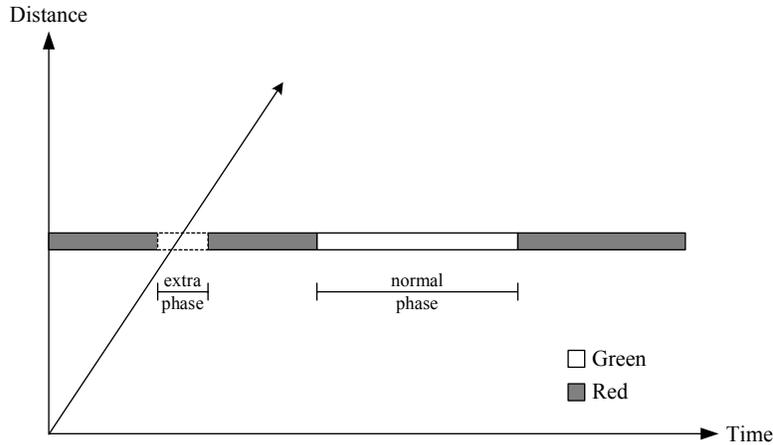


Figure 2-13: Vehicle trajectory with insertion of extra transit phase.

A major concern with active priority strategies is the effect they have on other traffic. Under light traffic conditions, active priority may have little effect on other traffic because excess capacity within the cycle can be redistributed to the transit phase. However, active priority can have major negative effects in peak period operations, when intersections are operating near saturation with little or no time to spare from the non-transit movements. Both simulation studies and field tests have demonstrated the detrimental effect on cross-streets, especially those near saturation (Garrow and Machemehl, 1998; Furth and Muller, 2000). Due to these effects, the system-wide value of active transit priority may only be worthwhile if transit has a high ridership in the corridor, causing the benefits per person to outweigh the costs.

Facility design can also present problems in implementing active priority strategies. For example, near-side transit stops (i.e. those placed just upstream of an intersection) complicate active priority because the green phase may be extended unnecessarily while the vehicle is held at the stop. Even if the dwell time is taken into consideration, because this dwell time is variable, the extension required will always be uncertain. For this reason, far-side bus stops are preferable when active priority strategies are employed.

Unlike for passive priority, much research is being devoted to the development and analysis of active priority strategies. These strategies fall into three general categories: unconditional, conditional, and adaptive.

Unconditional Strategies

An unconditional strategy is one which gives priority status to every transit vehicle detected, meaning that the signal controller will attempt to initiate one of the priority actions described above upon detection of any transit vehicle. The disadvantage of this strategy is that priority may be granted unnecessarily, such as to a vehicle that is ahead of schedule. However, unconditional priority requires no further information other than the presence of the vehicle to be transmitted to the signal controller, which makes it the only option for transit systems with limited communication capabilities.

Conditional Strategies

Conditional strategies grant priority status based on certain criteria, in most cases properties of the specific transit vehicle. The most common criterion for conditional priority is the lateness of the vehicle relative to its schedule. However, further criteria such as vehicle headway (i.e. the time between successive vehicles) or passenger load are being considered for future applications.

The advantages of conditional strategies over unconditional strategies are demonstrated in research by Furth and Muller (2000). In a field test in Eindhoven, the Netherlands, comparisons were made between conditional priority, unconditional priority, and the base case of no transit priority. While the unconditional strategy had the most reduction in travel time for the transit vehicles, moving to a conditional strategy led to major improvements in service to other vehicles with only a small sacrifice in operating speed to the transit vehicles. Other benefits of conditional priority cited were its ability to provide a means of operational control and improvements to schedule adherence.

Development of conditional priority strategies in recent literature focus on maximizing schedule adherence of buses while minimizing impacts on other traffic (Skabardonis, 2000). Strategies have also been developed for use in specific applications and under constraining external conditions. For example, a framework for integrating transit priority with arterial signal progression, developed by Vasudevan and Chang (2001) considers the schedule delay of the transit vehicle, delay caused by interruption of arterial progression, and vehicle queue lengths in the determination of the control

decision. Other strategies focus on incorporating bus priority into existing controller hardware and software, which can place significant restrictions on the priority implementation (Balke et al., 2000).

Adaptive Strategies

Adaptive transit priority strategies are those which use optimization-based control schemes to determine if and how to grant priority. In such schemes, the delay of the transit vehicle is considered along with the delay faced by all other vehicles. The controller then calculates the optimal solution for how to allocate time between the competing approaches. Because phases and timings are not fixed, adaptive strategies do not require predefinition of specific priority actions, such as phase extension or insertion, as the controller is constantly changing the allocation of green time based on demand.

Transit priority strategies can easily be implemented within most existing adaptive systems by giving more weight to transit vehicles in the optimization routine. For example, weighting transit vehicles by 50 will mean that the controller will treat the bus as if it were 50 cars, thereby favoring that vehicle's approach in the optimization (Peek Traffic, 2000).

However, implementing transit priority within existing adaptive control strategies has certain flaws (Duerr, 2000). A general problem is that adaptive control systems consider network-wide effects in their optimization, while providing transit priority is a local controller concern. This may lead to conflicting goals in the optimization and therefore sub-optimal results. Another problem is that most adaptive control systems use macroscopic models of traffic flow in their estimation and optimization routines, and these models can not capture certain details of transit vehicle movements. For example, dwell time at transit stops and interactions between the transit vehicle and other vehicles will not be considered, so travel time for transit vehicles may be underestimated. Finally, constraints on the optimization may limit the opportunity for transit priority, especially during peak hours when transit priority is most essential. For example, many systems have constraints on allowable queue lengths for the intersection approaches. During peak demand these constraints may be limiting, such that no extra time can be taken from other approaches and given to the transit movement. This may essentially eliminate the

possibility of transit priority under certain conditions, especially in peak conditions when priority is most needed.

Recent research has addressed these issues with the development of adaptive strategies that focus on transit priority at the intersection level (Chang et al., 1994; Duerr, 2000).

2.2.3 Evaluation of Priority Strategies

Evaluations of existing transit priority systems generally show the implementations to be effective in reducing delays for transit vehicles. The recent implementation of a transit priority system in Los Angeles, for example, is cited as providing an 8-10% reduction in travel time on the lines equipped with priority, with “minimal” adverse impacts on cross-street traffic (Hu et al., 2000). The evaluation in Eindhoven (Furth and Muller, 2000) goes further by field-testing different strategies on a network with an established transit priority system. The primary measure of effectiveness in this evaluation is intersection delay, aggregated both for buses and for other vehicles. Impacts on other vehicles are broken down by approach, as cross-street traffic is impacted negatively while through-street traffic, which shares right-of-way with the priority-equipped buses, benefits. Bus delays were reduced from an average of 27 seconds with no priority to an average of three seconds under unconditional priority. However, this comes at an average cost of 40 seconds per vehicle to other traffic under peak conditions. Conditional priority, while not offering such large reductions in delay for the buses, essentially eliminated the delay for other vehicles. Schedule adherence was another measure of effectiveness used for evaluation. Conditional priority was found to have a significant effect, reducing schedule deviations caused both by traffic conditions and by imprecise dispatch control.

For design and development of transit priority strategies, however, field-tests are usually infeasible. Microscopic traffic simulation is usually the most viable alternative for testing designs before field implementation. Most research that develops new strategies relies on micro-simulation for evaluation purposes (Balke et al., 2000; Garrow and Machemehl, 1998). The most common metric used for such evaluations is travel time through the network, as this is the most direct measure of the impact of the control

strategy. Often this is translated into delay, or time lost in intersection queues. The impact on transit vehicles is usually separated from the impact on traffic as a whole in evaluation studies, allowing the benefits to transit vehicles to be contrasted with the costs to negatively impacted vehicles. Microscopic traffic simulation is an ideal tool for such evaluations, as detailed records of individual vehicle performance can be gathered.

A framework for impact assessment studies of transit priority developed by Dale et al. (1999) seeks to provide a consistent methodology for evaluation. The nine measures of effectiveness selected for evaluation are total intersection delay, minor movement delay, minor movement “cycle failures” (i.e. the number of vehicles which must wait for more than one cycle length), bus travel time, bus schedule reliability, bus intersection delay, intersection delay per person, vehicle emissions, and accident frequency. Although these criteria are mostly oriented toward field evaluation studies, it is recognized that future evaluations are likely to rely more on simulation studies in order to reduce evaluation costs. While the authors cite limitations of simulation, including questionable accuracy in the replication of actual conditions and distrust of simulation studies by stakeholders, the benefits of reduced cost, diminished risk, and greater control over the study lead to the conclusion that the use of simulation is a valuable and cost-effective for transit priority evaluation.

2.3 Summary

Traffic control strategies can vary both in their type of logic and in the scope over which they are applied. The control logic determines how the controller responds to traffic on a local level, while the control scope determines the area considered in the making of control decisions. Transit priority is not tied to a particular control type. Instead, it is a strategy that is implemented within an existing control system. Passive priority strategies, which employ static signal settings that favor transit vehicles, are easily implemented in the field with existing hardware. Active priority strategies, however, require specialized transit vehicle detectors and controller hardware that can respond to detected vehicles in real time.

Although a wide range of research into active signal priority strategies is being conducted, the same general concepts are at the root of most strategies. With traditional

signal controllers, the actions that can be undertaken in response to a detected vehicle are quite limited. These include extension of the current green interval, starting the green interval early, and inserting an extra phase for the transit vehicle. Most priority strategies, therefore, only differ in the criteria they use to decide when to grant priority. Adaptive controllers allow a more flexible logic to be defined, but conflicts between the local objective of providing priority to a transit vehicle and the network objective of optimizing traffic flow can be problematic.

In order to simulate transit priority strategies, the logic of the underlying traffic signal control system must be simulated as well. Because these control types can vary widely, the simulated signal controller must be flexible enough to model these different systems in addition to the transit priority strategy to be implemented. The research in this thesis aims to design and implement such a controller in a microscopic simulation environment, thereby creating a laboratory for the design and evaluation of transit signal priority strategies.

Chapter 3

Generic Traffic Signal Controller: Design and Implementation

This chapter describes the design and implementation of a simulated traffic signal controller within a microscopic traffic simulation laboratory. The controller is designed to be flexible enough to model a wide range of control types while also providing a framework for implementation of advanced control strategies. This chapter details the logic of the controller and illustrates its control capabilities with example specifications for various control strategies.

3.1 MITSIMLab

MITSIMLab (Yang et al., 2000) is a microscopic traffic simulation laboratory that has been developed as a research tool for design and evaluation of ITS applications. The laboratory is based on the interaction of its two core simulation modules, a microscopic traffic simulator (MITSIM) and a traffic management simulator (TMS), simulating the dynamic interaction between the traffic management system and drivers on a network.

MITSIM simulates the road network and its traffic, modeling movements of individual vehicles with driving behavior and travel behavior models. The driving behavior models represent decisions about acceleration and lane-changing that drivers make based on interactions with other vehicles and in response to traffic control devices. The travel behavior models represent decisions such as route choice, including response

to route guidance information. TMS simulates the traffic control system on the network, including elements such as traffic signals, ramp meters, freeway control systems, lane-use signs, variable message signs, and in-vehicle route guidance.

The interaction between these modules is shown in Figure 3-1. TMS receives surveillance data from MITSIM and generates the control and routing strategies to be implemented. These strategies determine the status of control devices in MITSIM, and drivers in the simulation respond to the updated states of these devices. A graphical user interface (GUI) provides visual output of the simulation results via animation of the vehicle movements on the network, supplementing the detailed output records provided by MITSIM.

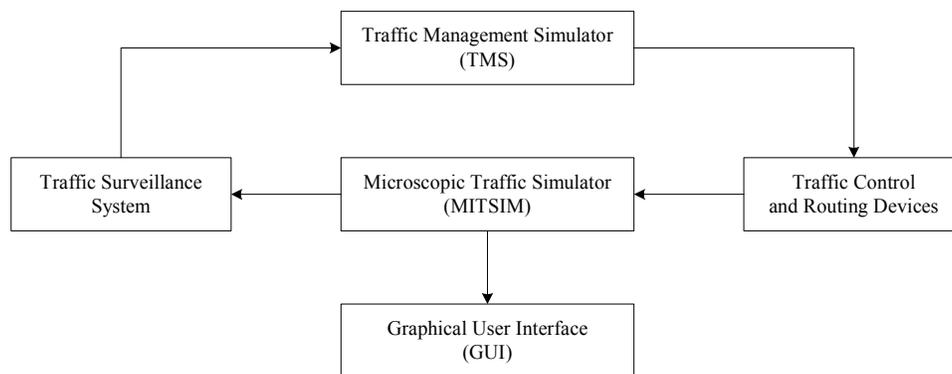


Figure 3-1: Elements of MITSIMLab and their interactions.

The pre-existing implementation of traffic signals in TMS is limited to pretimed and simple actuated controllers, and is therefore not adequate for simulating advanced control strategies such as transit priority. The primary limitation of traffic signals in TMS is the use of fixed phase plans and phase orders, which makes it difficult or impossible to model the more flexible operation that is required by advanced strategies. Another restriction is that coordinated signal operation is only possible using pretimed logic, which prevents the use of any extension or priority functions in corridors with arterial progression.

The phase-based design of the TMS signals also creates problems in non-U.S. applications, where control logic is more often specified in terms of signal groups. In the signal group specification, phase plans are not necessarily explicit and each signal group can have its own independent logic. Modeling such controllers with a phase-based

approach is difficult and in some cases impossible. A new simulated controller that overcomes these limitations has been implemented within TMS and is described in the following sections.

3.2 Controller Design

The newly implemented signal controller in TMS has been termed the “generic” controller because it has been designed to be able to simulate the widest possible range of signal controllers. The goal was to design a controller that could be used for any future applications, regardless of the specific control system in use. This is especially important because MITSIMLab is applied in multiple countries with different signal controller standards.

Clearly, it is impossible to capture the precise details of every possible controller type, but all controllers have a common base of core functions that can be identified. The approach of the generic controller is to break down control strategies into these basic logic elements and to implement these elements within a modular framework. Specific control logic can then be recreated from these basic components, and the modular structure allows any specialized features to be implemented easily.

3.2.1 Basic Logic Elements

Regardless of the complexity of the control system in use, the output of the control logic that is seen by a driver is extremely simple: a color indication on a signal face. Therefore, when a control logic is viewed on an individual signal group basis, the logic reduces to the single decision of whether to hold the current indication or to change to a different one. Input variables for this logic are also relatively limited. These include time-based inputs, data from sensors on the network, and data from other signals and controllers.

The most basic logic elements are time-based, either relative to a local timer or to a system clock time. A local timer is used if a signal has restrictions on the duration for which an indication is displayed, such as minimum and maximum lengths. For example, the logic may specify that a green signal will be held if it has been active for less than six seconds and will be forced to change to yellow if it has been active for more than 30

seconds. Alone, these logic units can simulate a pretimed controller by specifying the duration of each interval for every signal. Clock time is used as an input if actions must occur at specific times, such as when signals are running in coordination. In these cases, start and end times for signal intervals may be specified relative to a system clock.

More advanced logic elements use sensor readings from the network as inputs and perform different actions depending on those values. For example, a vehicle detected on a side-street approach may force the primary-street signal to change from green to yellow in preparation for serving the newly arrived vehicle. In another example, a signal group may skip its green interval if detectors on its approach are unoccupied.

States of other signals in the intersection can also be used as logic inputs, allowing the definition of complementary or conflicting signal groups or the specification of a phase plan. For example, one signal group may be held in green as long as another complementary group is still green. In another case, a signal group may be held in red if a conflicting group is active, preventing conflicting movements from being allowed simultaneously. Moreover, by defining conflicting movements and identifying which signal groups wait for which other groups, an implicit phase order can be defined.

Just as the examples described above can be combined to form a description of a control strategy, the basic logic elements can be combined to construct a full control logic. The generic controller models these basic elements and provides an interface to allow these components to be ordered and combined.

3.2.2 Structure

The generic controller has been implemented in TMS as a new controller type and, like the other controllers, specifies the signals under its control. Unlike the existing controller types, however, the logic for the generic controller is specified in terms of signal groups instead of signal phases. This distinction is very significant. Specifying logic in terms of signal phases divides the signal cycle into distinct time periods. In operation, the controller progresses in time, displaying phases in a specified sequence. By specifying logic in terms of signal groups, the signal cycle is divided by groups of vehicle movements. In operation, these signal groups progress individually in time, changing states when specified. Specification in terms of signal groups allows much

greater flexibility in defining the controller logic, as all the possible phases need not be enumerated.

For this reason, the signal group is used as the structural unit of the generic controller. Each group is defined by the intersection movements that it controls and by the logic that governs its operation. In the generic controller implementation within TMS, the signal group holds data about its current status and its relationship to other groups. This information includes its current indication (e.g. green arrow), its current action (e.g. holding the current period), the next indication to show (e.g. yellow arrow), its conflicting groups, and stored sensor data. This information also included time-based data, such as the initialization time of the signal group, the start time of the current period, and the time since the last vehicle actuation if subject to actuated control.

A signal group also has a status flag variable which provides more information about the current status, such as whether the group is active, completed, waiting for another group, or skipped. This status flag can be read by the controller and by other signal groups whose logic uses the status of that particular group as an input variable.

The parameters and logic for each signal group is specified using a set of conditions, equivalent to the basic logic units described above. The following section describes the controller logic in detail.

3.2.3 Logic

In the MITSIM traffic simulation module, the status and position of every vehicle is updated at a specified step size, typically 0.1 seconds. A similar approach is used for the generic controller, which evaluates each signal group at every time step and determines if the state of any group is to be updated. An overview of the logic of the generic controller is shown in Figure 3-2.

The first step is the initialization of the controller, in which the parameters for the controller are read in from the signal input file. These parameters include general data—the IDs of the signals that the controller directs, the movements controlled by each signal group, and the initial state of each signal group—as well as the conditions which specify the control logic for each signal group. As part of this step, each group is also set to its specified initial state.

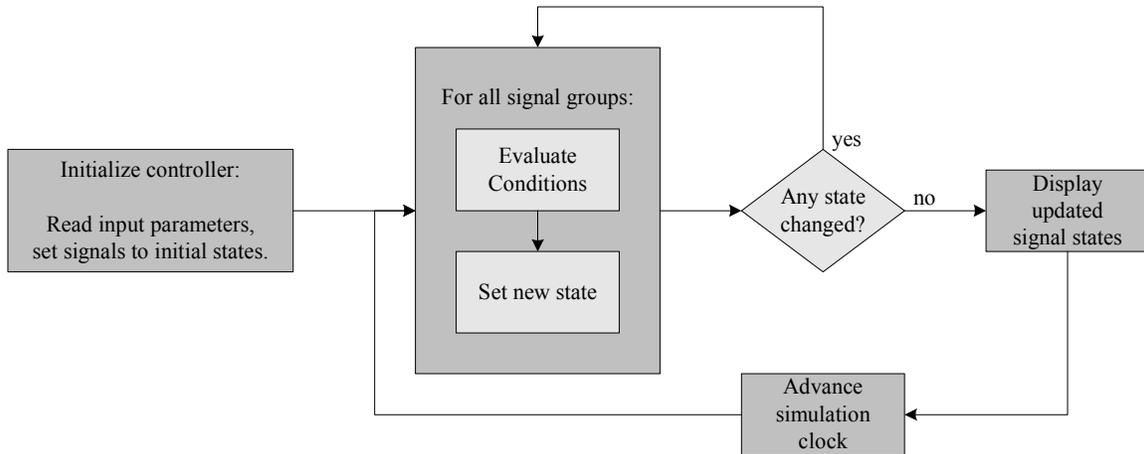


Figure 3-2: Overall logic of the generic controller.

Next the controller iterates through all the signal groups. For each signal group, it evaluates the logic conditions and determines whether the group’s state should be updated. Because the group states may be interdependent, with the state of one group as an input to the logic of another group, this evaluation step must be iterative. Therefore, when the state of all the signal groups has been determined, the controller again cycles through the groups to check if the new states cause more signal groups to change. This process is repeated until the states of all signal groups are stable, at which point the controller displays the updated states of the traffic signals in TMS. This process is repeated each time step, typically 0.1 seconds of simulation time.

This iterative process is required because the controller evaluates the signal groups in the order in which they are defined, an order that may be arbitrary. In order for the evaluation order not to affect the control logic, this iteration step allows all signal groups to be updated to their final states before the clock time advances. With properly defined logic, the signal group states will stabilize after a number of iterations, the updated states will be displayed, and the clock time will be advanced.

The most critical part of this process is the evaluation of the conditions that may apply for each signal group. The conditions are evaluated in the order in which they appear in the signal input file. Unlike for the signal groups, the order in which the conditions are listed is significant, as this is an integral part of the definition of the control logic. There are four types of conditions, corresponding to different actions that

each produces: *general* conditions perform miscellaneous functions, *change* conditions advance the signal group to the next period, *hold* conditions keep the group in the current period, and *skip* conditions indicate that the next specified number of conditions should be skipped. Figure 3-3 shows the detailed steps for the evaluation of conditions for each signal group.

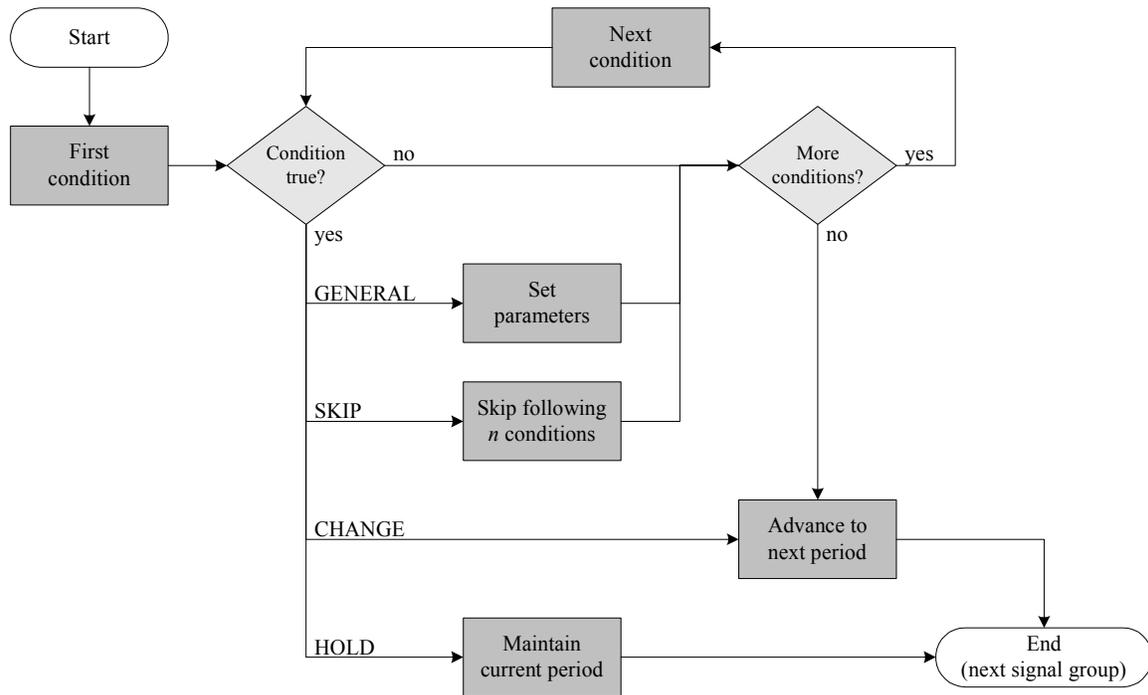


Figure 3-3: Condition evaluation logic within signal group of generic controller.

General conditions are typically listed and evaluated first, as they set various parameters and perform calculations that must occur every time step. For example, a gap timer which indicates the time since the last vehicle detection can be implemented as a general condition, as this timer must be either incremented or reset at each time step.

Change conditions specify the conditions that will force the signal group to change to a specified state. For example, a signal may be forced to change indication after a specified maximum duration or, if operating in coordination, at a specific clock time. If a change condition is met, the new state will be displayed and the remaining conditions will be skipped.

Hold conditions specify the conditions that will keep the signal group in its current state. For example, a signal may be prevented from changing if it has been active for less

than a minimum duration, if specified detectors are occupied, or if conflicting movements are active. If a hold condition is met, the current state will be maintained and the remaining conditions will be skipped.

Skip conditions are used to create more advanced logic using combinations of conditions. If a skip condition is met, the next specified number of conditions will be skipped. This type of condition can be used to specify an alternate logic for different circumstances. For example, a group may have a minimum time of 15 seconds under normal conditions but a minimum time of only eight seconds when demand is present on other approaches. A skip condition can therefore be used to ignore the longer minimum time when specified detectors are occupied.

If all conditions have been evaluated and no change or hold conditions have been met, the signal group will advance to the next period by default. This means that a hold condition must be valid in order to keep the group in the current period. The default order of the periods for each group is based on the standard intervals of green-yellow-red used in the United States and many other countries. Therefore, with the default settings, green is followed by yellow, yellow is followed by red, and red is followed by green. By using a general condition, however, an alternate next period can be specified for a particular period in order to model alternate interval progressions. For example, to model a starting red/yellow indication as standard in most European countries, red/yellow would be defined as the next period after red, and green would be defined as the next period after red/yellow.

3.2.4 Conditions

This section describes the common conditions that can be used for constructing the logic of each signal group. A table of these conditions and their input parameters can be found in Appendix A.

General conditions

- *Next Period:* This condition specifies the default period to follow the current period. This condition is only required when an order other than green-yellow-red is used.

- *Delayed Start (Offset)*: This condition delays the initialization of the signal group for a specified time offset. This condition is used to start the group at the proper time for coordinated operation.
- *Advance Start*: This condition starts the group a specified amount of time into the initial period, serving as an alternate method for specifying an offset.
- *Gap Timer*: This condition defines the detectors that will reset the gap timer, whose value is the time since the last vehicle actuation. This timer is used for actuated control, in which the detection of a vehicle may extend the current period.

Change conditions

All change conditions have “current period” and “next period” as parameters. The condition is only considered when the signal is in the “current period.” If the conditions are met, the signal will change to the “next period.”

- *No Demand*: This condition will force a change if none of the specified detectors is occupied at the beginning of the period. This condition is used under actuated control when a phase can be skipped if demand is not present. In the signal group specification, this is achieved by changing the green signal immediately to red, thereby skipping the green and yellow periods.
- *Force-Off*: This condition will force a change at a specified time within each cycle. As this condition is used under coordinated operation, the offset and cycle length are input parameters along with the force-off time.
- *Maximum Time*: This condition will force a change when the period has been active for longer than the specified duration.
- *Follow Other Group*: This condition will force a change a specified length of time after another group has changed to a particular state. This condition can be used to define a group that mirrors the operation of another group, either with or without a time lag.

Hold conditions

All hold conditions have “current period” as a parameter. The condition is only considered when the signal is in the “current period.”

- *Minimum Time*: This condition will hold the period if the period has been active for less than the specified duration.
- *Extension Time*: This condition will hold the period if the value of the gap timer is less than the specified extension length. This condition is used to extend a period under actuated control. A maximum length for the period is also specified. Alternate specifications of the extension condition allow the extension length to vary as a function of the time since the beginning of the period, as shown in Figure 2-7.
- *Complementary Groups*: This condition will hold the period passively if other specified groups are still being held actively, such as by extension due to demand. This condition is used when groups are constrained to end simultaneously.
- *No Conflicting Calls*: This condition will hold the period if none of the specified detectors is occupied. This condition can be used to define a resting phase, which is the set of signal states displayed when no demand is present at the intersection.
- *Conflict Clearance*: This condition will prevent the group from changing periods until all conflicting groups are no longer active and a specified time delay has passed. This condition defines conflicting movements, the all-red clearance interval between phases, and an implicit phase order.
- *Hold in Window*: This condition will hold the period while the time in the cycle is within a specified window. In addition to the start and end times of the window, the cycle length and an optional offset time are input parameters.
- *Hold Indefinitely*: This condition will hold the period if no other condition supersedes it.

Skip conditions

Skip conditions, if met, will skip the next specified number of conditions listed in the input file. If the condition is not met, all following conditions will be evaluated normally.

- *Detector Occupied*: This condition is met if any of the specified detectors is actuated.
- *Detector Not Occupied*: This condition is met if none of the specified detectors is actuated.

- *Probabilistic Skip*: This condition is met with a specified probability. This condition can be used to model probabilistic events not explicitly modeled in the simulation, such as calls on pedestrian signal groups.
- *Status Query*: This condition is a generalized skip condition that queries the status of a specified signal group. This condition will be met if a specified flag of this group is either set or not set, depending on an input switch. The status variables that can be queried include whether the group is active (not red), is passive (not being held), or was skipped.

3.3 Control Logic Capabilities

By combining the conditions above in a specific order, a full controller logic can be specified. The types of control strategies that can be simulated include isolated controller operations (both fixed-time and demand-responsive) and coordinated operation (also both fixed-time and demand-responsive). Adaptive control strategies have not yet been implemented, but a framework for incorporating such logic has been established. This section will give input specification examples for each type of control strategy.

3.3.1 Pretimed Control

Isolated Intersection Control

In isolated pretimed control, timings for all periods and groups are fixed. The inputs can be specified in two ways: by phases or by signal groups. In the signal group specification, each movement has specified durations for its green, yellow, and red periods. As shown in Table 3.1, this can be represented by minimum times for each period. Because different groups will have different starting times, an extra general condition which specifies the offset between the groups will also be required in order to have the timings synchronized.

Table 3.1: Pretimed-isolated logic specification (by signal groups).

Period	Condition	
	Type	Description
	General	Delayed start
Green	Hold	Minimum time
Yellow	Hold	Minimum time
Red	Hold	Minimum time

In the phase-based specification, green, yellow, and all-red times are given for each phase as well as the order in which the phases are displayed. Table 3.2 shows how this translates into conditions for the generic controller. The green and yellow periods have minimum times defined, while the red period uses the conflict clearance condition, which specifies the group it follows and the all-red clearance interval.

Table 3.2: Pretimed-isolated logic specification (by phases).

Period	Condition	
	Type	Description
Green	Hold	Minimum time
Yellow	Hold	Minimum time
Red	Hold	Conflict clearance

Coordinated Control

Pretimed-coordinated control relies not on interval duration but on clock time. This allows synchronization with other controllers in order to create the green band for arterial progression. Table 3.3 shows one method of specifying the timing in which the end times for all intervals are specified using force-off conditions. These conditions take the force-off time, cycle length and, optionally, an offset time as parameters. The “hold indefinitely” conditions are required to prevent the group from moving to the next phase by default before the change condition becomes valid.

An alternate specification based on phases is shown in Table 3.4. It is identical to the pretimed-isolated specification in Table 3.2 except for the green hold condition. Instead of the duration of the green being specified, it uses the start and end times in the cycle as reference points, holding the period in green between those times.

Table 3.3: Pretimed-coordinated logic specification.

Period	Condition	
	Type	Description
Green	Change	Force-off
	Hold	Hold indefinitely
Yellow	Change	Force-off
	Hold	Hold indefinitely
Red	Change	Force-off
	Hold	Hold indefinitely

Table 3.4: Pretimed-coordinated logic specification (alternate).

Period	Condition	
	Type	Description
Green	Hold	Hold in window
Yellow	Hold	Minimum time
Red	Hold	Conflict clearance

3.3.2 Actuated Control

Isolated Intersection Control

Isolated actuated control, at a minimum, requires detector states as inputs and logic for extension of the green interval. As shown in Table 3.5, these inputs are in the form of the gap timer and extension time conditions. The gap timer condition specifies which detectors will reset the timer and thereby extend the interval. The extension time condition specifies the extension logic, namely the maximum time gap allowed between successive vehicles and the maximum length of the extended period.

Table 3.5: Actuated-isolated logic specification (basic).

Period	Condition	
	Type	Description
	General	Gap timer
Green	Hold	Minimum time
	Hold	Extension time
Yellow	Hold	Minimum time
Red	Hold	Conflict clearance

A specification including more advanced features is shown in Table 3.6. One addition is a change condition which allows the green and yellow intervals to be skipped

if no demand exists. The second addition is the “complementary group” condition, which will be evaluated once the minimum time and extension time conditions are no longer valid. This condition will hold the group passively in green as long as the complementary group is being held, constraining the groups to end together. The third addition is the “no conflicting calls” condition, which directs the controller to hold this group in green if no calls are present on conflicting groups (as indicated by the detectors). This indicates that this group is active during the controller’s resting phase.

Table 3.6: Actuated-isolated logic specification (advanced).

Period	Condition	
	Type	Description
	General	Gap timer
Green	Change	No demand
	Hold	Minimum time
	Hold	Extension time
	Hold	Complementary group
	Hold	No conflicting calls
Yellow	Hold	Minimum time
Red	Hold	Conflict clearance

Coordinated Control

Actuated-coordinated control overlays the clock-time constraints required for coordination on top of the extension logic of actuated control. Phases can be extended, for example, but coordination may require the next phase to start at a specified time in order to maintain the green band for arterial progression. The logic specification, therefore, is simply the actuated-isolated specification with an additional force-off condition, as shown in Table 3.7. This force-off condition is needed to ensure that the next group will start no later than the time required for maintaining coordination. If the specified time in the cycle is reached and this group is still in green, this condition will force the signal to change to yellow in preparation for the next phase. This condition can also be used with the advanced features of isolated actuated control shown in Table 3.6, allowing non-coordinated cross-street phases to be skipped, for example.

Table 3.7: Actuated-coordinated logic specification.

Period	Condition	
	Type	Description
	General	Gap timer
Green	Change	Force-off
	Hold	Minimum time
	Hold	Extension time
Yellow	Hold	Minimum time
Red	Hold	Conflict clearance

3.3.3 Adaptive Control

Adaptive control strategies have not yet been implemented within the generic controller, but the generic controller provides a framework for their future incorporation. Implementation can occur in two ways, either by direct coding of the logic within the generic controller or with a link to an external logic module. Direct coding of adaptive logic would require the addition of new conditions with the more advanced logic. For example, instead of being a fixed input parameter, force-off times could be calculated dynamically based on sensor input, similar to the way extension lengths are calculated.

Linking TMS with an external logic module might be preferred if the logic is extremely complex or in cases where the details of the logic are proprietary information. This module would receive data from TMS, such as detector data and the current signal states, perform calculations internally, and then output settings back to TMS to be implemented. Such settings could be split, cycle, and offset values which the individual generic controllers then implement, or they could be direct control orders such as force-off times for each period. In the case of network control strategies, the individual generic controllers in TMS may still be required to manage phase transitions and any locally implemented strategies such as signal preemption.

3.4 Supported Controller Types

The generic controller can be used to simulate various existing traffic signal controllers, and input file templates have been created which replicate the features of specific controller types. These include the NEMA and Model 170 controllers, which are

standard in the United States, as well as standard European controllers that specify logic based on signal groups.

3.4.1 NEMA/Model 170

NEMA and Model 170 controllers, which are the standard traffic signal controllers used in US applications, have similar functionality. Their primary difference is that NEMA is a functional standard while Model 170 is a hardware standard with functions implemented by means of installed software modules. Operationally, however, both controllers are based on phase diagrams that define the allowed phases and the order in which they can be displayed. An example of a simple phase diagram is shown in Figure 3-4. In this four-phase example, the east-west movements are served first, with the left turns in Phase 1 and the through and right movements in Phase 2, followed by the left and through/right movements for the north-south street in Phases 3 and 4, respectively.

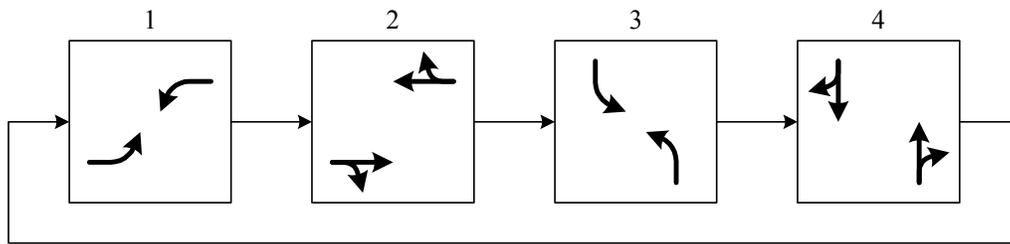


Figure 3-4: Four-phase controller diagram.

Modeling the behavior of the four-phase controller with the generic controller requires the definition of four signal groups corresponding to the four phases. Each group will have logic specified for its green timing, which may be pretimed or demand-responsive. The phase order is specified using the “conflict clearance” condition, which will hold a group in red until another specified group has been completed. In this example, Group 2 will wait for Group 1, Group 3 for Group 2, Group 4 for Group 3, and Group 1 for Group 4, thus defining the proper phase order.

An eight-phase controller that allows more flexibility in the phase progression is more commonly used, however, as it supports fully-actuated control. Figure 3-5 shows the typical phase diagram for such a controller. The eight-phase controller operates two four-phase rings simultaneously, allowing the rings to advance independently. For example,

the controller will start by displaying Phases 1 and 5, and then each phase can advance to its following phase when it is ready. This is possible because neither of Phases 1 or 2 conflict with Phases 5 or 6, meaning that the phase pairs of 1 & 5, 1 & 6, 2 & 5, and 2 & 6 are compatible and thus can be active simultaneously. A restriction in the independence of the rings is imposed by the barrier that separates the primary street and cross-street movements. Because all movements on one side of the barrier conflict with all movements on the other side, the rings must advance simultaneously across the barrier to prevent conflicting movements from being active simultaneously.

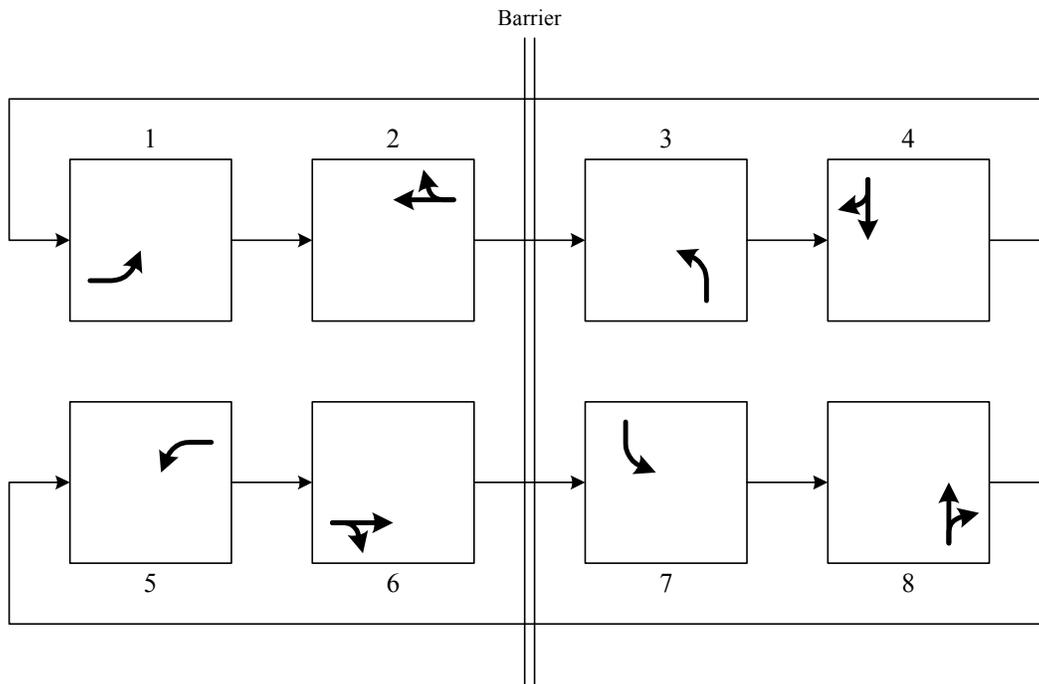


Figure 3-5: Eight-phase (dual-ring) controller diagram.

Figure 3-6 shows the different phase sequences possible with the eight-phase controller. The phasing is similar to the four-phase controller with the addition of alternate transition phases where left and through movements may operate concurrently. Which transition phase used, if any, will depend on the ending times of Phases 1 and 5, determined either by preset timings or by vehicle actuations.

With the generic controller, this can be modeled with eight signal groups, each having a preceding group defined to specify the phase order. The barrier is modeled with the “complementary group” condition, which will hold a group passively in green while

another group is still active. In the example above, Groups 2 and 6 are defined as complementary. This condition constrains their green intervals to end at the same time, assuring that the phase rings cross the barrier simultaneously. Groups 4 and 8 are similarly defined, modeling the return across the barrier to Phases 1 and 5.

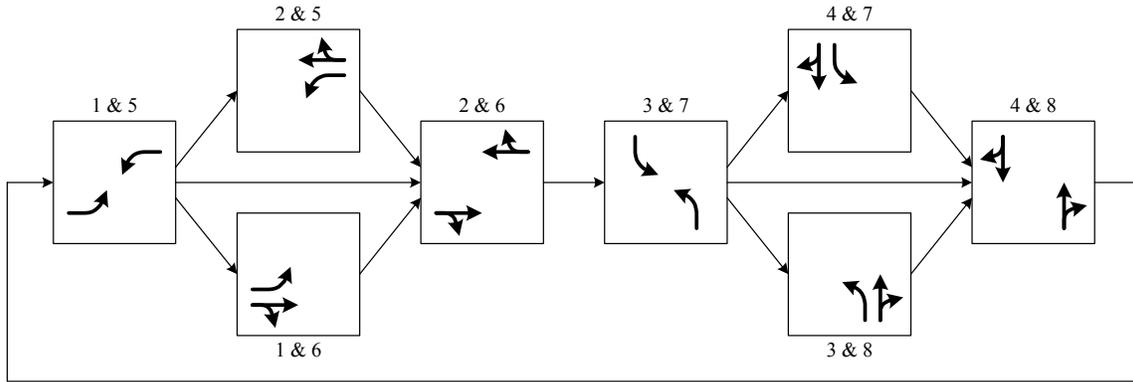


Figure 3-6: Phase order for dual-ring controller.

The NEMA and Model 170 controllers can operate in multiple ways, including pretimed, actuated-isolated, and actuated-coordinated, each of which can be modeled by the generic controller as described in Section 3.3 above. Other advanced functions specific to the NEMA and Model 170 controllers have also been implemented, including the gap reduction feature as shown in Figure 2-7. An example input file for a simulated NEMA-type phase-based controller can be found in Appendix B.

3.4.2 European Standards

Unlike the NEMA and Model 170 controllers used in the United States, European controllers are based on signal groups, similar to the generic controller implementation. Figure 3-7 shows the example signal group diagram presented in Section 2.1.1. In this example, the cycle starts with groups 1 and 3 in green periods. When group 3 terminates, group 2 begins its green period after a red clearance interval. Groups 1 and 2 then terminate together, followed by group 4 after a clearance interval. After group 4 terminates and a red clearance interval has passed, the cycle repeats with groups 1 and 3 in green.

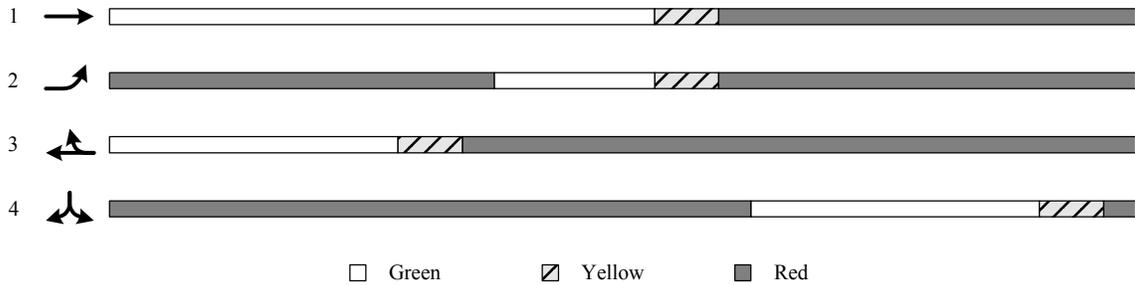


Figure 3-7: Signal group diagram.

The times at which each group changes periods depend on the type of control applied. Under pretimed control, for example, the changes in period occur at fixed times within the cycle. With actuated control, the duration of each green period may be variable, and following groups must wait for conflicting groups to end before starting in green. Under coordinated operation, the controller relies on coordination pulses from a master controller or master clock to keep in synchronization with other controllers. When received, a pulse will trigger specified actions within the controller, most commonly terminating an active group and transitioning to the next period.

Because the generic controller is also based on signal groups, modeling the logic employed by European controllers is quite straightforward. The signal group order is specified with the “conflict clearance” condition in the same way that phase order is specified. Groups can also be synchronized with other signal groups, constrained to change together or after a specified delay. Coordination pulses can be modeled either as time-based conditions, such as a force-off time, or by the use of status flags which can be set externally and used to trigger other actions. Logic for the various control types, i.e. pretimed/actuated and isolated/coordinated, can be modeled by the generic controller as described in Section 3.3 above. An example input file for a simulated European standard controller can be found in Section C.1 of Appendix C.

3.5 Transit Priority Capabilities

The basic functions of the generic controller, in combination with transit vehicle detectors, can be used to simulate basic active priority strategies for transit. For example, under actuated control, the green interval is extended when vehicles are detected at an upstream sensor. The same logic elements used to model this behavior can also be

applied to simulate green interval extension for a transit vehicle. A simple specification is shown in Table 3.8. This specification is identical to the basic actuated-isolated logic specification (Table 3.5), but here the gap timer and extension time conditions are used for transit priority. In this case, the gap timer condition specifies the ID of the transit vehicle detector, and the extension timer condition specifies the amount of extra green time that is granted once the transit vehicle is detected. If no transit vehicles are present, therefore, the signal group will operate in a pretimed mode, displaying green for the time specified in the minimum time condition before changing to yellow. Only if a transit vehicle is detected will the extension time condition be valid, holding the green for a specified duration after time of detection.

Table 3.8: Simple transit priority specification – phase extension.

Period	Condition	
	Type	Description
	General	Gap timer (transit vehicle detector)
Green	Hold	Minimum time
	Hold	Extension time (transit priority)
Yellow	Hold	Minimum time
Red	Hold	Conflict clearance

An early green for transit vehicles can also be modeled with the existing conditions. In this case, detection of the transit vehicle causes the preceding phase to be terminated early, allowing the phase in which the transit vehicle moves to start earlier than normal. The logic for the shortened phase is specified as shown in Table 3.9. This specification is simply the phase-based pretimed-isolated logic specification of Table 3.2 with an added skip condition that is dependent on the transit vehicle detector state. If a transit vehicle is detected, the hold condition for the green interval will be skipped, terminating the green interval if it is active. Once this phase has changed to red, the normal logic will cause the next phase to start, providing an earlier green for the transit vehicle.

Table 3.9: Simple transit priority specification – phase shortening.

Period	Condition	
	Type	Description
Green	Skip	Skip if detector actuated (transit)
	Hold	Minimum time
Yellow	Hold	Minimum time
Red	Hold	Conflict clearance

3.6 Implementation of Advanced Control Strategies

While the standard conditions of the generic controller can be used to model a wide range of control types and strategies, specialized conditions may be required in order to model the logic details of advanced control strategies. The generic controller is designed such that new conditions can be easily added and applied within the existing controller framework. By adding new conditions, new logic capabilities can be incorporated into the controller as needed. For example, support for adaptive control logic can be added in this way, as discussed in Section 3.3.3.

The next chapter details the implementation of transit signal priority within the framework of the generic controller. Whereas the preceding section described how simple priority functions can be modeled, the next chapter will detail the logic and implementation of a full advanced strategy for transit signal priority.

Chapter 4

Implementation of Transit Signal Priority

This chapter describes the implementation of transit signal priority within MITSIMLab. The model for the implementation is PRIBUSS, an advanced bus priority strategy developed and employed in Stockholm, Sweden. This chapter details the logic of PRIBUSS and how it is modeled within the framework of the generic controller presented in the previous chapter.

4.1 PRIBUSS

PRIBUSS is an active signal priority strategy for buses that was developed for use in the city of Stockholm, Sweden. Its name is an acronym for “*Prioritering av Bussar i Samordnade Signalsystem*,” or “Prioritization of Buses in a Coordinated Signal System.” The strategy was developed by *Gatu- och Fastighetskontoret* (GFK), the administration in charge of traffic planning and operations in the city, in cooperation with *Storstockholms Lokaltrafik* (SL), Greater Stockholm’s local public transit agency. The objective of the system is to provide priority to public transit buses without significant disruption of signal operations, especially under coordinated control.

4.1.1 General Concepts

There are four functions that PRIBUSS employs in order to grant priority. Three of these functions are equivalent to the basic active priority actions described in Section 2.2.2. “*Green Extension*” extends the green period for enough time to allow the bus to pass through the intersection. “*Phase Shortening*” ends the current phase early in order

to start the bus phase early. “*Extra Phase Insertion*” gives green to the bus out of sequence, adding an additional phase for the bus. The fourth action, “*Restart Green*,” is a variant of Green Extension in which a signal whose green period has just ended will repeat the green period for enough time for an approaching bus to pass. These actions will be described in more detail in the following section.

Implementation of the PRIBUSS strategy requires the detection of buses upstream of the intersection in order to start any priority action. When a bus crosses an upstream detector, this “check-in” detector indicates the presence of the bus to the signal controller. In order for the controller to know when the bus has passed the intersection and is no longer in need of priority, a “check-out” detector is also needed in or immediately downstream of the intersection. The controller keeps count of the number of buses between the check-in and check-out detectors by means of a counter. This counter is increased by 1 when the check-in detector is actuated and is decreased by 1 when the check-out detector is actuated. Priority is only called for when the counter has a value of 1 or more. When the value of this counter becomes 0, indicating that no more buses are immediately upstream of the intersection, any priority action underway will be stopped.

The type of priority action called for depends on when in the cycle the bus is detected. As an example, Figure 4-1 shows the valid times for each priority call superimposed on a signal group diagram.* Group 1 is the priority group, meaning that the bus has right-of-way when this group is in green. Group 2 is the group which displays green immediately following the priority group, and Group 3 is the group which displays green immediately preceding the priority group.

If a bus is detected during the green period of Group 1, it can call for green extension. However, the priority action may or may not be needed, depending on when in the period the bus is detected. For example, a bus detected at the start of the green period will most likely not need any priority action to be taken, as adequate green time will exist for it to clear the intersection. A bus that is detected near the end of the green period, however, will require green extension in order to make it through the intersection before the end of the period.

* Note that traffic signal controllers in Sweden precede the green period with a starting red/yellow signal indication, as shown in the signal diagram in Figure 4-1.

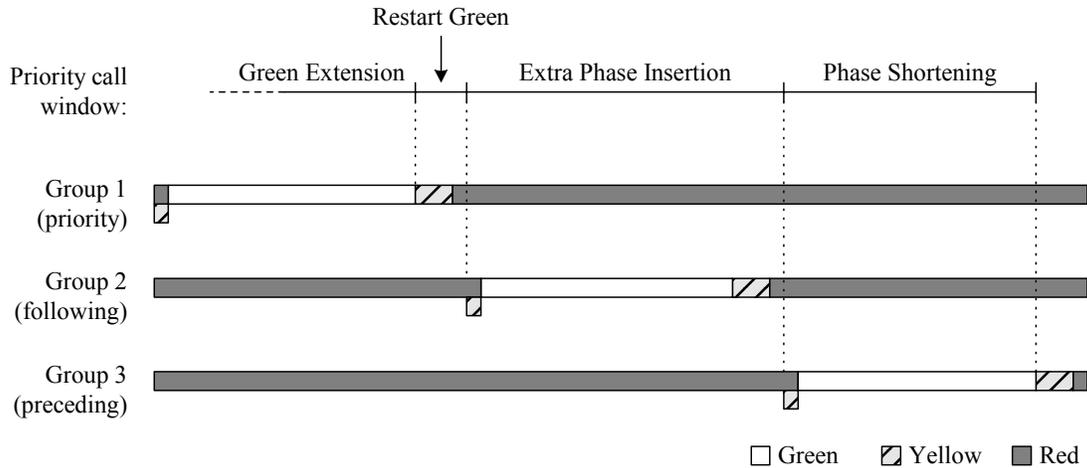


Figure 4-1: Time windows for priority calls.

A call for green restart can only occur once the green period has ended and before the starting red/yellow indication of the following group is displayed. If a bus is detected during this window, Group 1 can return to green to allow the bus to pass. Once Group 2 has started, the next available priority action is extra phase insertion, which can be called for until Group 3 becomes active. A bus detected during this time will be served by a green period inserted between the green periods of Groups 2 and 3. Once Group 3 has started, a detected bus will call for phase shortening, which will terminate the green period of Group 3 early to give an early start to Group 1. No priority call is needed once Group 3 has ended, because the green period of the priority group is the next in sequence.

A large number of parameters is required to implement the PRIBUSS strategy in a signal controller. However, a smaller set of parameters governs the key details of the implementation and is common to all priority actions:

- *Guaranteed Time:* Each signal group has a minimum duration during which any conflicting priority calls are inhibited. This ensures a minimum service time for non-priority phases.
- *Detector Locations:* Each priority group requires a bus detector upstream of the intersection to indicate the presence of the vehicle. The distance of the detector from the stop line is used to estimate the arrival time of a vehicle to the intersection once it has been detected.

- *Allowed Window*: The window within which priority calls can be registered is specified by start and end times in each cycle. These values will depend on multiple factors, including the estimated travel time of the bus from its detection point to the intersection and the time available from other signal groups for priority actions.

4.1.2 Priority Actions

This section provides detailed explanations of the four priority actions that can be implemented as part of the PRIBUSS strategy.

Green Extension

If a bus is detected during the green period for its approach and that period is about to end, priority can be given by means of green extension. This function will hold the green period for the approaching bus in order to give the bus enough time to pass through the intersection. Once the bus has passed, the period will terminate and the controller will return to the normal timing.

Figure 4-2 shows the method by which green extension is implemented. In this diagram, the vertical axis represents time. On the left side of the figure is a time-space diagram that shows the locations of the check-in and check-out detectors. On the right side of the figure is a signal group diagram that shows signal indications over time. The signal groups shown in the diagram are the two groups affected by the priority action: the priority group, and the following conflicting group that receives a delayed start.

Extension is only useful to buses which arrive at the intersection between times C and D. Time C is the normal ending time for the green period as specified in the signal timings. Buses arriving earlier than time C can cross the intersection during the normal green period and do not require additional extension time. Time D is the latest ending time allowed for the green period. This point is determined by the latest time the following signal group can start and still receive its guaranteed green time. Buses arriving after time D can not be given green extension because no more time can be taken from the following group.

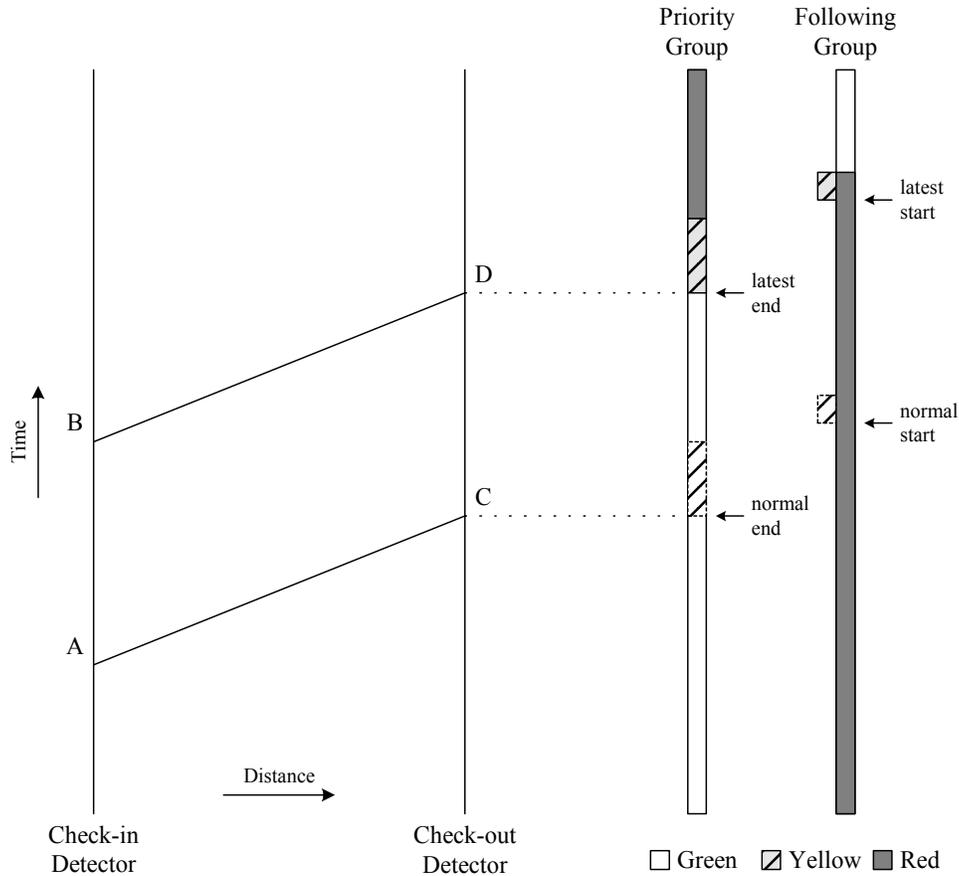


Figure 4-2: Execution of green extension.

Because the check-in detectors are typically located 100-200 meters upstream of the intersection, there is a delay between when the vehicle is detected and when it reaches the intersection. This delay is equal to the travel time between the check-in detector and the intersection. Because of this delay, a bus that crosses the check-in detector before time C may reach the intersection after time C, requiring green extension. Therefore, the time period in which a detected bus will receive priority must be shifted earlier to account for this delay. This is represented by the time period A-B in the figure. A bus that checks in during this time period will reach the intersection in the C-D time period and will require priority. This assumes, however, that the assumed travel time between the detector and the intersection is correct. In operation, this travel time will vary due to traffic and other conditions, so a longer travel time is usually assumed in order to be conservative.

A further distinction is made between the time period in which priority can be initiated and the time period in which an ongoing priority action can be continued with

the detection of another bus. At time C, the bus phase will end if a call for extension has not been initiated. Therefore, time period A-C is the window in which extension can be initiated. If extension is already ongoing, however, additional buses detected before time B will further extend the period.

Shortening of Current Phase

If a bus is detected during the red period for its approach and the bus is due to receive green in the next scheduled phase, the current phase can be shortened to allow the next phase to receive green earlier than normal. Figure 4-3 shows how this priority action is implemented. The signal groups shown are the two groups affected by the priority action: the priority group, and the preceding conflicting group that is terminated early. The time window in which shortening can act is defined by period C-D in the diagram. Time C is the earliest time that the green period for the bus can start. This point is determined by the guaranteed time for the preceding signal group, which defines the earliest time that its green period can end. Time D is the normal start time for the bus green period, at which time shortening no longer applies.

The time window in which detection of a bus will call for shortening is defined by time period A-B. Time A, which is the earliest time a call for shortening can be made, is equal to the start time of the preceding phase. A bus detected before time A can receive an inserted phase, so shortening is not required. A bus detected after time A will call for priority; however, priority will not be granted until time C, when the guaranteed time for the preceding phase has been met. Time B, which is the latest time a call for shortening can be made, is equal to the normal end time of the preceding phase. At time B, the transition to the bus's normal phase begins, so a bus detected after time B will receive green at the usual time

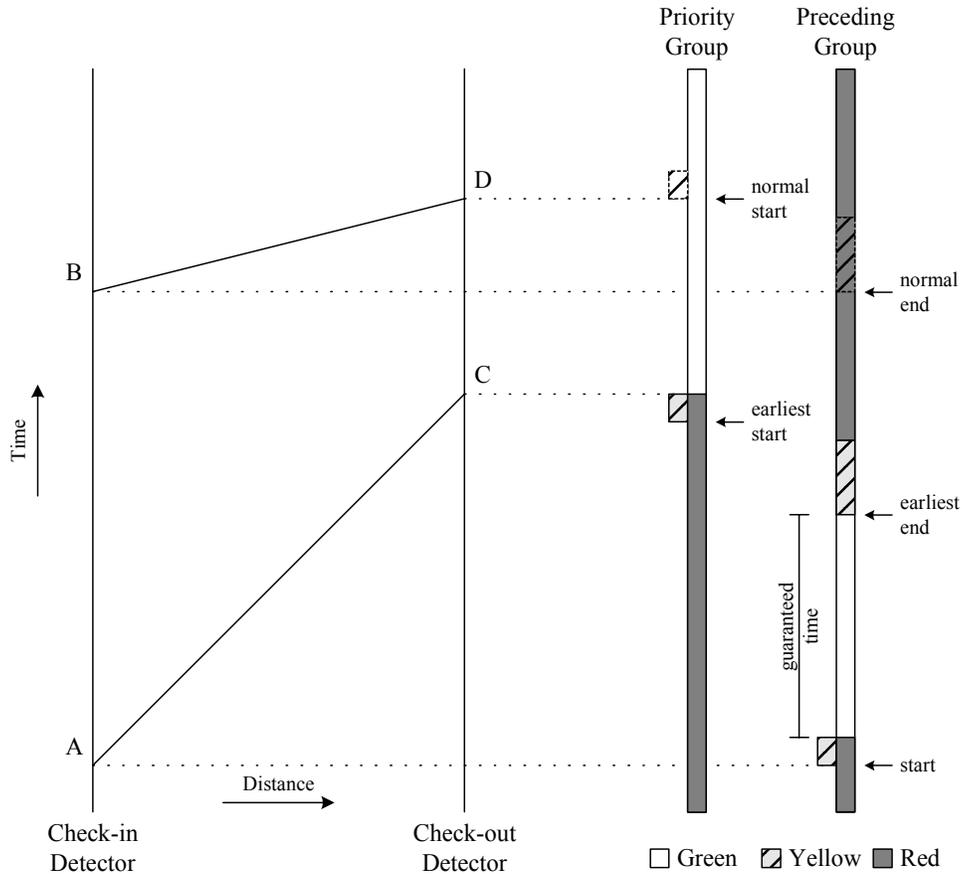


Figure 4-3: Execution of shortening of current phase.

Insertion of Extra Phase

If a bus is detected during the red period for its approach and the bus is not due to receive green in the next scheduled phase, an extra phase for the bus approach can be inserted between the current phase and the next phase. Figure 4-4 shows how phase insertion is implemented. Three signal groups are affected by the phase insertion and are shown in the diagram: the priority group being inserted, the preceding group that is terminated early, and the following group which receives a delayed start. The time window in which the extra phase can be inserted is defined by period C-D in the diagram. Time C is the earliest time that the green period for the inserted bus phase can start. This point is determined by the guaranteed time for the preceding signal group, which defines the earliest time that its green period can end. Time D is the latest time that the green

period for the inserted bus phase can end. This point is determined by the latest time the following signal group can start and still receive its guaranteed green time.

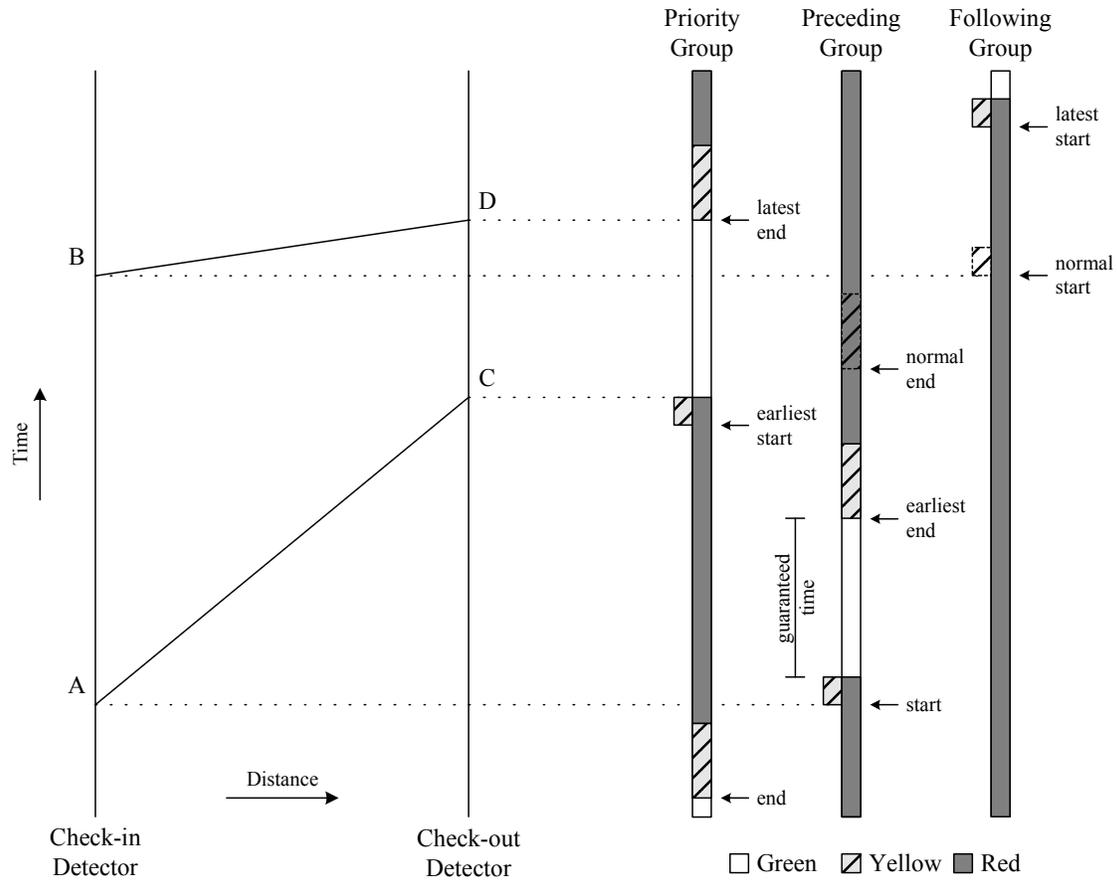


Figure 4-4: Execution of insertion of extra phase.

The time window in which detection of a bus will call for an inserted extra phase is defined by time period A-B. Time A, which is the earliest time this call can be made, is equal to the start time of the preceding phase. A bus detected before time A can receive a green restart, so insertion is not required. A bus detected after time A will call for priority; however, priority will not be granted until time C, when the guaranteed time for the preceding phase has been met. Time B, which is the latest time a call for insertion can be made, is equal to the normal start time of the following phase. A bus detected after time B may call for phase shortening or insertion between two other phases, but it can not be served until after the active phase has received its guaranteed time.

Green Restart

If a bus is detected when the green period for its approach has just ended, the green period can be restarted to allow the bus to pass through the intersection without waiting for the next phase. Figure 4-5 shows how this priority action is implemented. As with green extension, the two affected groups shown are the priority group and the following conflicting group that receives a delayed start. The time window in which green restart can act is defined by period C-D in the diagram. Time C is the time at which the bus period can restart green, taking into account the red clearance time and the starting red/yellow indication.* Time D is the latest time that the restarted green period can end. This point is determined by the latest time the following signal group can start and still receive its guaranteed green time.

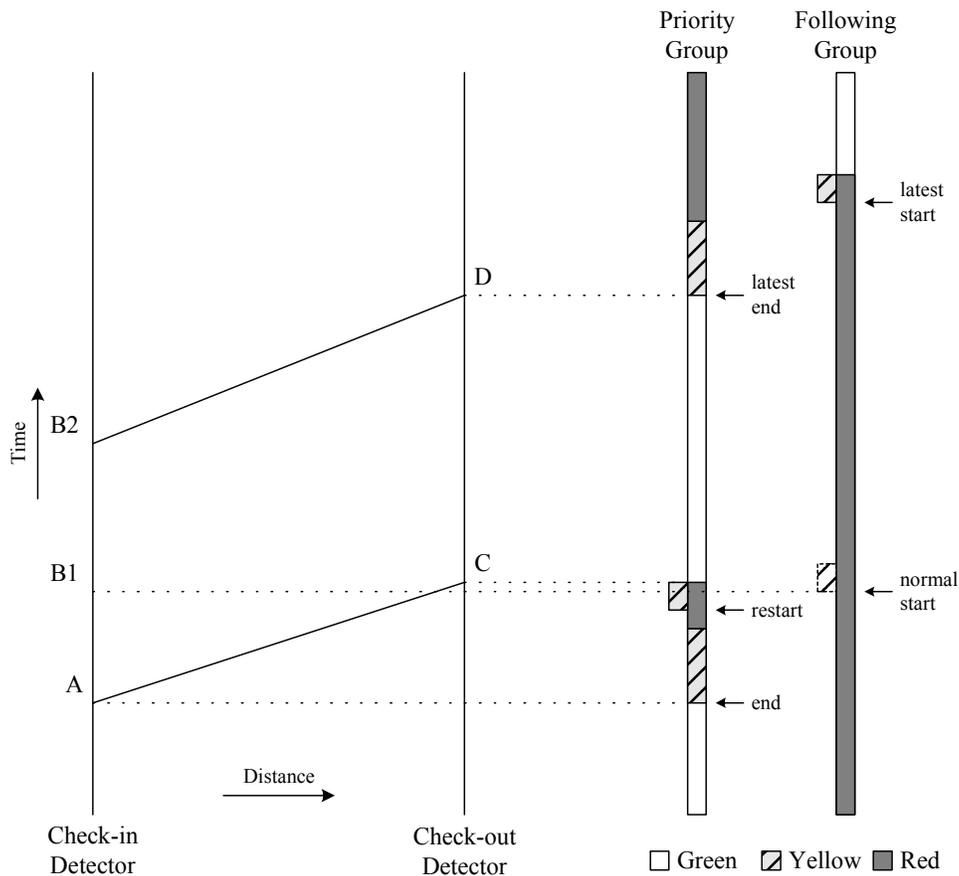


Figure 4-5: Execution of green restart.

* In this situation, the red clearance time is not technically needed, because the same group has just been active and no conflicting movements need to be cleared. However, a short red interval is still customarily used to maintain the standard yellow→red→red/yellow signal sequence, reducing driver confusion.

As with green extension, there is a distinction between the time period in which green restart can be initiated (A-B1) and the time period in which the restarted period can be continued (A-B2). Time A, which is the earliest time this call can be made, is equal to the normal end time of the green interval of the priority phase. A bus detected before time A can receive green extension, so green restart is not required. Time B1, which is the latest time green restart can be initiated, is equal to the normal start time of the red/yellow period of the following signal group. If green restart has not been initiated by time B1, the following period will begin and green restart will no longer be possible. If green restart is already ongoing, however, additional buses detected before time B2 (equal to time D less the travel time between the detector and the intersection) will extend the restarted period. Additional buses detected after time B2 will not extend the period because they will not be able to reach the intersection before the restarted phase must be terminated.

4.2 Implementation in MITSIMLab

In order to model the specialized features of PRIBUSS within MITSIMLab, new conditions and status flags are added within the framework of the generic controller. Another addition is a check-in counter like that employed in the PRIBUSS logic. With these enhancements, the generic controller can be used to simulate the logic and functionality of the PRIBUSS system.

4.2.1 Generic Controller Enhancements

Status Flags

Within the generic controller, each signal group has a flag variable that holds information about the current status of the group (see Section 3.2.2). For the modeling of PRIBUSS, additional status flags were added to indicate the priority status of a signal group and, specifically, the priority action being implemented. The flags used by each priority action are listed below:

- *Green Extension*: This action requires a single “extension” flag that is applied to the signal groups in the phase receiving priority.

- *Phase Shortening*: This action requires two flags. The groups in the phase that is terminated early are marked with “shortened” flags. The groups in the priority phase that follows are marked with an “early start” flag.
- *Phase Insertion*: This action requires three flags. An “inserted” flag is applied to the groups of the priority phase that is inserted out of sequence. The groups of the phase terminated early are marked with “insertion after” flags, and the groups of the phase following the inserted phase are marked with “inserted before” flags.
- *Green Restart*: Because this action is simply a variant of “green extension,” it can use the same “extension” status flag, which is applied to the groups of the restarted phase.

Conditions

Following the PRIBUSS logic, the priority status flags are set only when a bus is detected within a specified time window. The flags are unset either when the check-out detector returns to zero or when the termination point of the priority action is reached (corresponding to Time D in Figure 4-2 through Figure 4-5). The logic by which a flag is set and unset is implemented as a general condition. Updating of the check-in counter is also performed by general conditions. The new general conditions used for modeling PRIBUSS are listed below. Details of the conditions, including the input parameter requirements, are shown in Table A.1 of Appendix A.

- *Check-In*: This condition adds 1 to the check-in counter when the specified detector is actuated.
- *Check-Out*: This condition subtracts 1 from the check-in counter when the specified detector is actuated.
- *Set/Unset “Extension” Flag*: This condition sets the “extension” flag of the signal group if a bus checks-in within the specified time window in the cycle. The flag is unset when the check-in counter becomes equal to zero or when the specified termination point is reached.
- *Set/Unset “Early Start” Flag*: This condition sets the “early start” flag of the signal group if the check-in counter is greater than zero during the specified time window in the cycle. The flag is unset when the check-in counter becomes equal to zero or when the specified termination point is reached.

- *Set/Unset “Shortened” Flag*: This condition sets the “shortened” flags of the specified signal groups if the check-in counter is greater than zero during the specified time window in the cycle. The flags are unset when the check-in counter becomes equal to zero or when the specified termination point is reached.
- *Set/Unset “Insertion” Flag*: This condition sets the “inserted” flag of the signal group if the check-in counter is greater than zero during the specified time window in the cycle. The flags are unset when the check-in counter becomes equal to zero or when the specified termination point is reached.
- *Set/Unset “Insertion After” Flag*: This condition sets the “inserted after” flags of the specified signal groups if the check-in counter is greater than zero during the specified time window in the cycle. The flags are unset when the check-in counter becomes equal to zero or when the specified termination point is reached.
- *Set/Unset “Insertion Before” Flag*: This condition sets the “inserted before” flags of the specified signal groups if the check-in counter is greater than zero during the specified time window in the cycle. The flags are unset when the check-in counter becomes equal to zero or when the specified termination point is reached.

The rest of the logic can be implemented with the basic conditions of the generic controller by making use of the skip condition which queries the status flag of a specified signal group. Using this skip condition essentially allows an alternate logic specification that goes into effect only when a priority flag is set. This usage will be demonstrated with the examples in the following section.

4.2.2 Logic Specification

This section will give input specification examples for all four priority actions that are used by PRIBUSS. For simplicity, the examples will be based on the basic pretimed-coordinated example shown in Table 4.1 (see Section 3.3.1 for description). A sample MITSIMLab signal input file with PRIBUSS features implemented can also be found in Section C.2 of Appendix C.

Table 4.1: Basic pretimed-coordinated logic specification.

Period	Condition	
	Type	Description
Green	Hold	Hold in window
Yellow	Hold	Minimum time
Red	Hold	Conflict clearance

Green Extension

Table 4.2 shows the conditions required for modeling a signal group subject to green extension. Although the original conditions are still valid, they are shown in gray to distinguish them from the additional conditions needed for priority. There are three new general conditions. The first two conditions define the sensors that are used as the check-in and check-out detectors. The third condition defines the logic for calling and terminating priority, specifying the allowed time window in which priority can be initiated or continued and the conditions for termination of priority.

Table 4.2: “Green Extension” logic specification – priority signal group.

Period	Condition	
	Type	Description
	General	Check-in
	General	Check-out
	General	Set/unset “extension” flag
Green	Hold	Hold in window
	Skip	Skip next (1) if “extension” flag not set
	Hold	Hold indefinitely
Yellow	Hold	Minimum time
Red	Hold	Conflict clearance

Without a priority call, the group operates as normal, because the new hold condition is skipped. Once priority is called and the extension flag is set, however, this condition will be valid and can hold the green period past its normal time window. The green period will be held until the extension flag is unset by the general condition, at which point the green period will terminate, assuming that the “hold in window” condition is no longer valid.

Although green extension delays the start time of the immediately subsequent signal groups, no changes are needed for these groups. The existing conflict clearance

conditions will serve to hold the groups in red until the priority group has ended and the specified clearance time has elapsed.

Shortening of Current Phase

To model phase shortening, new conditions must be specified both for the priority group and for the groups of the preceding phase that is shortened. Table 4.3 shows the conditions required for modeling the priority group, and Table 4.4 shows the conditions required for modeling the shortened signal groups.

Table 4.3: “Phase Shortening” logic specification – priority signal group.

Period	Condition	
	Type	Description
	General	Check-in
	General	Check-out
	General	Set/unset “early start” flag
	General	Set/unset “shortened” flags
Green	Hold	Hold in window
Yellow	Hold	Minimum time
Red	Hold	Conflict clearance

Table 4.4: “Phase Shortening” logic specification – shortened signal group.

Period	Condition	
	Type	Description
Green	Skip	Skip next (1) if “shortened” flag set
	Hold	Hold in window
	Hold	Minimum time
Yellow	Hold	Minimum time
Red	Hold	Conflict clearance

With no priority call, the groups operate as normal. If a priority call is received, however, the general conditions of the priority signal group will set the priority flags for both the priority group and for each preceding group to be shortened. Once the shortened flag is set for a preceding group, the original green hold condition will be skipped by the new skip condition. This will cause the green period to terminate unless the green period has not been active for the minimum time specified in the new hold condition. This minimum time is the guaranteed time for the shortened group, and this condition will inhibit termination of the phase unless the guaranteed time has been served. The priority

group will begin once the shortened groups are terminated and the specified clearance interval has passed, as in normal operation.

Insertion of Extra Phase

To model phase insertion, new conditions must be specified for the groups of the inserted priority phase as well as for the groups of the phases both preceding and following the inserted phase. Table 4.5 shows the conditions required for modeling the inserted priority groups, Table 4.6 shows the conditions for the preceding groups, and Table 4.7 shows the conditions for the following groups.

Table 4.5: “Phase Insertion” logic specification – priority signal group.

Period	Condition	
	Type	Description
	General	Check-in
	General	Check-out
	General	Set/unset insertion flag
	General	Set/unset insertion flags (preceding groups)
	General	Set/unset insertion flags (following groups)
Green	Hold	Hold in window
	Skip	Skip next (1) if “inserted” flag not set
	Hold	Hold indefinitely
Yellow	Hold	Minimum time
Red	Skip	Skip next (1) if “inserted” flag set
	Hold	Conflict clearance (for regular phase)
	Skip	Skip next (1) if “inserted” flag not set
	Hold	Conflict clearance (for inserted phase)

Table 4.6: “Phase Insertion” logic specification – preceding signal group.

Period	Condition	
	Type	Description
Green	Skip	Skip next (1) if “insertion after” flag set
	Hold	Hold in window
	Hold	Minimum time
Yellow	Hold	Minimum time
Red	Hold	Conflict clearance

Table 4.7: “Phase Insertion” logic specification – following signal group.

Period	Condition	
	Type	Description
Green	Hold	Hold in window
Yellow	Hold	Minimum time
Red	Skip	Skip next (1) if “insertion before” flag set
	Hold	Conflict clearance (regular phasing)
	Skip	Skip next (1) if “inserted before” flag not set
	Hold	Conflict clearance (after inserted phase)

With no priority call, the groups operate as normal. If a priority call is received, however, the general conditions of the priority group will set the priority flags for all of the groups impacted by the priority. The logic for the preceding groups is the same as for shortened groups: if flagged for priority, the period will terminate after the guaranteed time has been served. Once the preceding groups are terminated, the priority groups will begin after a specified clearance time. These groups start out of the usual sequence because an alternate conflict clearance condition has been defined that specifies the group to wait for the preceding groups, ignoring the usual conflict clearance condition. The priority group is then held in green until the priority flag is removed. The following groups also have an alternate conflict clearance condition that prevents them from starting until the priority group is complete. Once the priority group is complete, the following signal groups will begin after a clearance interval defined by the alternate conflict clearance conditions.

Green Restart

Table 4.8 shows the conditions required for modeling green restart. Because green restart is a variant of green extension, it can use the same general conditions. Without a priority call, the group operates normally. With a call for priority, the extension flag is set. If the call comes while the period is in green, the logic for green extension will be implemented. By definition, however, a call for restart can only come during the yellow period and the portion of the red period before conflicting groups have started. To allow a call during this time, the window in which the extension flag can be set is lengthened to include the yellow period and the start of the red period. If this flag is set, the normal

conflict clearance condition will be skipped and the red period will be held for a short minimum time before returning to green.

Table 4.8: “Green Restart” logic specification – priority signal group.

Period	Condition	
	Type	Description
	General	Check-in
	General	Check-out
	General	Set/unset extension flag
Green	Hold	Hold in window
	Skip	Skip next (1) if “extension” flag not set
	Hold	Hold indefinitely
Yellow	Hold	Minimum time
Red	Skip	Skip next (1) if “extension” flag set
	Hold	Conflict clearance (regular phasing)
	Skip	Skip next (1) if “extension” flag not set
	Hold	Minimum time

4.3 Strategy Implementation

Implementing transit signal priority under isolated intersection control is relatively trivial, as there are no external constraints on the operation of the controller. Coordinated control, however, can place tight restrictions on signal operation, making priority implementation significantly more challenging. For example, coordination may dictate a fixed common cycle length and consistent green periods every cycle in order to create a green band for vehicle progression. Providing transit priority under such conditions can mean breaking some of these constraints, disturbing the underlying control strategy.

The impacts from this disturbance may not necessarily be significant. Under low traffic conditions, for instance, breaking coordination to give priority to a transit vehicle may not have a large cost to other traffic, because the low volumes will mean that the system can recover quickly. Under peak period operation, however, the high traffic volumes create conditions which are much more sensitive to disturbances, with intersections potentially operating close to capacity. The next chapter presents a case study which uses the transit signal priority implementation within MITSIMLab to evaluate the performance of the PRIBUSS strategy under such conditions.

Chapter 5

Case Study

The PRIBUSS strategy is employed extensively throughout the city of Stockholm. The most notable implementation is on the three bus rapid transit lines that operate in the Stockholm inner city. With vehicles painted blue to distinguish them from the typical red city buses, these “blue bus” lines are designed to complement the subway network, providing high frequency service on the most highly traveled transit corridors. The vehicles are high-capacity articulated buses, with low floors and four sets of doors to reduce boarding and alighting time. The fleet is also equipped with a GPS-based automatic vehicle location system, used for real-time arrival information at bus stops, fleet management, and for traffic signal priority.

In this study, MITSIMLab and the enhanced generic controller are used to model the operations of one of these blue bus lines, simulating a portion of its route and the surrounding urban network. The purpose of the case study is to evaluate the existing PRIBUSS implementation in terms of impacts on travel times in the network, both for the priority-equipped buses and for other traffic, and in terms of impacts on service reliability for the bus line. Performance under alternate parameter settings is then evaluated, and operational recommendations are made.

5.1 Study Network

The network selected for the case study is located on Södermalm, the southernmost of the islands that comprise the Stockholm inner city. The network consists of three major arterials that converge at Hornstull, a commercial hub at the western end of the island.

This location was selected because it is a major through route for automobile traffic and because it is also along the route of one of the city's bus rapid transit lines.

A map of the southern portion of the Stockholm inner city is shown in Figure 5-1 with the location of the network highlighted. A schematic diagram of the network is shown in Figure 5-2. Hornstull is located at the northern end of Liljeholmsbron, a bridge which serves as the entry point to the Stockholm inner city from the southwest, and all traffic entering at this location must pass through the Hornstull intersection. At this intersection, the entering traffic stream diverges, with vehicles bound for the northern portion of the city continuing straight on Långholmsgatan and those bound for the southern and central portions turning right onto Hornsgatan. Långholmsgatan also sees significant traffic because it leads to Västerbron ("the West Bridge"), which is one of the few connections between Södermalm and the rest of the inner city to the north. Cross-street traffic within the network is quite low, mostly as a result of street closures that prevent through traffic from bypassing the area using smaller side streets.

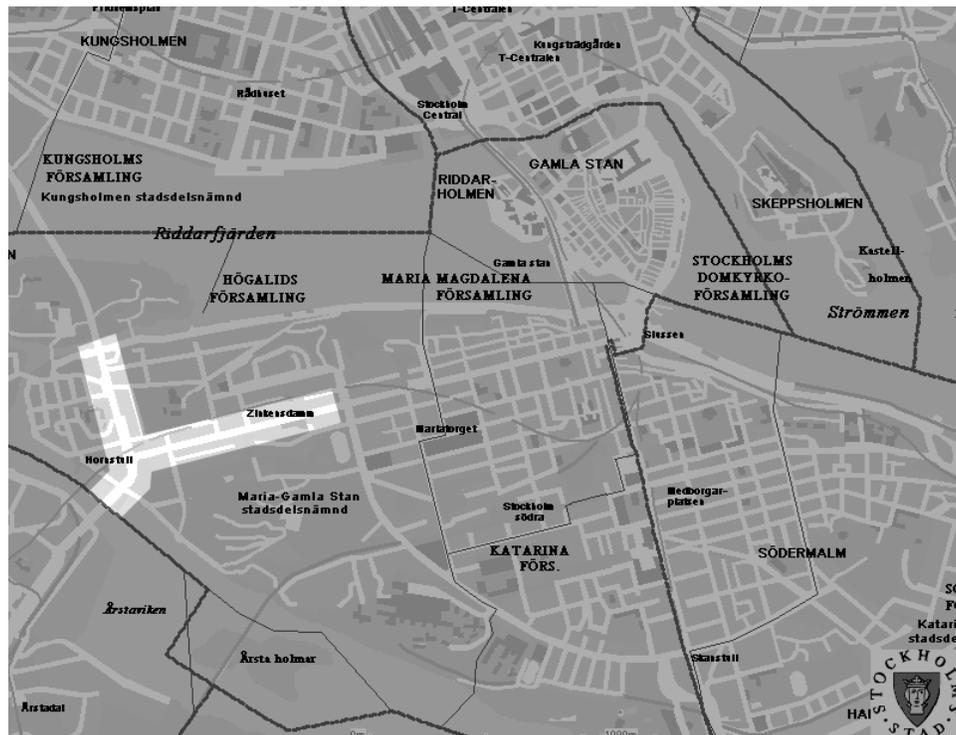


Figure 5-1: Hornstull network location.

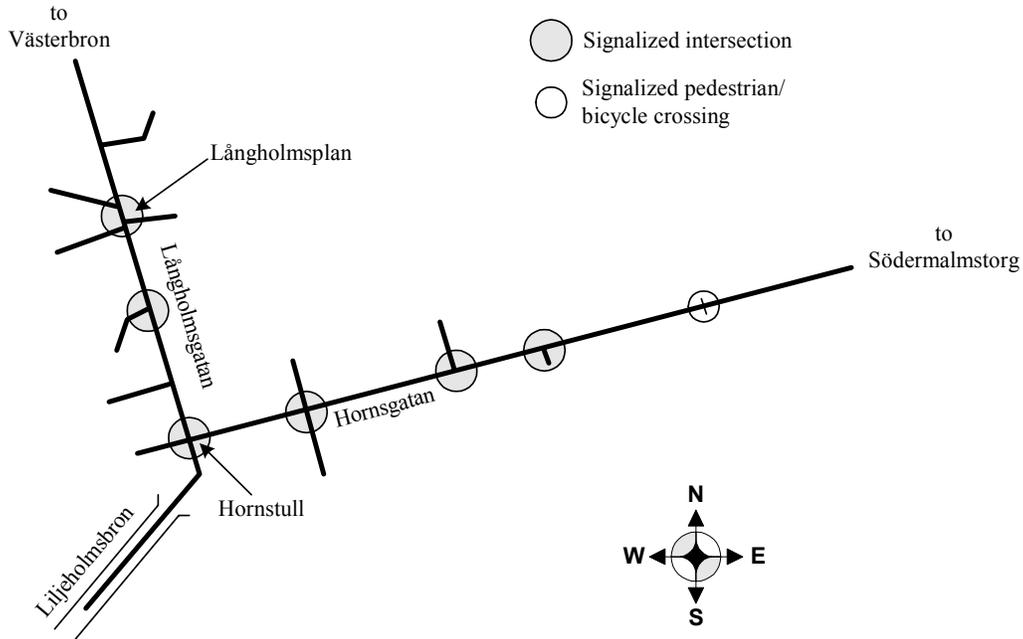


Figure 5-2: Schematic of Hornstull network.

Six signalized intersections and one signalized pedestrian/bicycle crossing are located along the network, as shown in Figure 5-2. The signals can function under actuated control, with actuation either from inductive loop detectors (for vehicle and cycle signal groups) or from push-button actuators (for pedestrian signal groups). During daytime hours, however, the signals run in pretimed-coordinated operation, providing progression in both directions along the Hornsgatan-Långholmsgatan corridor. Although the standard city street speed limit is 50 km/h (31 mph), the coordination is timed for a lower-speed progression of 12 m/s, or 43 km/h (27 mph). During overnight hours, each intersection runs in isolated actuated operation.

Four bus routes pass through the simulated network, including one of the city's three blue bus lines. The local red buses (Lines 40, 66, and 74) that cross the network run every 15 minutes during peak hours, while the blue bus (Line 4) operates on 7.5 minute headways throughout the day. There are four bus stop locations in each direction on the network, the locations of which are indicated in Figure 5-3.

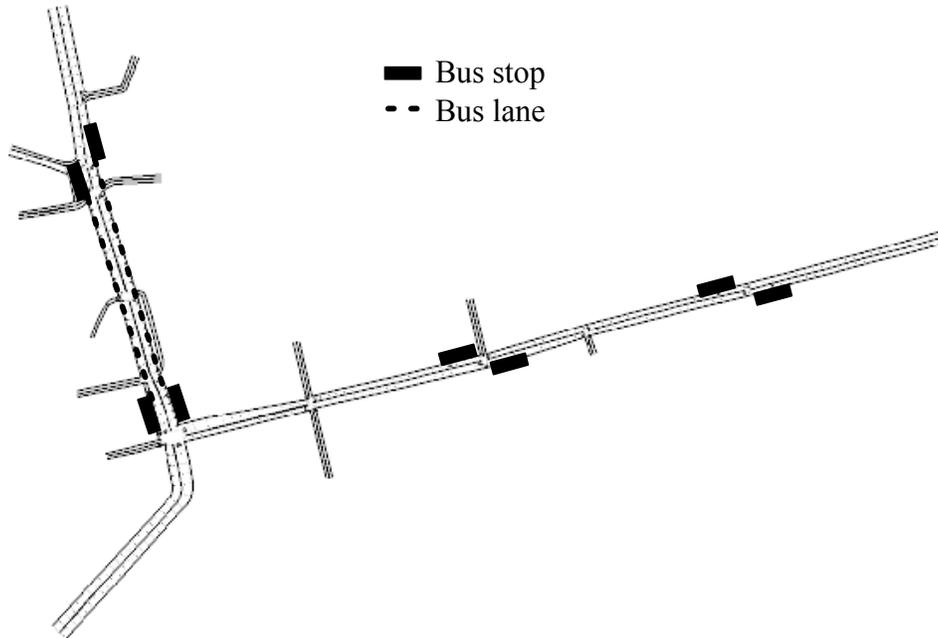


Figure 5-3: Bus facilities on Hornstull network.

Of the eight bus stops, all are far-side bus stops (i.e. located just downstream of an intersection) except for the two southbound stops on Långholmsgatan that use near-side stops just upstream of an intersection. On Långholmsgatan between Långholmsplan and Hornstull, the right-side parking lanes in both directions serve as exclusive bus lanes during peak hours. However, this means that southbound buses, which must turn left at Hornstull, need to make that turn from a right-side bus lane. Because this movement conflicts with all southbound movements at the intersection, the southbound buses move as a separate signal group. The configuration of this intersection is shown in Figure 5-4.

The blue buses of Line 4 are equipped for signal priority, and the PRIBUSS strategy is implemented at all of the intersections within the network. “Green extension” and “phase reduction” are allowed at all intersections in response to detection of a blue bus. In addition, “phase insertion” is allowed at the Hornstull intersection for the southbound left-turning bus group. “Green restart” is not used on the network, however, because inadequate time exists from non-priority phases for this action to be implemented.

Bus detection points are located both upstream and downstream of every signal. Typically, the upstream detection points are located 150 to 180 meters from the downstream signal, corresponding to a travel time of 12.5 to 15 seconds at the signal progression speed. In the locations with near-side bus stops, where buses must pickup

and discharge passengers upstream of the traffic signal, the radio detection signal is not sent until the bus has finished loading and is ready to proceed. In these cases, the detection signal is sent when the door of the bus closes.

Pedestrian traffic is also quite significant on the network, especially at the Hornstull intersection. Much of this traffic is due to the subway station entrance that is located on the northeast corner. Pedestrian demand is especially large crossing Långholmsgatan, the north leg of the intersection, because people transferring between the southbound buses and the subway must cross at this location.

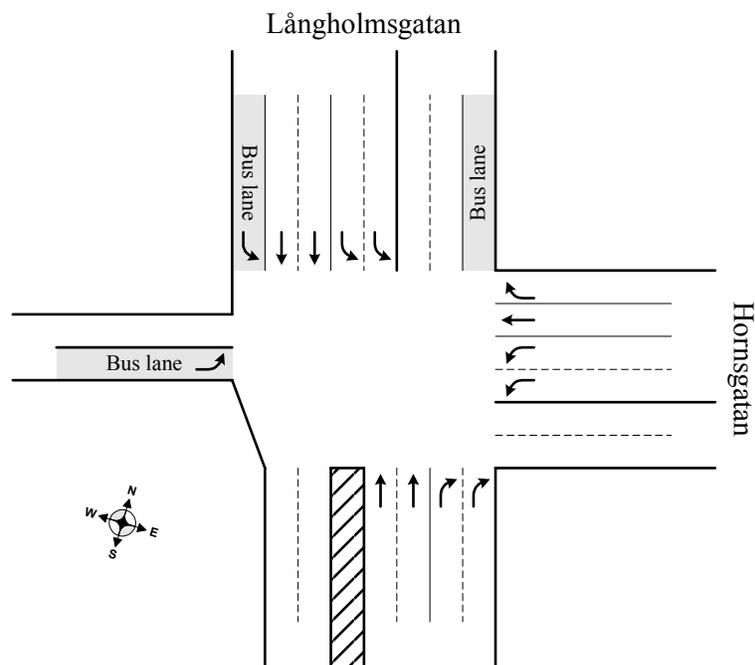


Figure 5-4: Lane configuration at Hornstull.

5.2 Simulation Setup

Modeling a network in MITSIMLab requires detailed input data. These include driving and travel behavior model parameters, geometric design data, travel demand data, and signal control logic and parameters. This information was provided by *Gatu- och Fastighetskontoret* (GFK) for use in preparing the case study network.

An extensive calibration of MITSIMLab to Stockholm driving conditions and a subsequent validation of these results were performed in 2000 (Ben-Akiva et al., 2002). This effort resulted in a set of parameters for the driving and travel behavior models in

MITSIM that should accurately represent conditions in Stockholm, and these parameters were used for the Hornstull case study.

Geometric data for the network was provided in the form of detailed maps and aerial ortho-photographs of the study area. The overall network structure was based on coordinates extracted from digital maps, and refinements were made using detailed intersection design plans, ortho-photographs, and observations of the site.

Traffic data was available in the form of 15-minute aggregate traffic counts on various days from 1998 to 2000, estimated 24-hour aggregate flows for all links from 1994, and turning movement counts at the Hornstull intersection made in 1988. Traffic counts aggregated by 15-minute intervals were available for both directions at three locations in the network, as shown in Figure 5-5. The most recent data that covered all three locations were from September 2000, and counts were extracted for the morning peak hour of 7:30 to 8:30 AM, the time period of interest.

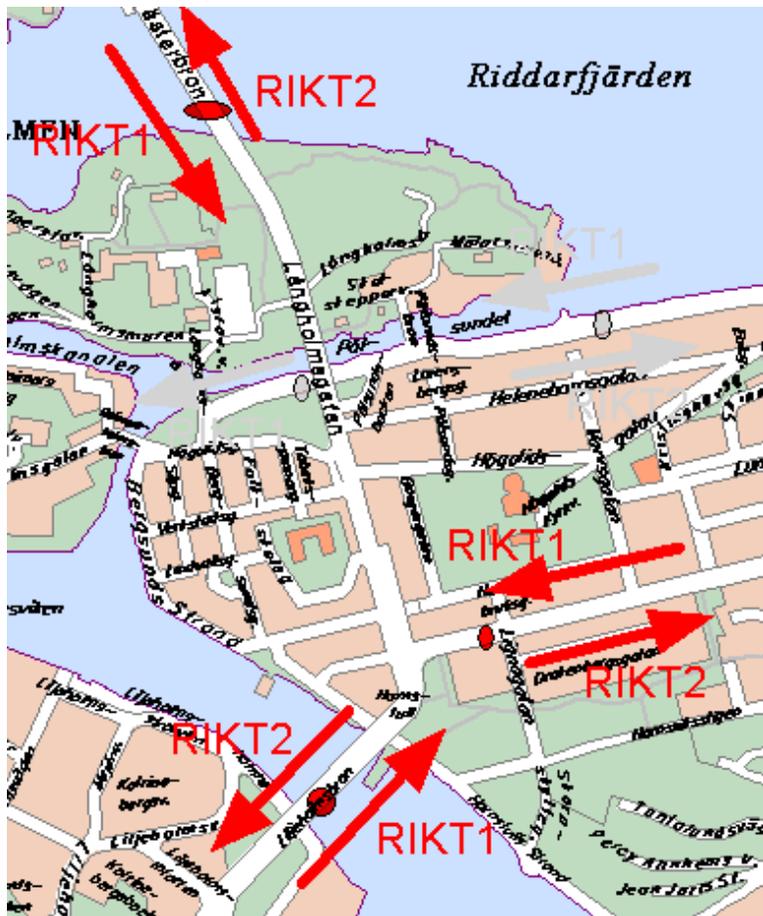


Figure 5-5: Traffic count locations.

Counts on other links were estimated using the 24-hour estimated flow map, scaling the flows to match the peak hour counts. With the link counts established, an origin-destination matrix consistent with those volumes was generated, with distribution of the demand based on the turning percentages from the 1988 counts and local knowledge of the network. Because the counts do not show much variation over the peak hour, a static origin-destination matrix for the peak hour was found to be adequate. A graphical representation of this O-D matrix is shown in Figure 5-6, with demand between origins and destinations represented by lines whose widths are proportional to the magnitude of the demand.

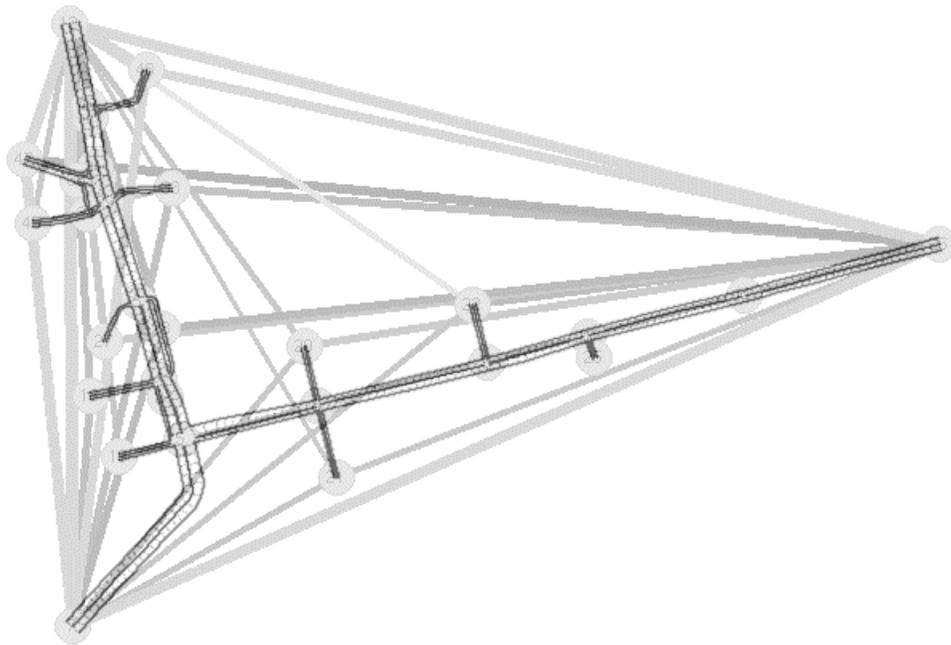


Figure 5-6: Graphical representation of peak-hour O-D matrix for Hornstull network.

Traffic signal timing plans that detail the operation of the signal controllers were provided by GFK. Details about the implementation of PRIBUSS in the network were not documented, but the logic and parameters specific to the network were determined through discussions with GFK traffic engineers (sample signal input files for the simulation can be found in Appendix C).

Additional information that was incorporated into the simulation setup includes data about truck, bus, and pedestrian demand. The percentage of heavy vehicles on the network during the peak hour was based on data provided from 1999. Bus routing and

demand were determined from maps and timetables published by *Storstockholms Lokaltrafik* (SL). Estimates of pedestrian demand were obtained from GFK in the form of pedestrian call frequency for each intersection. This level of detail is adequate because pedestrians are not explicitly modeled in MITSIMLab.

5.3 Experimental Design

For the simulation runs, MITSIMLab is run from 7:20 to 8:30 AM, corresponding to the morning peak hour preceded by a ten-minute “warm-up” period in which the network is loaded. Ten minutes is more than adequate for a complete loading of the network, since vehicle travel times through the network are well below this value. The simulation is run with a fixed demand level for the entire period.

The bus demands, based on the timetable published by SL, are four trips an hour for the red buses of Lines 40, 66, and 74 and eight trips an hour for the blue buses of Line 4. Because the terminals of the bus routes are outside of the study network, the entry times of the buses are allowed to vary stochastically to represent schedule deviations introduced outside of the simulation network. Within the network, the dwell times at the bus stops were also allowed to vary, distributed evenly between a lower bound of 15 seconds and an upper bound of 45 seconds.

The output collected from each run includes vehicle records and trajectory data. Vehicle records contain data on each vehicle processed by the simulation, including the vehicle’s identification code, type (e.g. car, truck or bus), origin, destination, departure time, arrival time, distance traveled, and average speed. Trajectory output records the identification code, position (i.e. distance from the origin), and speed of each vehicle after each second of simulation time. From this output data, the travel time and trajectory of each vehicle can be determined. Travel times can then be aggregated to obtain average travel times by vehicle type, origin-destination pair, or any other recorded variable. Trajectory data for an individual vehicle can be plotted to give a graphical indication of delays due to traffic signals or other factors.

The primary measure of effectiveness used in the evaluation is travel time through the network, aggregated across the vehicle groups of interest. Effects on priority-equipped buses are measured for all buses and also separated by direction of travel. Effects on

other traffic are measured for all vehicles and also separated by origin type, i.e. main arterial or cross-street. Travel time variability, as reflected by the standard deviation of travel time, is another useful measure of effectiveness, as this represents travel time reliability within the network. This measure is also aggregated across vehicle groups of interest.

Because MITSIM is a stochastic simulation model, its output will vary from run to run. The output of each run represents a sample, and therefore a number of simulation runs are required to obtain output statistics with a specified accuracy. The number of replications required to obtain reliable estimates of a simulated measure of interest y^s is given by:

$$R = \left(\frac{t_{\alpha/2} s}{e} \right)^2 \quad (5.1)$$

where

$t_{\alpha/2}$ = critical value of the t -distribution at a level of significance α ,

s = estimated value of the standard deviation of y^s ,

e = allowable error (in the same units as y^s).

To obtain an estimate of the standard deviations of each measure of interest, output from ten runs of the simulation was used. The measure of interest with the highest standard deviation determines the minimum number of replications. This measure was found to be the average bus travel time, with the large standard deviation caused by the relatively small number of observations per run. The estimated error, at a 95% confidence level, for ten replications is $\pm 1.1\%$, and this level of accuracy was deemed sufficient. For non-bus travel times, which are averages of many more observations, the corresponding error is $\pm 0.25\%$ at a 95% confidence level, indicating that these travel times should be quite consistent across the simulation runs.

The base case for the evaluation is the AM peak demand with no signal priority implemented. The PRIBUSS strategy is evaluated by applying the three available priority measures in all possible combinations. These runs are repeated with a demand at 115% of the AM peak demand to determine the effectiveness of PRIBUSS under heavy

traffic conditions. The sensitivity of the results to various parameters of PRIBUSS is also determined, and alternate settings are tested to evaluate their effectiveness.

5.4 Results

5.4.1 Travel Time Comparisons

Results from the comparison of priority strategies under the AM peak demand are shown in Table 5.1. This table presents the average vehicle travel times (in seconds) within the network for all vehicles, for priority-equipped buses, and for vehicles other than buses. Results for the buses are further broken down by direction of travel, i.e. northbound or southbound. Results for other vehicles are also presented for two sub-categories: vehicles that enter the network on the main arterials, and vehicles that enter from the side streets under signalized control. This distinction separates the vehicles that have right-of-way in the same signal phases as the buses from the vehicles that have right-of-way in conflicting phases. Table 5.2 presents these same results as a percentage change in travel time from the base scenario with no priority measures implemented.

Table 5.1: Average travel times for different priority implementations (peak demand).

Priority Implementation	Average Travel Time (seconds)						
	All Vehicles	Buses			Other Vehicles		
		All	South-bound	North-bound	All	Side Origins	Arterial Origins
None	118	321	347	295	118	111	125
Extension	118	320	341	291	117	111	124
Shortening	118	319	342	292	118	113	123
Insertion	118	312	321	300	118	112	124
Extension & Shortening	118	313	334	290	118	113	122
Extension & Insertion	117	309	325	292	117	112	123
Shortening & Insertion	118	288	288	286	118	114	122
All	118	287	286	287	117	113	122

Table 5.2: Change in average travel times for different priority implementations (peak demand).

Priority Implementation	Difference in Average Travel Time from Base Case (%)						
	All Vehicles	Buses			Other Vehicles		
		All	South-bound	North-bound	All	Side Origins	Arterial Origins
None	-	-	-	-	-	-	-
Extension	0.0%	-0.3%	-1.7%	-1.4%	-0.8%	0.0%	-0.8%
Shortening	0.0%	-0.6%	-1.4%	-1.0%	0.0%	1.8%	-1.6%
Insertion	0.0%	-2.8%	-7.5%	1.7%	0.0%	0.9%	-0.8%
Extension & Shortening	0.0%	-2.5%	-3.7%	-1.7%	0.0%	1.8%	-2.4%
Extension & Insertion	-0.8%	-3.7%	-6.3%	-1.0%	-0.8%	0.9%	-1.6%
Shortening & Insertion	0.0%	-10.3%	-17.0%	-3.1%	0.0%	2.7%	-2.4%
All	0.0%	-10.6%	-17.6%	-2.7%	-0.8%	1.8%	-2.4%

Aggregating across all vehicles, no significant difference in travel time is caused by the implementation of any priority strategy. Breaking down the effects by vehicle type, however, shows that buses do benefit from a reduction in average travel time with the PRIBUSS implementation. The PRIBUSS strategy as implemented (i.e. with all three priority measures allowed) results in over a 10% reduction in bus travel time from the case with no priority implemented. No significant effect on travel time for other vehicles is detectable.

Impact on Buses

Separating the buses by direction of travel, as presented graphically in Figure 5-7, shows that travel times for southbound buses are significantly higher than for northbound buses. Observation of the simulation shows that this is due to the signal settings at Hornstull. Because the southbound left-turning buses require a separate signal group from other left-turning vehicles, only six seconds out of the 100-second cycle is given to this signal group in the absence of priority. In comparison, northbound right-turning buses have green for 49.5 seconds out of the cycle, creating much less of a bottleneck.

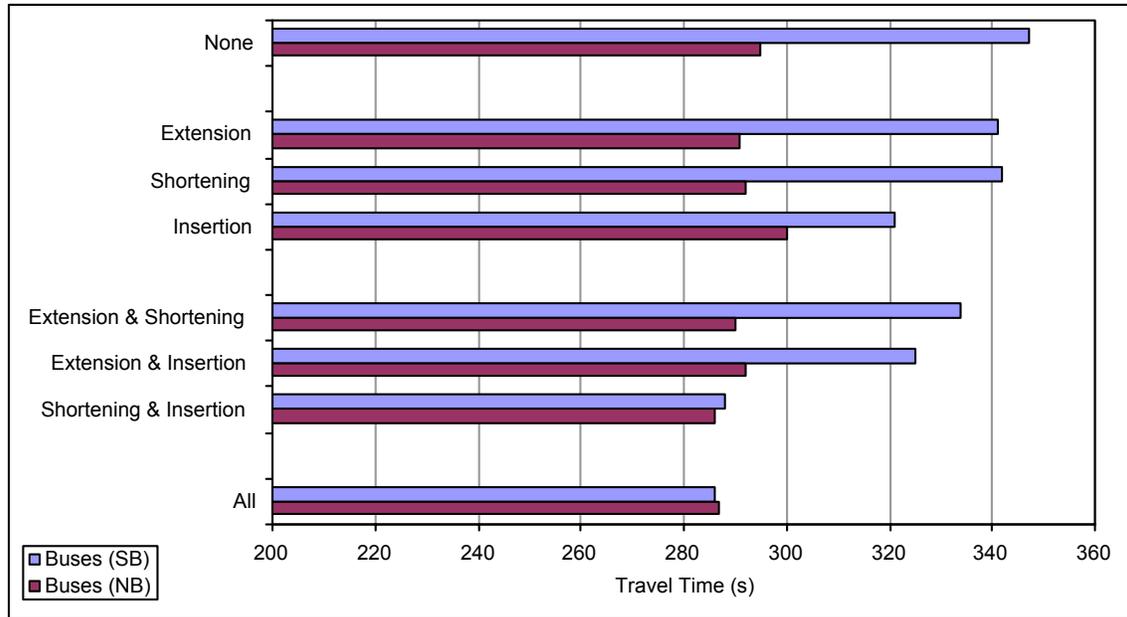


Figure 5-7: Average bus travel times by priority implementation (peak demand).

For southbound buses at Hornstull, implementing phase insertion gives additional green time but, more importantly, breaks up the 90-second block of red time that the signal group experiences without priority. This effect is evident in the results for the individual priority implementations, in which phase insertion reduces average travel time by 26 seconds while shortening and extension only reduce travel time by five and six seconds, respectively. As insertion is only allowed for that one signal group in the entire network, the large reduction in travel time is being generated at the Hornstull intersection alone.

The results for southbound buses also show that the effect of combinations of priority measures is greater than the sum of the individual effects. This implies that the individual priority actions are not independent, as is expected on a network with closely spaced intersections and coordinated signal timings. In fact, observation of the simulation shows that the interactions between adjacent intersections, especially due to signal coordination, have a significant impact on the progression of the bus. This is especially evident when individual priority measures are used alone, as often an advantage granted by priority action at one intersection may be lost at a downstream intersection where priority can not be granted. For example, if a bus has green extended at one intersection but then arrives too late for extension at the next intersection, much of

the travel time savings from the first action is being lost by waiting at the second intersection. If shortening is allowed at the second intersection, however, the time savings from the first priority action can be maintained.

This effect can be demonstrated by comparing trajectory plots for individual vehicles. Figure 5-8 illustrates the trajectories of a southbound bus with and without implementation of signal priority. The trajectories are from two different runs with all conditions identical other than the signal priority implementation. The horizontal lines represent the location of traffic signals on the network, their distance measured from the bus entry point in the network. The locations of the bus stops can be inferred from the horizontal portions of the trajectories not located at signals. For this comparison, dwell time for the buses was fixed at 30 seconds at each stop in order to be consistent between the runs. In this example, priority reduces delay at nearly every signal, resulting in a cumulative travel time savings of nearly 80 seconds.

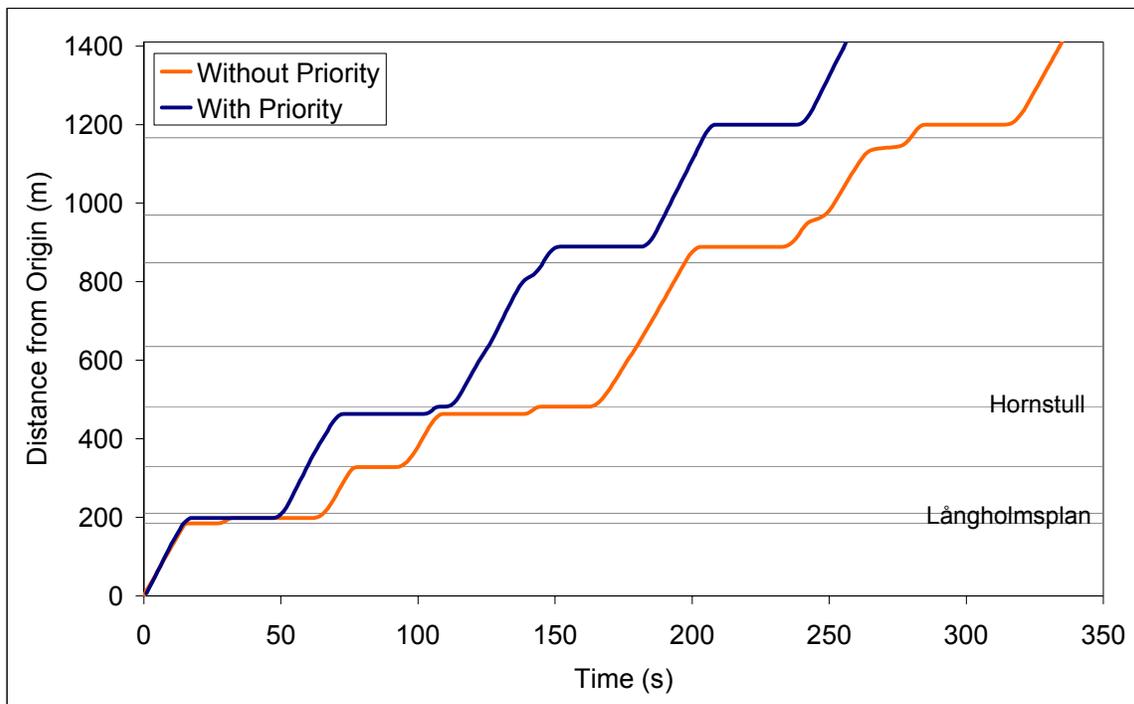


Figure 5-8: Comparison of bus trajectories showing benefit from priority.

The need for signal priority is limited somewhat by the signal coordination that is implemented along the Hornsgatan-Långholmsgatan corridor. Although the presence of bus stops means that buses are unable to move continuously through the network as part

of the vehicle platoon that the signal progression creates, buses can benefit between stops from the green band that the coordination scheme provides. For example, if a bus departs a bus stop and enters the traffic flow within the green band, it should be able to travel to the next stop as part of the vehicle platoon and not require the use of any priority actions. Visual observation of the simulation shows this to be the case for most northbound buses on the network. Because side street demand is quite low, ample green time is given to the primary street movements, which means that the green band for vehicle progression is quite wide. Most buses, therefore, move within one green band between two adjacent bus stops and then join the next green band after stopping for passengers. Because of the favorable conditions provided by the signal coordination, the extra benefit of signal priority is minimal. For northbound buses, priority offers little benefit, with travel time savings no greater than 3%. Implementing phase insertion alone actually increases their travel time, as this action serves southbound buses at Hornstull by taking time from the other approaches, including that of the northbound buses. However, this effect is negligible when other priority measures are implemented in combination.

Figure 5-9 shows an extreme example of a bus trajectory where priority offers no travel time benefit at all. The signal coordination is such that this vehicle can move between bus stops with nearly no delays from the traffic signals. Signal priority, therefore, is not able to reduce the travel time any further. In reality, variability of dwell times and queue lengths will mean that replication of this trajectory will be rare, but clearly signal priority can not have an effect when delay from signals is minimal.

Impact on Other Vehicles

For non-bus traffic, no aggregate effect on travel time is discernable. Separating the vehicles by point of origin, i.e. those entering the network on the arterial versus those entering from side streets under signal control, reveals that priority does impact travel time, however. Side streets see no significant effect from extension or insertion actions, but are somewhat impacted by shortening, with nearly an increase in average travel time of approximately 2%. This effect is canceled out in the aggregate analysis, however, by the slight benefit seen by the greater volume of mainline traffic, resulting in no net change in travel time for all vehicles.

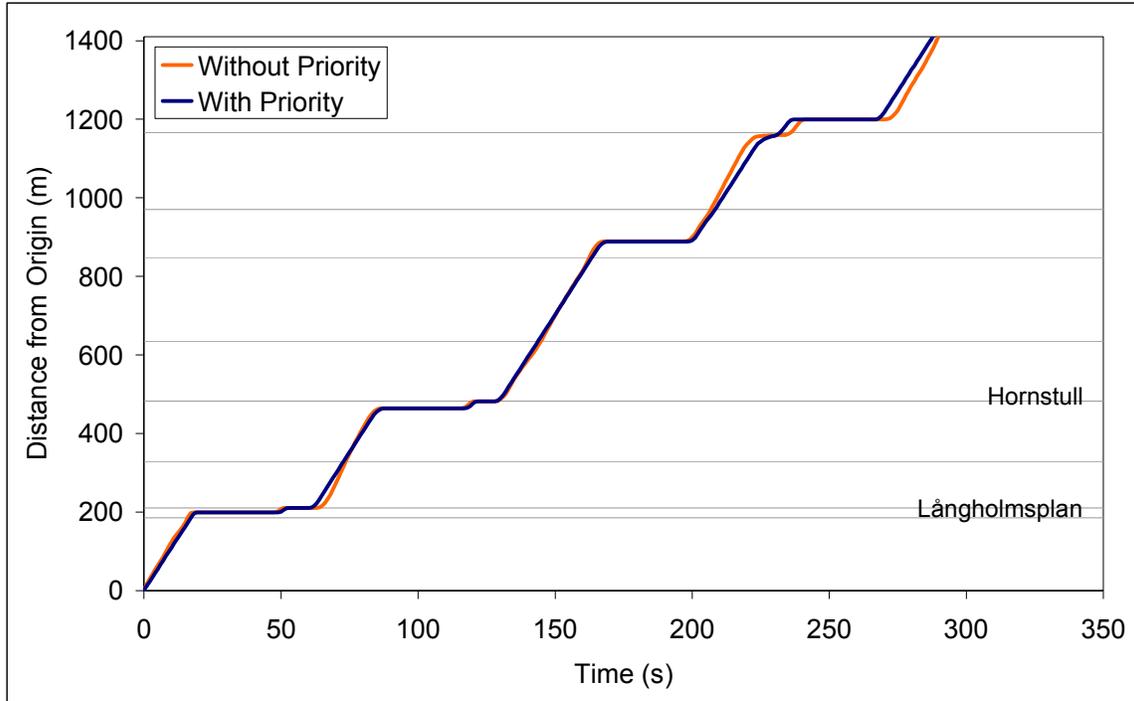


Figure 5-9: Comparison of bus trajectories showing no benefit from priority.

5.4.2 Travel Time Variability

To examine the effect of priority on travel time variability, the standard deviation of the travel times is compared for each scenario. Table 5.3 shows the standard deviations for each scenario, again segmented by vehicle type and origin. Table 5.4 presents these same results as a percentage change in standard deviation from the base scenario of no priority implementation. The results follow the same general pattern as the travel time results, with a reduction in average travel time accompanied by a reduction in variability and an increase in travel time accompanied by an increase in variability. This is a logical result because the effect of priority is limited to buses at the upper range of travel times. For example, the fastest bus trips that occur without priority are those in which the bus never happens to encounter a red light, which is a possibility because of the signal coordination. Adding priority will not change this vehicle's travel time, because the signals do not impact the bus. Priority will, however, reduce the travel time of the longest bus trips, which are those where signal delays are greatest. By shortening travel

times for the longest trips, priority acts to reduce the spread and therefore the variability of travel time.

Table 5.3: Standard deviations of travel time for different priority implementations (peak demand).

Priority Implementation	Standard Deviation of Travel Time (seconds)						
	All Vehicles	Buses			Other Vehicles		
		All	South-bound	North-bound	All	Side Origins	Arterial Origins
None	47	45	41	29	46	39	53
Extension	47	40	36	24	46	39	52
Shortening	47	42	36	26	46	40	52
Insertion	47	35	37	26	46	39	52
Extension & Shortening	47	38	37	26	46	40	52
Extension & Insertion	46	34	31	27	45	39	51
Shortening & Insertion	47	27	21	32	46	40	51
All	46	24	20	27	45	39	51

Table 5.4: Change in standard deviations of travel time for different priority implementations (peak demand).

Priority Implementation	Difference in Std. Deviation of Travel Time from Base Case (%)						
	All Vehicles	Buses			Other Vehicles		
		All	South-bound	North-bound	All	Side Origins	Arterial Origins
None	-	-	-	-	-	-	-
Extension	0.0%	-11.1%	-12.2%	-17.2%	0.0%	0.0%	-1.9%
Shortening	0.0%	-6.7%	-12.2%	-10.3%	0.0%	2.6%	-1.9%
Insertion	0.0%	-22.2%	-9.8%	-10.3%	0.0%	0.0%	-1.9%
Extension & Shortening	0.0%	-15.6%	-9.8%	-10.3%	0.0%	2.6%	-1.9%
Extension & Insertion	-2.1%	-24.4%	-24.4%	-6.9%	-2.2%	0.0%	-3.8%
Shortening & Insertion	0.0%	-40.0%	-48.8%	10.3%	0.0%	2.6%	-3.8%
All	-2.1%	-46.7%	-51.2%	-6.9%	-2.2%	0.0%	-3.8%

This reduction in variability is most striking for southbound bus traffic, which sees a reduction in standard deviation of over 50% with all priority measures implemented. Such a result has significant benefits for bus reliability and schedule adherence, factors which can be more important than travel time for transit level of service. Passenger

waiting time, for example, is minimized with even headways between vehicles, and increased schedule adherence reduces average passenger wait times. Travel time reliability is another important measure and is often a key factor in mode choice. Reducing travel time variability has positive effects on both of these measures, and this should be recognized as a key benefit of signal priority.

5.4.3 Effects of Increased Demand

Results from the travel time comparisons of priority strategies under the high demand scenario are shown in Table 5.5 in terms of absolute value and in Table 5.6 in terms of percentage change from the no priority base case. While travel times are longer, as expected, the results are quite similar to the base demand case in terms of percentage change in travel time due to priority. Northbound buses are helped slightly more by priority under higher demand, however, with travel time improvements of up to 5%, as compared to 3% in the base demand scenario. Negative impacts to side-street traffic are still quite low under all priority scenarios.

Table 5.5: Average travel times for different priority implementations (high demand).

Priority Implementation	Average Travel Time (seconds)						
	All Vehicles	Buses			Other Vehicles		
		All	South-bound	North-bound	All	Side Origins	Arterial Origins
None	123	331	352	305	123	115	132
Extension	123	318	342	298	123	115	131
Shortening	123	319	341	297	123	116	131
Insertion	123	313	328	295	123	116	131
Extension & Shortening	123	309	332	289	123	117	130
Extension & Insertion	123	311	321	296	123	116	130
Shortening & Insertion	123	293	297	290	123	118	129
All	123	290	290	291	123	118	129

Table 5.6: Change in average travel times for different priority implementations (high demand).

Priority Implementation	Difference in Average Travel Time from Base Case (%)						
	All Vehicles	Buses			Other Vehicles		
		All	South-bound	North-bound	All	Side Origins	Arterial Origins
None	-	-	-	-	-	-	-
Extension	0.0%	-3.9%	-2.8%	-2.3%	0.0%	0.0%	-0.8%
Shortening	0.0%	-3.6%	-3.1%	-2.6%	0.0%	0.9%	-0.8%
Insertion	0.0%	-5.4%	-6.8%	-3.3%	0.0%	0.9%	-0.8%
Extension & Shortening	0.0%	-6.6%	-5.7%	-5.2%	0.0%	1.7%	-1.5%
Extension & Insertion	0.0%	-6.0%	-8.8%	-3.0%	0.0%	0.9%	-1.5%
Shortening & Insertion	0.0%	-11.5%	-15.6%	-4.9%	0.0%	2.6%	-2.3%
All	0.0%	-12.4%	-17.6%	-4.6%	0.0%	2.6%	-2.3%

To amplify the effects of the increased demand, an additional scenario with 130% demand was also tested for the “none” and “all” priority implementations. A comparison of bus travel times under the three different demand scenarios is shown in Figure 5-10, which presents results for the full priority implementation and the no-priority base case. This comparison shows that travel times for southbound buses are not significantly affected by increased demand, even under the extreme demand scenario. Observation of the simulation shows that this is due to the presence of the bus lane on Långholmsgatan, which allows the bus to bypass the congestion created by the signal at Hornstull. Past Hornstull, the increased demand does not have a major effect, so travel time remains unchanged. Northbound buses, however, are more impacted by the increased demand. Again, observation shows that congestion at Hornstull is the primary cause, as this leads to slower speeds and longer queues on Hornsgatan upstream of the intersection. Because Hornsgatan does not have a bus lane, northbound buses can not bypass this congestion, resulting in a significant increase in travel times. Priority implementation reduces northbound bus travel times by over 17%, mostly by clearing signal queues ahead of approaching buses.

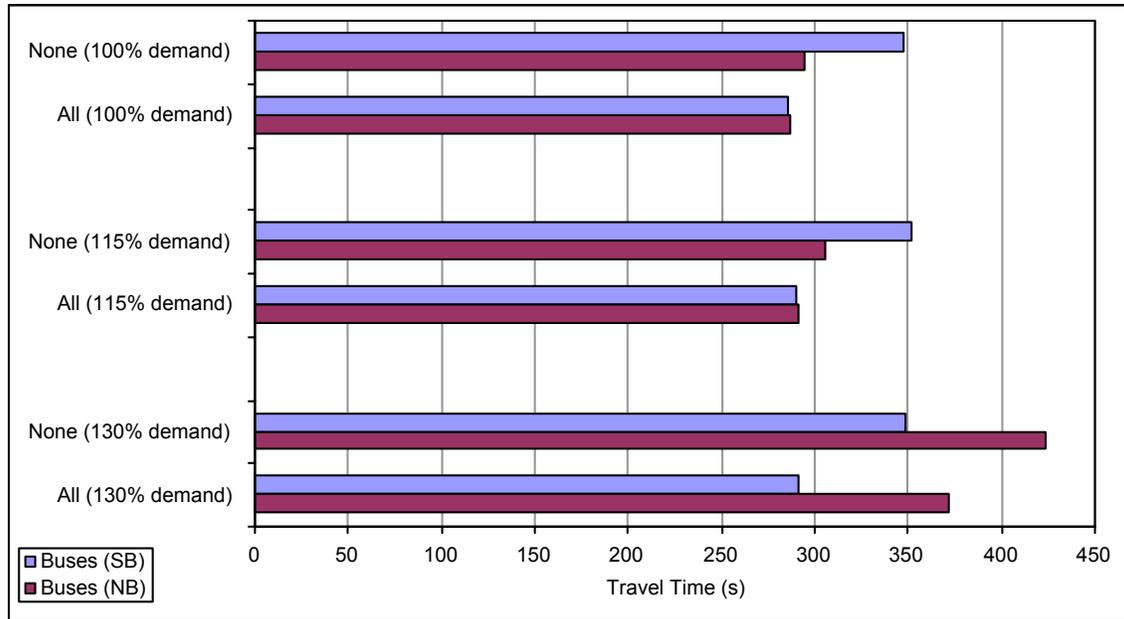


Figure 5-10: Average travel time comparison under different demand scenarios.

5.4.4 Parameter Sensitivity

As discussed in the previous chapter, the most crucial parameter for implementation of PRIBUSS is the guaranteed time for the non-priority phases. This parameter directly affects the other key settings of PRIBUSS, including the time window in which priority can be called and the termination point for a priority action currently underway. The implementations tested in the previous sections use the minimum green time for the signal group as the guaranteed time, a value of either four or six seconds for the intersections on the study network. Under normal operation, however, side-street phases typically receive 15-20 seconds of green. A more generous guaranteed time, therefore, may be required in order to provide adequate service to the side streets, especially under high demand. In addition, if pedestrian crossings of the arterial are concurrent with side-street movements, guaranteed time will have to be longer to permit adequate clearance time for pedestrians.

The effect of an increased guaranteed time was tested by running trials with a 10-second guaranteed time for all non-priority phases. The existing priority implementation (extension, shortening, and insertion) was tested under the three demand scenarios of 100%, 115%, and 130% of the AM peak-hour demand. The comparison of these results

with the non-priority base case and the minimum guaranteed-time implementation is shown in Table 5.7 in terms of absolute value and in Table 5.8 in terms of percent change.

Table 5.7: Average travel times for different priority implementations and demand levels.

Demand Level	Priority Implementation (Guaranteed Time)	Average Travel Time (seconds)						
		All Vehicles	Buses			Other Vehicles		
			All	South-bound	North-bound	All	Side Origins	Arterial Origins
100% (AM Peak)	None	118	321	347	295	118	111	125
	All (10 s)	117	295	298	294	117	112	123
	All (min.)	118	287	286	287	117	113	122
115%	None	123	331	352	305	123	115	132
	All (10 s)	122	300	305	291	122	116	129
	All (min.)	123	290	290	291	123	118	129
130%	None	163	379	348	423	162	132	198
	All (10 s)	158	340	304	388	158	135	185
	All (min.)	156	329	291	372	156	142	173

Table 5.8: Change in average travel times for different priority implementations and demand levels.

Demand Level	Priority Implementation (Guaranteed Time)	Difference in Average Travel Time from Base Case (%)						
		All Vehicles	Buses			Other Vehicles		
			All	South-bound	North-bound	All	Side Origins	Arterial Origins
100% (AM Peak)	None	-	-	-	-	-	-	-
	All (10 s)	-0.8%	-8.1%	-14.1%	-0.3%	-0.8%	0.9%	-1.6%
	All (min.)	0.0%	-10.6%	-17.6%	-2.7%	-0.8%	1.8%	-2.4%
115%	None	-	-	-	-	-	-	-
	All (10 s)	-0.8%	-9.4%	-13.4%	-4.6%	-0.8%	0.9%	-2.3%
	All (min.)	0.0%	-12.4%	-17.6%	-4.6%	0.0%	2.6%	-2.3%
130%	None	-	-	-	-	-	-	-
	All (10 s)	-4.2%	-12.1%	-12.7%	-11.9%	-3.4%	2.7%	-10.4%
	All (min.)	-5.9%	-15.6%	-16.4%	-17.3%	-5.1%	9.0%	-20.0%

Under normal peak hour traffic, the increased guaranteed time does slightly lessen the impact on side street traffic, but it also eliminates the benefit to the northbound buses altogether. The benefit to southbound buses is still quite large, however, since most of the benefit comes from phase insertion, which is not affected by the increased guaranteed time for the side streets.

For the 115% demand scenario, the benefit to northbound buses is maintained with the increased guaranteed time, as is most of the benefit to southbound buses. The

advantage of the longer guaranteed time for side-street traffic, however, is still very minimal.

The guaranteed time becomes an important factor under the 130% demand scenario. Applying priority with the minimum guaranteed times under the extreme demand scenario has a very significant impact on side street traffic, with a travel time increase of 9% over the no-priority case. The 10-second guaranteed time reduces this increase to less than 3% while still maintaining 77% and 68% of the travel time savings for southbound and northbound buses, respectively. This shows that a longer guaranteed time may be appropriate under increased demand conditions in order to maintain adequate service to side streets. However, if traffic demand increases on the arterial but not on the side streets, a greater benefit will be obtained by using the shortest possible guaranteed time, ensuring maximum travel time savings for buses and other arterial traffic.

5.5 Recommendations

The simulation results demonstrate that the PRIBUSS implementation has a significant benefit to buses in the Hornstull network without much negative effect on other vehicles. A general conclusion to be drawn is that all priority actions should be allowed whenever feasible, as this will take advantage of the interactions among them to produce a greater net benefit. If impacts to non-priority signal groups are too great, increasing the guaranteed time for those groups instead of prohibiting specific priority actions will lessen the impact while still maintaining travel time savings for the buses.

Another conclusion from the simulation study is that the effectiveness of bus signal priority is very dependent on local factors. Well-planned background signal timings, for example, may allow most buses to move with little impedance from the signals, so the additional benefit of active transit priority will be small in terms of average travel time. However, the benefit in terms of travel time variability may make the implementation worthwhile, by improving travel time reliability and schedule adherence along the length of the bus route. Other local factors that may limit the need for priority actions include geometric design elements, such as bus lanes and bus stop locations, and average bus stop dwell times. These factors will affect how buses progress through the network in relation

to the coordination green band and may allow buses to take advantage of arterial progression, reducing the number of calls for priority.

Due to the importance of such factors, it is difficult to draw up general rules for implementation of signal priority. For this reason, simulation is an ideal tool for the testing of alternate strategies and the evaluation of their effectiveness.

Chapter 6

Conclusions

6.1 Summary

This research has resulted in a major enhancement to the signal control capabilities of MITSIMLab. A new controller that addresses the limitations of the pre-existing controllers in MITSIMLab has been developed. It has been designed with a generic control logic that allows the simulation of a wide range of control strategies. Instead of requiring each signal phase to be specified explicitly, the generic logic allows each vehicle stream to be controlled independently, thus allowing more flexibility in the control logic specification.

This logic of this generic controller is based on a set of basic logic elements that are common to all signal control strategies. These logic elements define actions that can be initiated and the criteria that must be met in order for these actions to occur. Inputs for the logic include internal timers, detector states, and the states of other signals and controllers. Specific control strategies can be simulated by combining and ordering these basic elements. This flexibility in defining the logic allows the generic controller to simulate both pretimed and actuated control and to model different control strategies, such as isolated intersection control, arterial coordination, and network control. While the generic controller supports a wide range of controller types, its modular structure also allows additional logic elements and functions to be added easily. This allows specialized functions of specific control strategies to be implemented, and it also provides a framework for implementation of other advanced traffic management strategies.

This capability is utilized for the modeling of PRIBUSS, an advanced transit signal priority strategy developed for use in Stockholm, Sweden. PRIBUSS is an active priority strategy that seeks to reduce signal delays to transit vehicles by detecting those vehicles upstream of a traffic signal and adjusting the signal settings in real time. The strategy is designed to operate within the constraints of coordinated signal operation, with the objective of providing transit priority with as little disturbance to the coordination as possible.

The generic controller is used as a foundation for the modeling of this strategy. Functions specific to PRIBUSS are incorporated by means of specialized conditions and status variables. With the functions of PRIBUSS simulated by the generic controller, MITSIMLab is applied to an urban arterial network in Stockholm where the PRIBUSS strategy is implemented. The simulation laboratory is used to evaluate the effectiveness of the existing PRIBUSS implementation under various conditions and to compare its performance under alternate parameter settings.

6.2 Findings

On the study network, the existing PRIBUSS implementation is found to be beneficial in reducing bus delay, providing buses with average travel time savings of up to 18% over the base case of no priority implementation. Impacts on other traffic are insignificant, except for vehicles entering the network from cross streets under signalized control. These vehicles experience an increase in delay of less than 2%. Under a high demand scenario of 115% of the AM peak volumes, similar benefits are found, with cross-street delays still under 3%.

Another benefit from the existing implementation is the reduction in travel time variability for buses that accompanies the reduction in travel times. This benefit is quite significant, providing reductions in average standard deviation of up to 50% in the AM peak scenario and 37% in the high-demand scenario.

The existing implementation utilizes all three priority actions that can be implemented within the network's signal plan: green extension, phase shortening, and phase insertion. Evaluation of alternate implementation strategies using subsets of these actions shows that the individual benefits are not independent, as the benefit from the

combination of actions is greater than the sum of the benefits from the actions applied individually. This implies that the interaction between intersections is a quite significant. With closely spaced intersections, therefore, using priority actions in combination is most efficient.

If impacts on cross-street traffic are a concern, eliminating the use of one of the three priority actions can reduce cross-street delays, but this will also reduce the benefit to buses significantly. A more efficient way to mitigate the cross-street delays is by adjusting the guaranteed time for the non-priority phases. Under extreme demand conditions, for example, increasing the guaranteed time from the minimum of four or six seconds to 10 seconds is found to reduce cross-street delays by nearly 70% while still maintaining over 75% of the travel time savings for buses.

The simulation also demonstrates that the benefit obtained from priority is very dependent on factors external to the priority settings, such as traffic conditions, bus facility design, and the background signal settings. Signal priority is found to be most beneficial in congested conditions, where queues at signals cause significant delay in the absence of priority. Under less congested conditions, signal delay may not be large enough to justify the priority implementation. Similarly, infrastructure elements such as bus lanes, which allow queues to be bypassed, may act independently to reduce signal delay, therefore reducing the contribution of the signal priority implementation.

The most significant factor affecting the benefit of priority, however, is the underlying signal settings over which priority is applied. If the phase splits favor transit vehicles with a higher percentage of green time, the number of calls for priority will be reduced because buses will be less likely to encounter a red signal. A similar effect occurs under signal coordination, in which the buses benefit from arterial progression and require fewer priority calls.

A general conclusion from the evaluation is that the effectiveness of transit signal priority is very site- and condition-specific. Therefore, results from one application may not necessarily apply in different situations. For this reason, simulation is a valuable tool for the design and evaluation of priority strategies, as it can be used to determine if and how such strategies can be most effectively implemented.

6.3 Future Research

The generic controller implemented within MITSIMLab is currently able to simulate most basic control types and strategies that are in standard use. However, this controller is also designed as a research tool for the evaluation of advanced control strategies. Adaptive control is a key area of research that can benefit from evaluation with simulation, as the cost of implementing such systems can limit the scope of field evaluations. Similarly, network control strategies can be evaluated in a simulation environment, where measures of effectiveness can be collected much more easily than in the field.

In terms of the Stockholm application, an evaluation of a network with a longer bus corridor would be beneficial, as the longer travel times would allow better measures of schedule adherence and would allow studies of behavior such as bus bunching. Expanding the network to include multiple bus routes would also allow the modeling of more complex situations, such as the management of simultaneous priority calls on conflicting signal groups.

The use of more advanced transit priority strategies is another area in which MITSIMLab can be applied. Conditional priority logic, for example, is under consideration for use in Stockholm, with buses being granted priority on the basis of factors such as schedule deviation and passenger load. Adaptive signal control is also being researched and field-tested in the city, and incorporating bus priority into such a system is a further area of study. MITSIMLab, with the enhanced control capabilities provided by the generic controller, can now serve as a laboratory for the design and evaluation of these and other advanced traffic management systems.

Appendix A

Logic Conditions for Generic Traffic Signal Controller

The tables in this section detail the logic conditions that can be used to specify the operational logic of the generic controller (see Chapter 3 for complete description). For each condition, the following information is given:

- *Code*: a unique three-digit code that identifies the condition in the signal input file.

The first digit of the code indicates the condition type:

0XX = general

1XX = change

2XX = hold

3XX = skip

- *Description*: a brief description of the logic of the condition.
- *Number of Parameters*: the number of input parameters that the condition takes. “N” indicates a number that will vary based on the input specification.
- *Parameters*: the input parameters required for specifying the condition.

Table A.1 and Table A.2 list the conditions used for basic signal control functions, as described in Section 3.2.4. Table A.3 lists the conditions used for transit signal priority functions, as described in Section 4.2.1.

Table A.1 : Basic logic conditions (general and change conditions).

Code	Description	Number of Parameters	Parameter [0]	Parameter [1]	Parameter [2]	Parameter [3]	Parameter [4]
000's (General Conditions):							
001:	Next Period - sets [next period] to follow [current period]	2	current period	next period			
002:	Delayed Start (Offset) - group starts with signals blank - is initialized after a delay of [lag time]	1	lag time				
003:	Advance Start - group is started [lead time] seconds into the initial period	1	lead time				
004:	Gap Timer - resets gap timer if any of [detectors] is actuated	N	detector[s][N]				
100's (Change Conditions):							
101:	No Demand - [current period] changes to [next period] if none of [detectors] is occupied	N+2	current period	next period	detector[s][N]		
102:	Force-Off - [current period] changes to [next period] if "time in cycle" equals [force-off time] where "time in cycle" = (time_since_initialization % [cycle length]) - [offset]	5	current period	next period	force-off time	cycle length	offset
103:	Maximum Time - [current period] changes to [next period] if active for more than [maximum time]	3	current period	next period	maximum time		
104:	Follow Other Group - [current period] changes to [next period] [delay] seconds after [sync group] changes to [sync period]	5	current period	next period	sync group	sync period	delay

Table A.2: Basic logic conditions (hold and skip conditions).

Code	Description	Number of Parameters	Parameter [0]	Parameter [1]	Parameter [2]	Parameter [3]	Parameter [4]
200s (Hold Conditions):							
201:	Minimum Time - [current period] held if active for less than [minimum time]	2	current period	minimum time			
202:	Extension Time - [current period] held if gap timer exceeds [extension time] - not held if period has been active for more than [maximum time]	3	current period	extension time	maximum time		
205:	Complementary Groups - [current period] held if [complementary groups] still active	N+1	current period	complementary groups[N]			
206:	No Conflicting Calls - [current period] held if none of [detectors] is actuated	N+1	current period	detectors[N]			
207:	Conflict Clearance - [current period] held if [conflicting groups] not completed - held until conflicts have been complete for [clearance time]	N+2	current period	clearance time	conflicting groups[N]		
208:	Hold Indefinitely - [current period] held	1	current period				
210:	Hold in window - [current period] held if "time in cycle" is between [start] and [end] where "time in cycle" = (time_since_initialization % [cycle length]) - [offset]	5	current period	start window	end window	cycle length	offset
300s (Skip Conditions):							
301:	Detector Occupied - Skip next condition if any of [detectors] is occupied	N	detectors[N]				
302:	Detector Not Occupied - Skip next condition if none of [detectors] is occupied	N	detectors[N]				
303:	Skip next [num. to skip] conditions if [group]'s [flag] is [on/off]	4	num. to skip	group	flag	on/off	

Table A.3: PRIBUSS logic conditions.

Code	Description	Number of Parameters	Parameter [0]	Parameter [1]	Parameter [2]	Parameter [3]	Parameter [4]	Parameter [5]	
000s (General Conditions):									
006:	Bus priority check-in - increments check-in counter if [detector] is actuated	1	detector						
007:	Bus priority check-out - decrements check-in counter if [detector] is actuated	1	detector						
008:	Bus priority: extension - set flag if bus checks-in and if "time in cycle" is between [start window] and [end window] - unset flag if check-in counter is zero or if "time in cycle" is greater than [end extension] where "time in cycle" = (time_since_initialization % [cycle length]) - [offset]	5	start window	end window	end extension	cycle length	offset		
009:	Bus priority: insertion - set flag if check-in counter is > 0 and if "time in cycle" is between [start window] and [end window] - unset flag if check-in counter is zero or if "time in cycle" is greater than [end insertion] where "time in cycle" = (time_since_initialization % [cycle length]) - [offset]	5	start window	end window	end insertion	cycle length	offset		
010:	Set bus insertion flags for preceding groups - only valid in window - resets flags at reset time	N+5	start window	end window	reset time	cycle length	offset	groups[N]	
011:	Set bus insertion flags for following groups - only valid in window - resets flags at reset time	N+5	start window	end window	reset time	cycle length	offset	groups[N]	
012:	Bus priority: early start - set flag if check-in counter is > 0 and if "time in cycle" is between [start window] and [end window] - unset flag if check-in counter is zero or if "time in cycle" is greater than [reset time] where "time in cycle" = (time_since_initialization % [cycle length]) - [offset]	5	start window	end window	reset time	cycle length	offset		
013:	Set shortened flags for preceding groups - only valid in window - resets flags at reset time	N+5	start window	end window	reset time	cycle length	offset	groups[N]	
014:	Reset check-in counter - resets counter to zero at specified time in cycle	3	reset time	cycle length	offset				

Appendix B

Example Signal Input File: Phase-Based Specification

This section presents an example signal timing plan for an eight-phase dual-ring traffic signal controller, as described in Section 3.4.1, and the corresponding MITSIMLab signal input file. The timing plan is for actuated-coordinated operation (see Section 3.3.2 for description). Table B.1 shows the signal timing data for each phase.

Table B.1: Signal timing data for eight-phase controller.

Phase:	1	2	3	4	5	6	7	8
type	fixed	sync		fixed	fixed	sync		
minimum green (sec)			5				5	10
extension time (sec)			3				3	5
yellow time (sec)	3	4	3	4	3	4	3	4
red clearance (sec)	1	1	1	1	1	1	1	1
split (%)	20	35	20	25	20	35	20	25
split time (sec)	24	42	24	30	24	42	24	30
phase sequence: ring 1	1	2	3	4				
ring 2					2	1	3	4
cycle length = 120 sec offset = 20 sec Notes: 1) Sync phases and fixed phases are not subject to extension, so minimum green and extension time are not needed. 2) Splits are given as percentages of the cycle length; multiplying the split by the cycle length gives the split time.								

Figure B-1 shows the phase diagram for the controller with all necessary timing data. The corresponding MITSIMLab signal input file follows.

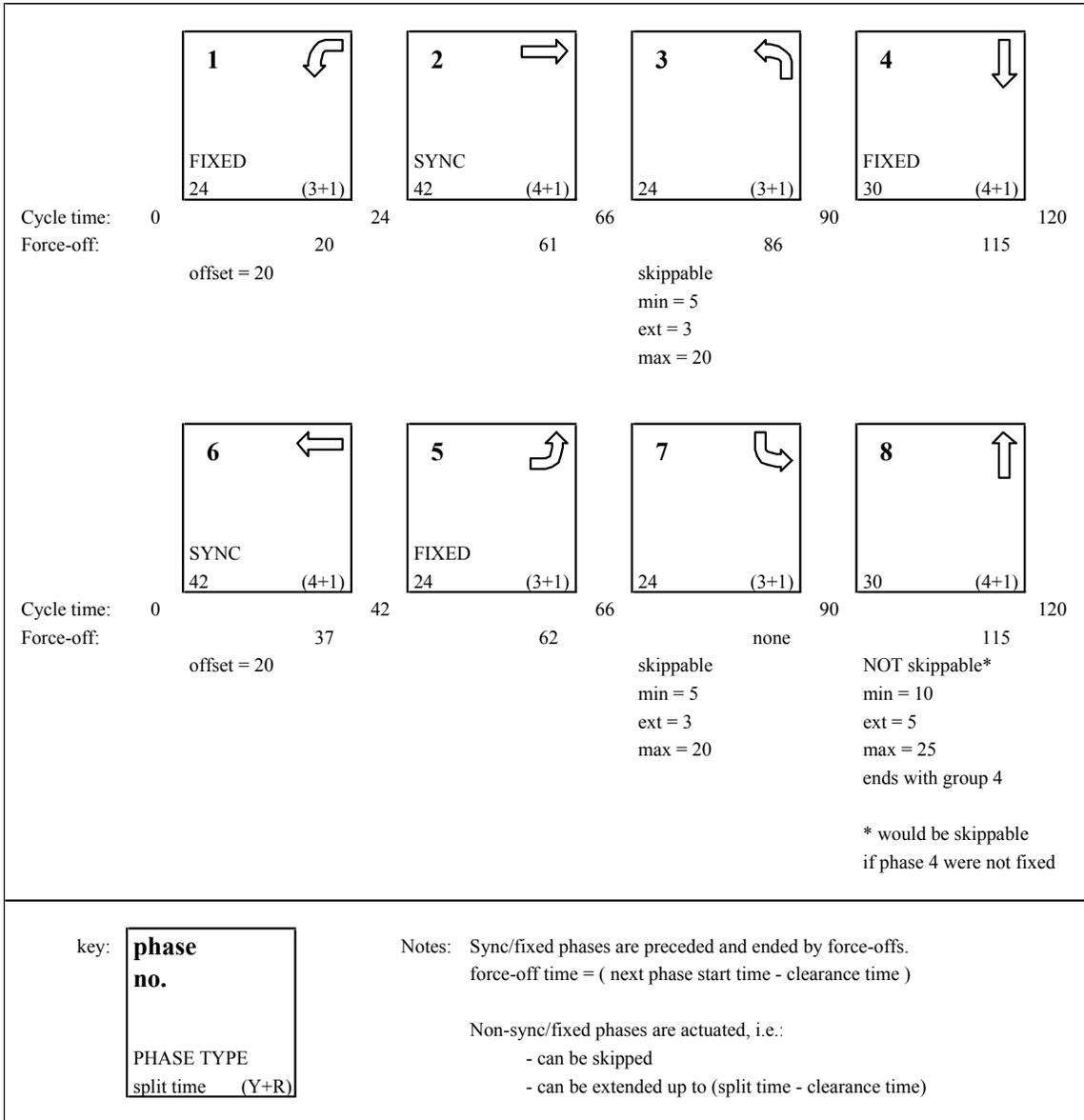


Figure B-1: Phase and timing diagram for eight-phase controller.

```

00:00:00      # Signal plan start time
{
  385          # ControllerID
  20 1 3      # ControllerType SignalType NumEgresses
  {
    4 # NumSignals
    {
      150 { 7 4 4 } # SignalID { GroupID for each movement }
      151 { 5 2 2 }
      152 { 3 8 8 }
      153 { 1 6 6 }
    }

    8 # NumSignalGroups
    {
      1 3 5 # GroupID InitialPeriod NumConditions
      {
        # ConditionCode [NumParameters] { Parameters }
        002 { 20. } # delayed start (offset)
        102 { 3 2 20. 120. 20. } # G: force-off
        208 { 3 } # G: hold
        201 { 2 3. } # Y: min
        207 3 { 1 1. 4 } # R: conflict clearance
      }

      2 1 4
      {
        102 { 3 2 61. 120. 20. } # G: force-off
        208 { 3 } # G: hold
        201 { 2 4. } # Y: min
        207 3 { 1 1. 1 } # R: conflict clearance
      }

      3 1 7
      {
        004 1 { 260 } # gap timer
        101 3 { 3 1 260 } # G: no demand
        102 { 3 2 86. 120. 20. } # G: force-off
        201 { 3 5. } # G: min
        202 { 3 3. 20. } # G: extension
        201 { 2 3. } # Y: min
        207 3 { 1 1. 2 } # R: conflict clearance
      }

      4 1 4
      {
        102 { 3 2 115. 120. 20. } # G: force-off
        208 { 3 } # G: hold
        201 { 2 4. } # Y: min
        207 3 { 1 1. 3 } # R: conflict clearance
      }

      6 3 5
      {
        002 { 20. } # delayed start (offset)
        102 { 3 2 37. 120. 20. } # G: force-off
        208 { 3 } # G: hold
        201 { 2 4. } # Y: min
        207 3 { 1 1. 8 } # R: conflict clearance
      }

      5 1 4 # GroupID InitialPeriod NumConditions
      {
        102 { 3 2 62. 120. 20. } # G: force-off
        208 { 3 } # G: hold
        201 { 2 3. } # Y: min
      }
    }
  }
}

```

```

    207 3 { 1 1. 6 }          # R: conflict clearance
  }

7 1 6
{
  004 1 { 257 }              # gap timer
  101 3 { 3 1 257 }          # G: no demand
  201 { 3 5. }               # G: min
  202 { 3 3. 20. }           # G: extension
  201 { 2 3. }               # Y: min
  207 3 { 1 1. 5 }          # R: conflict clearance
}

8 1 6
{
  102 { 3 2 115. 120. 20. } # G: force-off
  201 { 3 10. }              # G: min
  202 { 3 5. 25. }           # G: extension
  205 2 { 3 4 }              # G: complementary group
  201 { 2 4. }               # Y: min
  207 3 { 1 1. 7 }          # R: conflict clearance
}

} # End of all Signal Groups

} # End of Controller
} # End of Time Period

```

Appendix C

Example Signal Input Files: Signal Group Specification

This section presents a signal timing plan from the Hornstull case study network (see Chapter 5 for description) and the corresponding MITSIMLab input file. Section C.1 presents the base timing plan, and Section C.2 presents the plan with PRIBUSS functions implemented.

C.1 Pretimed-Coordinated Control

The signal group diagram containing the basic timing data is shown in Figure C-1. The corresponding MITSIMLab signal input file follows.

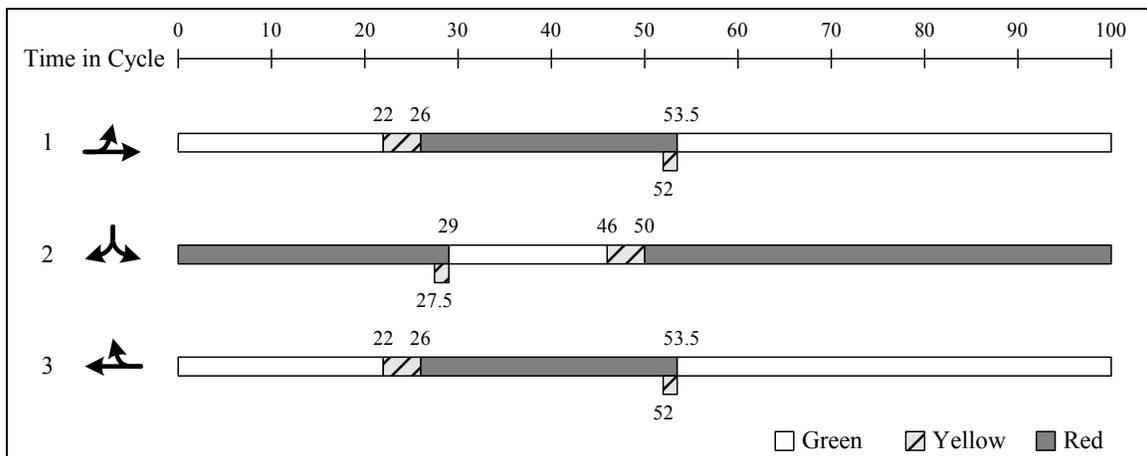


Figure C-1: Signal group diagram (Hornsgatan-Varvsgatan intersection).

```

06:30:00      # Signal plan start time
{
9             # ControllerID
20 1 2       # ControllerType SignalType NumEgresses
{
3   # NumSignals
{
21 { 2 2 }           # SignalID { GroupID for each movement }
22 { 1 1 }
23 { 3 3 }
}
}

3   # NumSignalGroups
{

1 3 6       # GroupID InitialPeriod NumConditions
{
# ConditionCode [NumParameters] { Parameters }
102 { 3 2 22. 100. 0. } # G: force-off
208 { 3 } # G: hold
201 { 2 4. } # Y: min
001 { 1 18 } # R: next period
207 3 { 1 2. 2 } # R: conflict clearance
201 { 18 1.5 } # R/Y: min
}

2 1 6       # GroupID InitialPeriod NumConditions
{
# ConditionCode [NumParameters] { Parameters }
102 { 3 2 46. 100. 0. } # G: force-off
208 { 3 } # G: hold
201 { 2 4. } # Y: min
001 { 1 18 } # R: next period
207 3 { 1 1.5 1 } # R: conflict clearance
201 { 18 1.5 } # R/Y: min
}

3 3 6       # GroupID InitialPeriod NumConditions
{
# ConditionCode [NumParameters] { Parameters }
102 { 3 2 22. 100. 0. } # G: force-off
208 { 3 } # G: hold
201 { 2 4. } # Y: min
001 { 1 18 } # R: next period
207 3 { 1 2. 2 } # R: conflict clearance
201 { 18 1.5 } # R/Y: min
}

} # End of signal groups

} # End of controller

} # End of Time Period

```

C.2 Pretimed-Coordinated Control with Bus Priority

In this example, priority-equipped buses can call for green extension or phase shortening. The parameters for the priority logic are shown in Table C.1 (see Section 4.1.2 for parameter descriptions). All values are times relative to time 0 in the cycle. The MITSIMLab signal input file incorporating the PRIBUSS functions follows.

Table C.1: PRIBUSS parameters (Hornsgatan-Varvsgatan intersection).

	Group	Call Window		Time-out (D)
		Start (A)	End (B)	
Green Extension	1	99	22	35
	3	1	24	35
Phase Shortening	1	33	46	53.5
	3	33	46	53.5

```

06:30:00      # Signal plan start time
{
9            # ControllerID
20 1 2      # ControllerType SignalType NumEgresses
{
3          # NumSignals
{
21 { 2 2 }      # SignalID { GroupID for each movement }
22 { 1 1 }
23 { 3 3 }
}
}

3          # NumSignalGroups
{

1 3 14      # GroupID InitialPeriod NumConditions
{
# ConditionCode [NumParameters] { Parameters }
006 { 109 }      # check-in
007 { 110 }      # check-out
014 { 53.5 100. 0. }      # reset check-in counter
008 { 99. 22. 35. 100. 0. }      # bus extension flag
012 { 33. 46. 53.5 100. 0. }      # bus shortening flag
013 6 { 33. 46. 53.5 100. 0. 2 }      # bus shortening flag preceding
210 { 3 40.5 22. 100. 0. }      # G: hold in window
303 { 1 1 64 0 }      # skip if this group not extended
208 { 3 }      # G: hold indefinitely
205 2 { 3 3 }      # G: complementary group
201 { 2 4. }      # Y: min
001 { 1 18 }      # R: next period
207 3 { 1 2. 2 }      # R: conflict clearance
201 { 18 1.5 }      # R/Y: min
}

2 1 8      # GroupID InitialPeriod NumConditions
{
# ConditionCode [NumParameters] { Parameters }
102 { 3 2 46. 100. 0. }      # G: force-off
303 { 1 2 2048 1 }      # skip if this group shortened
}

```

```

208 { 3 } # G: hold
201 { 3 4. } # G: min
201 { 2 4. } # Y: min
001 { 1 18 } # R: next period
207 3 { 1 1.5 1 } # R: conflict clearance
201 { 18 1.5 } # R/Y: min
}

3 3 14 # GroupID InitialPeriod NumConditions
{
# ConditionCode [NumParameters] { Parameters }
006 { 125 } # check-in
007 { 126 } # check-out
014 { 53.5 100. 0. } # reset check-in counter
008 { 1. 24. 35. 100. 0. } # bus extension flag
012 { 33. 46. 53.5 100. 0. } # bus shortening flag
013 6 { 33. 46. 53.5 100. 0. 2 } # bus shortening flag - preceding
210 { 3 40.5 22. 100. 0. } # G: hold in window
303 { 1 3 64 0 } # skip if this group not extended
208 { 3 } # G: hold indefinitely
205 2 { 3 1 } # G: complementary group
201 { 2 4. } # Y: min
001 { 1 18 } # R: next period
207 3 { 1 2. 2 } # R: conflict clearance
201 { 18 1.5 } # R/Y: min
}

} # End of signal groups

} # End of controller

} # End of Time Period

```

Bibliography

- Balke, K.N., C.L. Dudek, and T. Urbanik II (2000). Development and Evaluation of an Intelligent Bus Priority Concept. Presented at the 79th Annual Meeting of the Transportation Research Board.
- Ben-Akiva, M.E., A. Davol, T. Toledo, H.N. Koutsopoulos, W. Burghout, I. Andréasson, T. Johansson, and C. Lundin (2002). Calibration and Evaluation of MITSIMLab in Stockholm. Forthcoming, Transportation Research Board.
- Bretherton, D. (1996). Current Developments in SCOOT: Version 3. *Transportation Research Record 1554*, pp. 48-52.
- Chang, G., M. Vasudevan, and C. Su (1995). Bus-Preemption under Adaptive Signal Control Environments. *Transportation Research Record 1494*, pp. 146-154.
- Dale, J.J., R.J. Atherley, T. Bauer, and L. Madsen (1999). A Transit Signal Priority Impact Assessment Methodology—Greater Reliance on Simulation. Presented at the 78th Annual Meeting of the Transportation Research Board.
- Duerr, P.A. (2000). Dynamic Right-of-Way for Transit Vehicles: An Integrated Modeling Approach for Optimizing Signal Control on Mixed Traffic Arterials. Presented at the 79th Annual Meeting of the Transportation Research Board.
- EB Traffic (1990). Swedish Proposal for Traffic Engineering Vocabulary for Traffic Control.
- Federal Highway Administration (2000). *Manual on Uniform Traffic Control Devices*. U.S. Department of Transportation, <http://mutcd.fhwa.dot.gov/>.
- Federal Highway Administration (1996). *Traffic Control Systems Handbook*. Report No. FHWA-SA-95-032, U.S. Department of Transportation, Washington, DC.
- Federal Transit Administration (2000). *Advanced Public Transportation Systems: The State of the Art, Update 2000*. Report No. FTA-MA-26-7007-00-1, U.S. Department of Transportation, Washington, DC.
- Furth, P.G. and T.H.J. Muller (1999). TRAFCOD: A Method for Stream-Based Control of Actuated Traffic Signals. Presented at the 78th Annual Meeting of the Transportation Research Board.
- Furth, P.G. and T.H.J. Muller (2000). Conditional Bus Priority at Signalized Intersections: Better Service Quality with Less Traffic Disruption. *Transportation Research Record 1731*, pp. 23-30.

- Garrow, M. and R. Machemehl (1998). Development and Evaluation of Transit Signal Priority Strategies. Presented at the 77th Annual Meeting of the Transportation Research Board.
- Gartner, N.H., P.J. Tarnoff, and C.M. Andrews (1991). Evaluation of Optimized Policies for Adaptive Control Strategy. *Transportation Research Record 1324*, pp. 105-114.
- Gartner, N.H., C. Stamatiadis, and P.J. Tarnoff (1995). Development of Advanced Traffic Signal Control Strategies for Intelligent Transportation Systems: A Multi-Level Design. *Transportation Research Record 1494*, pp. 98-105.
- Homburger, W.S. and J.H. Kell (1988). *Fundamentals of Traffic Engineering, 12th Edition*. Institute of Transportation Studies, University of California at Berkeley, Berkeley, CA.
- Hu, K., S. Skehan, and R. Gephart (2001). Implementing a Smart Transit Priority System for Metro Rapid Bus in Los Angeles. Presented at the 80th Annual Meeting of the Transportation Research Board.
- Jourdain, S. (1992). *Urban Intersection Control*. The Book Guild Ltd., Sussex, England.
- Kell, J.H. and I.J. Fullerton (1982). *Manual of Traffic Signal Design*. Prentice-Hall, Englewood Cliffs, NJ.
- McShane, W.R., R.P. Roess, and E.S. Prassas (1990). *Traffic Engineering*. Prentice-Hall, Englewood Cliffs, NJ.
- Peek Traffic (1999). User's Description for PRIPTS in EC-1 EuroControllerCode, Rev. 2.1. Peek Traffic AB, Stockholm, Sweden.
- Peek Traffic (2000). SPOT/UTOPIA User Manual. Peek Traffic AB, Stockholm, Sweden.
- Skabardonis, A. (2000). Control Strategies for Transit Priority. Presented at the 79th Annual Meeting of the Transportation Research Board.
- Stockholms Gatukontor (1991). PRIBUSS: Projekterings- och Programmeringsanvisningar. Stockholm Street Administration, Stockholm, Sweden.
- Vägverket (1987). LHOVRA: A New Traffic Signal Control Strategy for Isolated Junctions. English Translation of Report TU 155. Swedish National Road Administration, Sweden.
- Vasudevan, M. and G. Chang (2001). Design Framework for Integrating Real-Time Bus Priority Control with Robust Arterial Signal Progression. Presented at the 80th Annual Meeting of the Transportation Research Board.
- Yang, Q., H.N. Koutsopoulos, and M. Ben-Akiva (2000). A Simulation Laboratory for Evaluating Dynamic Traffic Management Systems. *Transportation Research Record 1710*, pp. 122-130.
- Yang, Q. (1997). *A Simulation Laboratory for Evaluation of Dynamic Traffic Management Systems*. Ph.D. Thesis, Department of Civil and Environmental Engineering, Massachusetts Institute of Technology.