DETERMINATION OF TRAFFIC RESPONSIVE PLAN SELECTION FACTORS AND THRESHOLDS USING ARTIFICIAL NEURAL NETWORKS

A Thesis

by

ANUJ SHARMA

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

August 2004

Major Subject: Civil Engineering

DETERMINATION OF TRAFFIC RESPONSIVE PLAN SELECTION FACTORS AND THRESHOLDS USING ARTIFICIAL NEURAL NETWORKS

A Thesis

by

ANUJ SHARMA

Submitted to Texas A&M University
in partial fulfillment of the requirements
for the degree of

MASTER OF SCIENCE

Approved as to style and content by:

Carroll J. Messer
(Chair of Committee)

Mark Burris
(Member)

Cliff Spiegelman
(Member)

Bruce McCormick
(Member)

Paul Roschke
(Interim Head of Department)

August 2004
Major Subject: Civil Engineering

ABSTRACT

Determination of Traffic Responsive Plan Selection Factors and Thresholds Using Artificial Neural Networks. (August 2004)

Anuj Sharma, B.E., Regional Engineering College, Rourkela, India

Chair of Advisory Committee: Dr. Carroll Messer

Traffic congestion has become a menace to civilized society. It degrades air quality, jeopardizes safety and causes delay. Traffic congestion can be alleviated by providing an effective traffic control signal system. Closed-loop traffic control systems are an example of such a system.

Closed-loop traffic control systems can be operated primarily in either of two modes: Time of Day Mode (TOD) or Traffic Responsive Plan Selection Mode (TRPS). TRPS mode, if properly configured, can easily handle time independent variation in traffic volumes. It can also reduce the effect of timing plan aging. Despite these advantages, TRPS mode is not used as frequently as TOD mode. The reason being a lack of methodologies and formal guidelines for predicting the factors and thresholds associated with TRPS mode. In this research, a new methodology is developed for determining the thresholds and factors associated with the TRPS mode. This methodology, when tested on a closed-loop system in Odem, Texas, produced a classification accuracy of 94%. The classification accuracy can be increased to 98% with a proposed TRPS architecture.

ACKNOWLEDGEMENTS

I would like to thank my advisor, Dr. Carroll Messer, for his guidance and support during my graduate studies at Texas A&M University. I will always remember his willingness to see to my welfare as well as those of his other students.

I would also like to thank Dr. Cliff Spiegelman, Dr. Mark Burris, and Dr. Bruce McCormick for guiding me during the course of my thesis.

I am also grateful to Dr. Montasir Abbas who provided me with an insight in this research and also allowed me to use some field data collected for TxDOT project 4421. I would like to thank the Texas Transportation Institute for providing me the resources to undertake this study.

I am also thankful to my family and friends who were with me whenever I needed their support.

Above all, I would like to thank the good Lord for his guidance and protection.

TABLE OF CONTENTS

Page

LIST OF FIGURES

Page

LIST OF TABLES

CHAPTER I

INTRODUCTION

Traffic signals are provided to separate conflicting movements in time for a given space. A signal timing plan is developed for efficient and safe operation of the signal for a given traffic demand. Impromptu implementation of the best available timing plan with change in traffic demand has always been challenging. This becomes a major concern when traffic demand varies widely and is highly unpredictable.

Traffic controllers usually use the time of day (TOD) mode to blindly choose a timing plan from a set of pre-stored plans. The timing plan schedule, which is developed based on historical data, chooses a timing plan for the given time. This scheme works on the assumption that traffic demand on a system is recursive in nature and repeats itself with a fixed cycle length. However, this assumption is invalid in cases such as special events, holiday traffic, shifting of peak times due to slight changes in office timings, or just due to a recursive demand with variable cycle length. In order to implement the most efficient timing plan for a given demand, traffic responsive plan selection (TRPS) mode has been developed. TRPS mode uses system detectors to estimate demand on the network and chooses the best available timing plan for the existing condition.

This thesis follows the style and format of the *Transportation Research Board*.

PROBLEM STATEMENT

The TRPS mode is intended to operate as a traffic pattern classifier. It is provided with a mechanism to measure the volume count and occupancy from a set of system detectors, essentially a feature vector of demand, and classifies the present state as one of the available predetermined demand classes.

The main problem with implementing the TRPS mode is the plethora of set-up factors and thresholds that need to be determined for the efficient working of the mode. Like any other control system, desired benefits of TRPS mode can only be achieved if the TRPS factors and parameters are determined correctly. Traffic responsive plan selection logic is based on prior work done in the early 1970's on a federal urban traffic control system (UTCS) project in Washington, D.C. (*1, 2*). UTCS classified system detectors in three TRPS parameters (inbound, outbound and cross street detectors). System detector's volume and occupancy (V+O) were normalized and aggregated using weighting factors to compute the separate V+O values for TRPS parameters. UTCS compared these computed V+O values against respective thresholds to select a traffic responsive pattern. This logic has been implemented by signal controller manufacturers during the last 30 years with only slight variation in name and number of the TRPS parameters. A more detailed description of the TRPS factors and the threshold is provided in the following paragraphs.

System Detectors

The TRPS control system uses system detectors to sense existing demand on the network. System volume and occupancy data are then processed to select a suitable timing plan for the existing demand. System detectors should be located to provide a reliable estimate of the demand. Federal Highway Administration (FHWA) provides guidelines in locating system detectors (*2*). Several (*4, 5*) other researches have been done in this area. The main points to be kept in mind while locating the system detectors are:

- They should be able to capture the influence of major traffic generators
- They should be located outside the influence of adjacent intersections. Queue lengths and acceleration zones should be taken into consideration during the design
- Redundancy in information provided by the detectors should be reduced for optimal utilization of the available detectors

Among current traffic controllers, the system detectors have been categorized in three groups. Each of these categories serves a different purpose in the TRPS mechanism as follows:

- Cycle level detectors are used to determine the appropriate cycle length and, therefore, should be located near the critical intersection(s)

- Arterial detectors or directionality detectors are used to determine the appropriate offset level and, therefore, should be placed in the inbound and outbound directions on the arterial
- Non-arterial detectors are used to determine the appropriate splits level and, therefore, should be placed on the cross streets

Smoothing Factors

Smoothing factors, as the name suggests, are used to iron out short term fluctuations in demand. Different controller manufacturers use slightly different approaches for this purpose. These approaches essentially utilize two mathematical functions. The first approach is called moving average. In this method, a smoothed value is calculated by using the formulae listed below:

$$\textit{Smoothed value} = \textit{New Value}*(1\textit{-Smoothing Factor}) + \textit{Old Value}*\textit{Smoothing Factor}$$

$$0 \leq SmoothingFactor < 1 \tag{1.1}$$

Selecting a larger value of smoothing factor gives a higher weight to old data; whereas, selecting a value of ‘0’ implies that the new value is used without smoothing.

The other smoothing approach is to average the value over the previous '*n*' minutes sampling interval. Higher values of '*n*' would result in less sensitivity towards the changes occurring in a single interval.

Scaling Factors

Scaling factors are used to standardize the data in the range of 0 to 100. Standardized data are independent of approach capacity. The importance of standardizing data before implementing pattern classification algorithm is explained in greater details in Chapter III. Two sets of scaling factors are used, one each for count and occupancy. The highest values of count and occupancy observed on the site are recommended as scaling factors *(6)*.

Weighting Factors

A weighting factor is assigned to every system detector. Weighting factors are multiplied to count and occupancy to obtain computational channel parameters. Some controllers provide different weighting factors for count and occupancy. These factors form the basis of pattern recognition. This concept has been further explained in Chapter III.

TRPS Mechanism and Thresholds

TRPS utilizes a set of Pattern Selection (*PS*) parameters to select a timing plan based on the threshold values set for each of these parameters. Figure 1.1 shows the mechanism of TRPS mode. System detector α_1, γ_p etc. collect counts and occupancy and report these to the master controller. This information is scaled, smoothed, weighed and aggregated to obtain values for Computational Channel (*CC*) parameters. The name of *CC* parameters differs from one manufacturer to other. Most controller manufacturers have the same nomenclature for *PS* parameters, namely:

- Cycle
- Split
- Offset

PS parameters are a function of *CC* parameters. The calculated values of *PS* parameters are compared against the threshold values to select a timing plan number from a lookup table. Timing plan corresponding to this plan number is then activated in all local controllers supervised by the master controller.

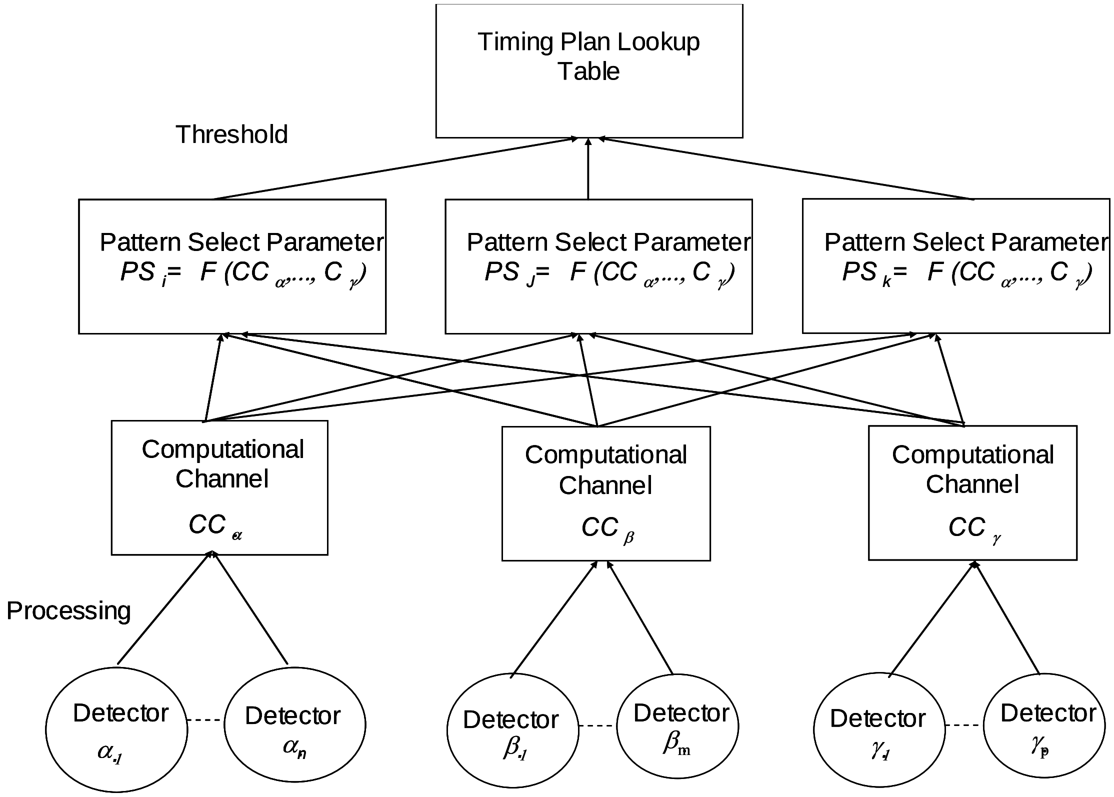


FIGURE 1.1 General TRPS mechanism

The TRPS mode varies slightly from one controller vendor to other. Following section describe TRPS mode as implemented by Naztec *(7)*.

Naztec TRPS

As described in Naztec operational manual for closed-loop master controllers *(6)*, Naztec uses only three flow values as *CC* parameters, namely:

- Inbound
- Outbound
- Cross traffic

Functions of *CC* parameters are then used to calculate the Pattern Select (PS) parameters (cycle, split and offset). Figure 1.2 shows the Naztec TRPS mechanism. One of the 144 patterns is selected from the lookup table using combinations of the cycle-offset-split PS parameter and corresponding timing plan is implemented.

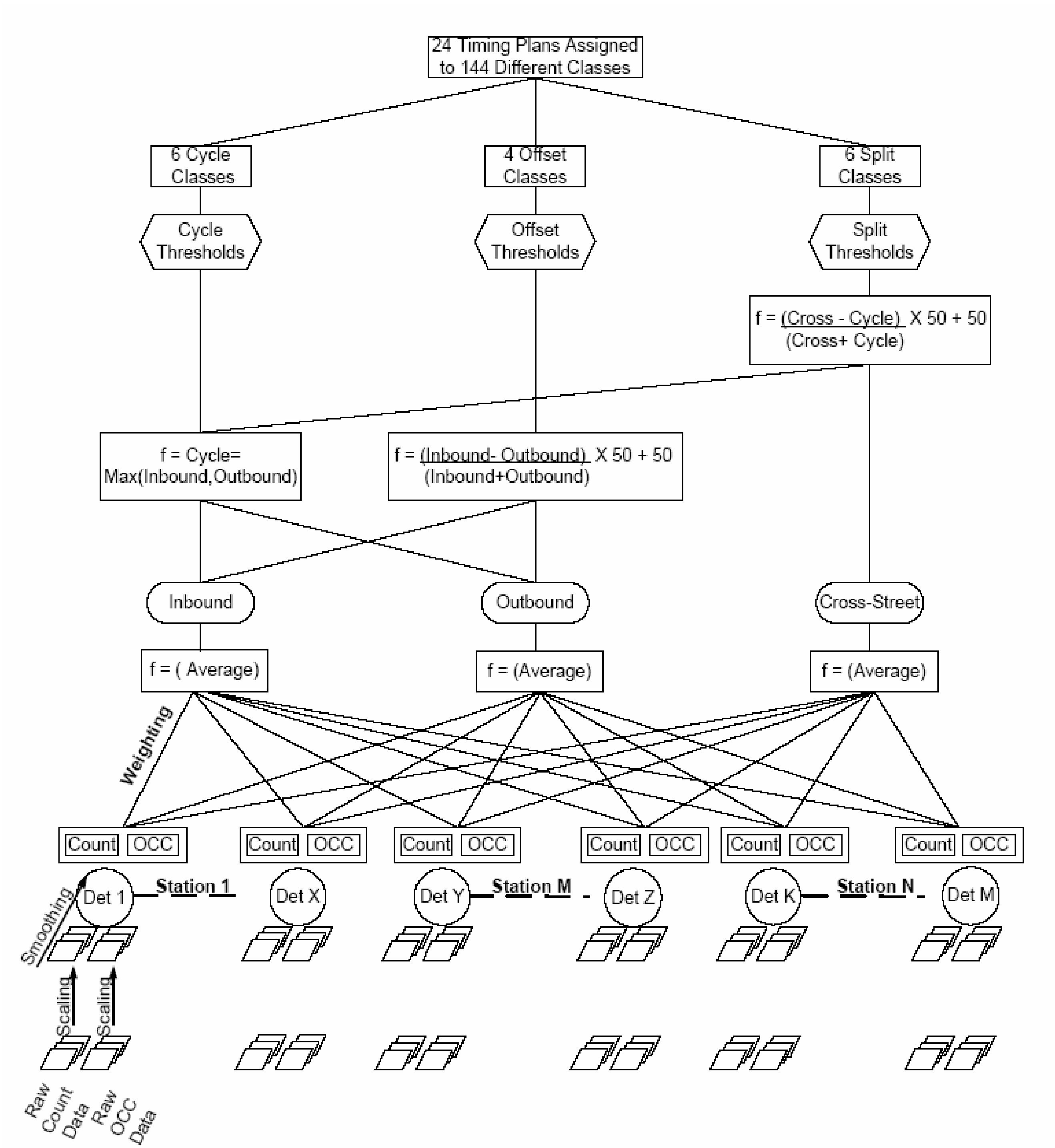


FIGURE 1.2 Naztec TRPS mechanism *(8)*

RESEARCH OBJECTIVE

The primary difficulty lies in the correct determination of the factors for an efficient operation of TRPS mode. The objective of this research is to develop a procedure for determining the TRPS factors and thresholds for providing more efficient operation. TRPS is ideally a pattern recognition problem where the detector inputs are used to assign the detected traffic state to a particular pre-stored timing plan. Abbas and Sharma *(9)* used canonical discriminant analysis *(10)*, a Bayesian-based approach, for addressing this pattern recognition problem.

In this research TRPS factors and thresholds will be determined using an artificial neural network (ANN), which is a non-parametric approach. Neural network, according to Haykin *(11)*, has an edge over deterministic approaches for pattern recognition, as it does not assume a distribution of the input. Therefore, they can be beneficial in cases where the inputs are generated by nonlinear mechanisms and when the distribution is heavily skewed and non-Gaussian, as is expected in this research.

RESEARCH SCOPE

This research is focused on closed-loop arterial signal control systems. Other networks could also be addressed in a similar manner, but they will not be addressed in this research. This research uses data collected in Odem, Texas during TxDOT research project 4421 *(8)*. Simulation data are used for determining the factors and thresholds for TRPS mode. This research uses the Naztec controller as an example to determine TRPS factors and thresholds.

CHAPTER II

BACKGROUND

The Urban Traffic Control System program (UTCS) *(1)* was started in 1967 by the U.S. Department of Transportation, Federal Highway Administration for developing, implementing and evaluating urban traffic control strategies. The strategies for traffic control were divided into three categories as listed in Table 2.1.

TABLE 2.1 Feature of UTCS/BPS Strategies (2)

Feature	**First Generation**	**Second Generation**	**Third Generation**
Optimization	Off-Line	On-Line	On-Line
Frequency of Update	15 Minutes	10-15 Minutes	3-6 Minutes
No. of Timing Patterns	Up to 40	Unlimited	Unlimited
Traffic Prediction	No	Yes	Yes
Critical Intersection Control	Adjusts Splits	Adjusts Splits	Adjusts Splits, Offset, and Cycle
Hierarchies of Control	Pattern Selection	Pattern Computation	Congested Medium Flow
Fixed Cycle Length	Within Each Section	Within Variable Groups of Intersections	No Fixed Cycle Length

In first generation control, a set of signal timing plans were calculated based on historical data. The control timing plan were selected based on the time of day mode (TOD), or traffic responsive (TRPS) mode, which used a set of system detectors to choose the optimal timing plan for the existing traffic volumes. TRPS was also tested with added features like critical intersection control (CIC) and bus priority signal (BPS). Second generation control was midway between first and third generation in terms of complexity and real-time computation. Signal timing plans were calculated online based on surveillance data and predicted volume conditions. The timing plans were updated every 10-15 minutes to avoid transition disturbances. The third generation control also calculated and implemented the timing plan in real time. The main differences between the second and third generation controls were that the frequency of update in third generation control was 5 to 10 minutes, and variable cycle lengths could occur within an optimization period over the controlled intersections.

The two important results reported by UTCS after evaluating all the controls were:

- Computer-based traffic responsive alternative generally matched or exceeded the performance of the other traffic control alternatives *(2)*
- The First Generation (TRSP) was found to be operationally effective, was the least expensive to apply, and should be given primary consideration for implementation *(1)*

After positive feedback given on the UTCS project, first generation controllers were widely implemented. There are researches in progress for the development of second and third generation control *(12)*. But the majority of traffic controllers present in the field are first generation controllers. First generation controllers are microcomputer-based traffic controllers. A master controller supervises a set of local controllers. This control concept is termed as being a "Closed-loop System." Closed-loop systems are described in next section.

CLOSED-LOOP SYSTEMS

Closed-loop systems consist of a series of signalized intersections operated by a single master controller. The master controller issues commands to implement timing plans stored in the local controllers. The master controller can also report information to a Traffic Management Center, if needed. Figure 2.1 shows a sketch of the closed-loop system. The main purpose of providing a closed-loop network is to coordinate the connected signalized intersections. The coordination of signals can result in a significant reduction in vehicular delay *(13)*. Coordinated signals can also reduce the number of vehicle stops, total fuel consumption, and vehicle emissions *(13)*. Wagner *(14)* found that "the most dramatic improvements in traffic performance on signalized arterials and networks are those resulting from the combined action of interconnecting previously uncoordinated pre-timed signals with a master controller, together with the introduction of new optimized timing plans." Similar results were also shown by the Traffic Light Synchronization II (TLS II) program, which aimed at optimization of traffic signal timing plans across Texas. TLS II reported a 13.5 % (20.8 million gallons/year) reduction in annual fuel consumption, 29.6% (22 million hours/year) reduction in annual delay, and 11.5% (729 million stops / year) reduction in annual stops *(15)* based on results obtained from 1348 intersections in 43 cities.

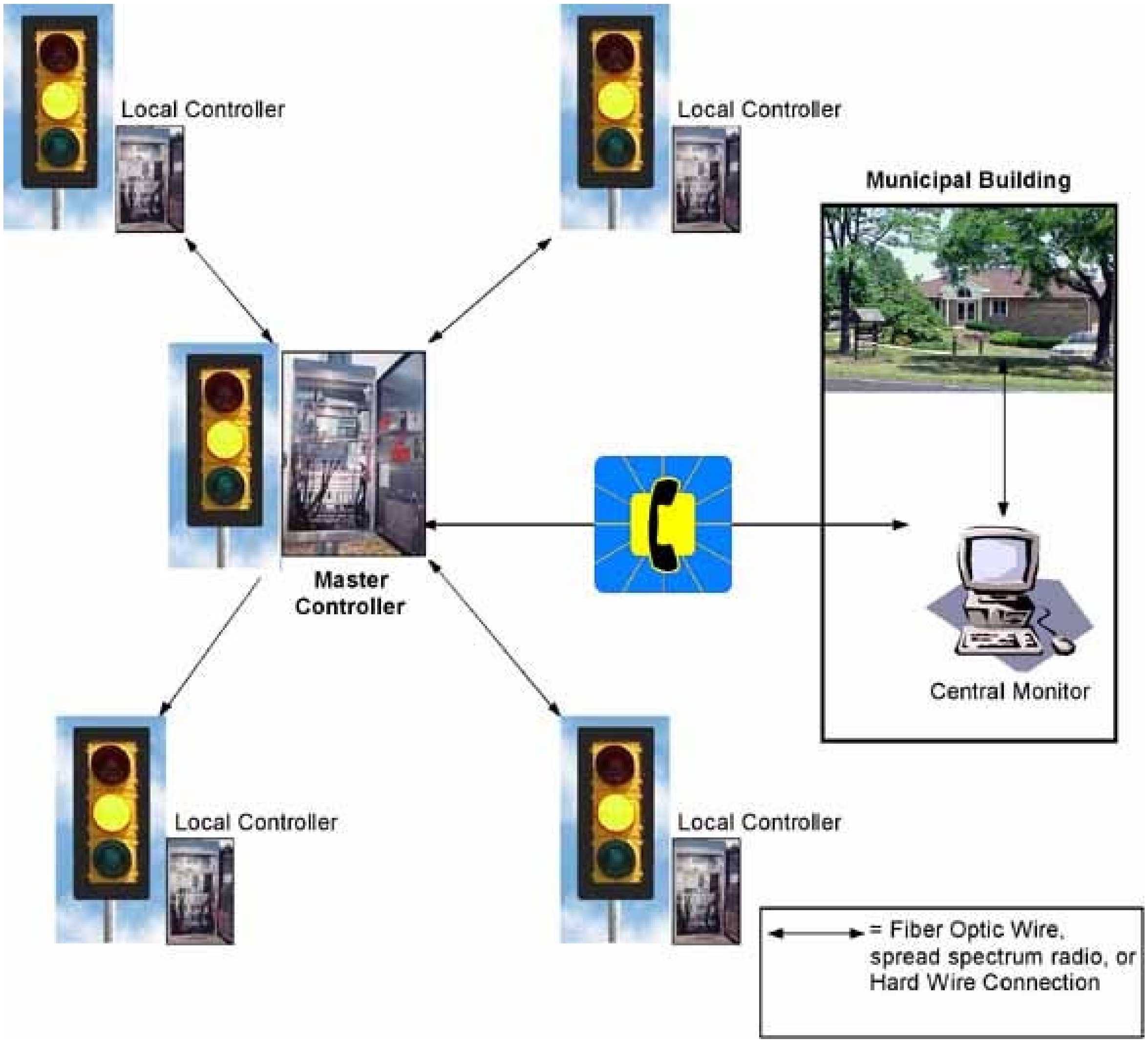


FIGURE 2.1 Components of a closed-loop system *(16)*

CLOSED-LOOP SYSTEM MODES OF OPERATION

The best way to increase the performance of a group of intersections is to operate them under the best suited timing plan for existing demand and coordinate them. This thesis aims at providing a methodology to implement the best suited timing plan. The timing plan required at any given time depends on the existing volume conditions (demand). The decision to select a particular timing plan at a given instant is taken by the controller. A closed-loop system essentially uses two control modes for the selection of the timing plan, namely:

- Time of day mode
- Traffic responsive plan selection mode

In the TOD mode, the timing plan is selected and implemented based on the time of the day. In this mode, it is assumed that the traffic patterns are recursive in time by day of week. This assumption may be violated in cases of special events, holiday traffic, or in the cases where recurrence of similar conditions occur, but in a random manner. These cases can be handled efficiently using the TRPS mode, which provides a mechanism to select a timing plan as a real-time response to measured changes in traffic demands. So TRPS mode, if correctly set, can result in efficient operation of closed-loop networks. In a study of two networks in Lafayette, Indiana, it was found that TRPS mode reduced total system delay by 14% compared to TOD mode for the midday traffic pattern *(17)*. It was also found that TRPS system reduced the total system delay for morning traffic by 38%. The following section presents the mathematical evaluation of the two modes.

Mathematical Evaluation

Timing plans are designed to serve traffic demand on a network. This relationship between the timing plans and demand of the network is mathematically shown below.

If D is the demand on a particular network, then timing plan (TP) is a function of demand:

$$TP = f(D) \tag{2.1}$$

In the case of time of day and traffic responsive mode, which uses a set of pre-calculated timing plans, the function $f(D)$ is defined as a step function

$$TP = \begin{cases} TP1 & if \quad 0 \le D \le d_1 \\ TP2 & if \quad d_1 \le D \le d_2 \\ . & . \\ . & . \\ TPn & if \quad d_{n-1} \le D \le d_n \end{cases} \tag{2.2}$$

where:

$d_i = i^{th}$ level of demand on system.

Time of Day Mode

In this mode a new variable time (T) is introduced. The relationship between time and demand is modeled using historical data. And this model is used to predict the future demand. So a timing plan is selected based on the time. This can be shown mathematically as follows:

$$D = g\ (T) \tag{2.3}$$

where function g is

$$D = \begin{cases} D1 & if \quad 0 \le T \le t_1 \\ D_2 & if \quad t_1 \le T \le t_2 \\ . & \quad . \\ . & \quad . \\ D_n & if \quad t_{n-1} \le T \le t_n \end{cases} \tag{2.4}$$

It is assumed that after time "t_n" the demand will repeat itself so the timing plans can be scheduled as shown below:

$$TP = h\ (t) \tag{2.5}$$

such that,

$$TP = \begin{cases} TP_1 & if \quad 0 \le T \le t_1 \\ TP_2 & if \quad t_1 \le T \le t_2 \\ . & \quad . \\ . & \quad . \\ TP_n & if \quad t_{n-1} \le T \le t_n \end{cases} \tag{2.6}$$

Figure 2.2 shows a hypothetical volume distribution plotted versus time. If TOD mode is implemented three timing plans can be assigned:

- An a.m. peak timing plan from 7:00 A.M. to 11:00 A.M.
- An off peak timing plan from 11:00 A.M. to 5:00 P.M.
- A p.m. peak timing plan from 5:00 P.M. to 7:00 P.M.

These timing plans would be repeated on each of the workdays as the traffic scenario can be assumed to be same through out the weekday.

Figure 2.3 shows TOD mode routine graphically. This model works fine in case of recursive demand with a fixed cycle of recursion. This is not true in cases of special events, holiday traffic or in the cases where recurrence of similar demand pattern occurs, but in a random manner.

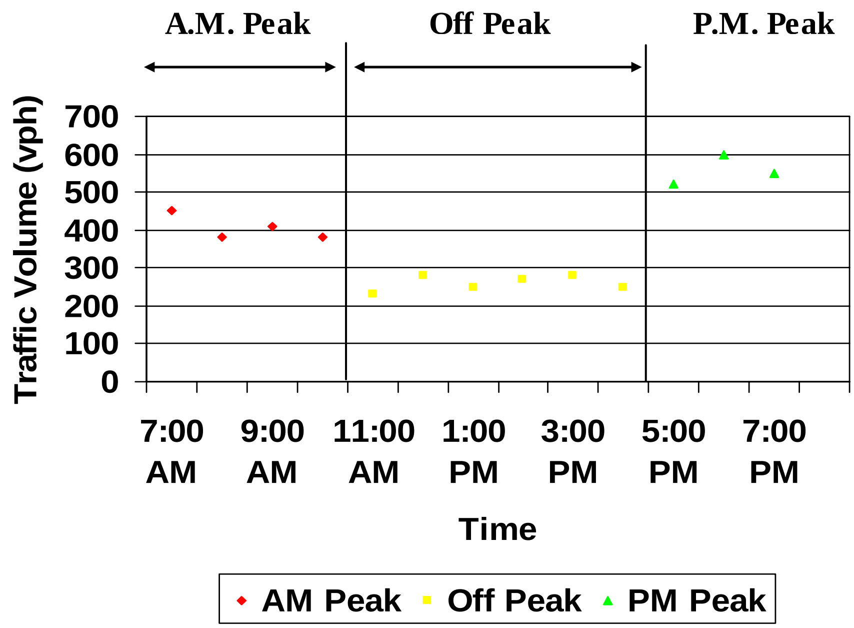


FIGURE 2.2 Example of TOD mode

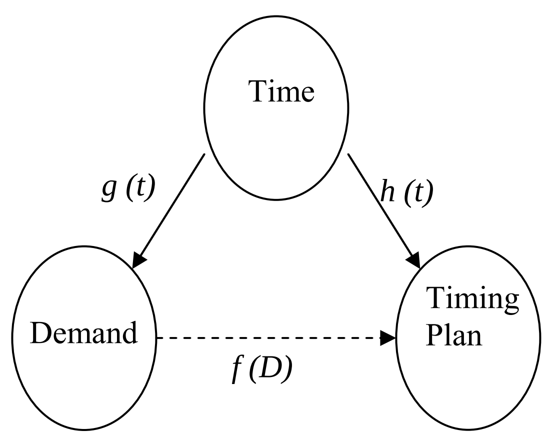


FIGURE 2.3 TOD mode routine

Traffic Responsive Mode

With the advent of new detection capabilities it became possible to measure the demand in real time and a new routine was developed to tap the detection resources. In this routine the demand is directly measured and the optimal timing plan is selected using the function f as defined in Equation 2.2

Figure 2.4 presents the TRPS mode implementation on the example shown in Figure 2.2. In case of TRPS mode three timing plans can be assigned:

- TP 1 for volume ranging from 0 vph to 300 vph
- TP 2 for volume ranging from 300 vph to 500 vph
- TP 3 for volume greater than 500 vph

These timing plans would be implemented based on the existing volume conditions.

Figure 2.5 shows TRPS routine graphically. Notice that in this case no model is needed as a bridge between the demand and timing plan, thus reducing the possibilities of error.

Based on the discussion above, it is clear that TRPS mode does not require any model relationship between time and demand; instead, it directly measures the demand in the field so the error induced due to modeling the time-demand relationship is eliminated. Hence, if properly configured, TRPS mode provides a more efficient operation.

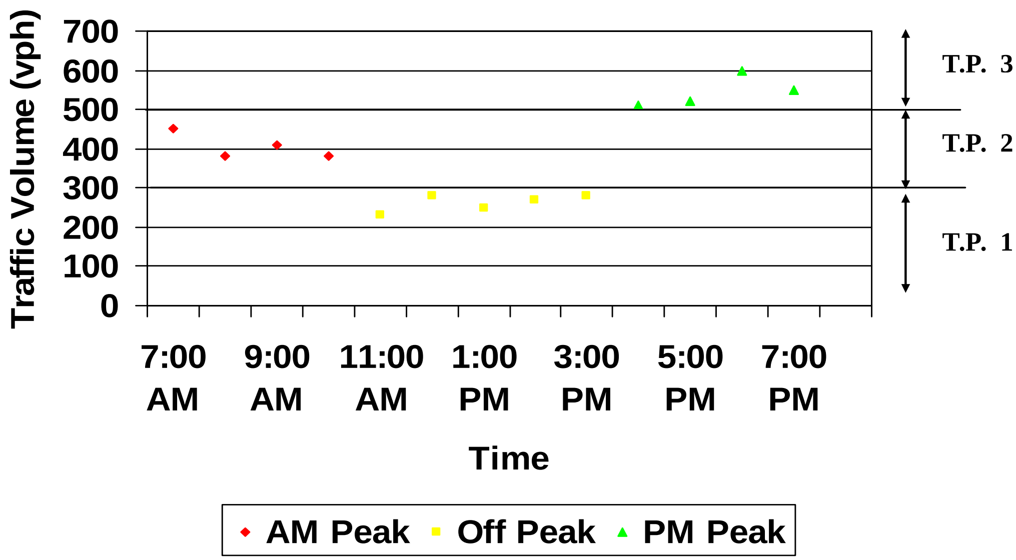


FIGURE 2.4 Example of TRPS mode

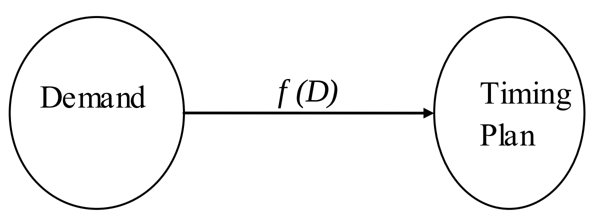


FIGURE 2.5 TRPS mode routine

PAST RESEARCH ON TRPS

Despite the above described benefits, the TRPS mode has been rarely implemented in the field. The main reason why the TOD mode is preferred as compared to TRPS mode is the lack of literature and methodology for configuration of the TRPS mode *(8)*. The insufficiency of literature and lack of experience often leads traffic engineers to fall back to the TOD mode although being aware of the fact that the TRPS mode might be efficient in operation. There has been limited research to determine the methodology for setting up the TRPS mode. The literature sources that have been found are discussed below.

Hadi and Courage *(18)* used five cycle lengths equally varying from the lowest cycle length (sum of minimum green) to the maximum cycle length (either as accepted by a local traffic agency or as determined using design volumes as maximum volumes on each link of the system). The cycle level threshold was experimentally determined by running different arterial volume level in TRANSYT-7F *(19)* and cutoff point were established, when a change in volume level causes change in the optimum cycle length as reported by TRANSYT-7F. Detectorized approaches measured the traffic flow conditions. Using upper cutoff limit as a design cycle, 5 offsets plans were obtained for 1%, 25%, 50%, 75% and 99% of total progression bandwidth. The thresholds were obtained marking the cutoff limit for change from one offset percentage to the other by varying the inbound-outbound condition for a given cycle length. Similarly, 3 sets of splits and their thresholds were obtained by using TRANSYT-7F for light, average and heavy cross street volumes.

In other research Abbas and Sharma *(9)* used canonical discriminant analysis for finding the TRPS factors and threshold. In this approach a set of timing plans were obtained using SYNCHRO 5.0 *(20)* for generating a set of timing plans for a design set of volume conditions varying at a controlled rate. The timing plans were then clustered. The field data were assigned to one of these clusters using root mean square distance as a measure of classification. CORSIM *(21)* was then used to simulate the chosen classes. The detector counts and occupancy were used as inputs in canonical discriminant analysis to obtain the system detector weights. The thresholds were determined by using discriminant analysis on the canonical variable obtained from previous analysis.

One of the major drawbacks in both of these approaches is that they use timing plans or the measures of effectiveness (MOEs) as given by signal design software for classification of demand. All the signal design softwares use different models and criteria for developing an optimal timing plan and predicting MOEs. If these timing plans or MOEs are used as classification criteria for demand, an error might be induced due to the modeling of the software. The methodology used in this thesis directly uses the volume counts for the classification of demand and keeps the timing plan assignment to these classes as a separate module.

The other important drawback of Haidi's research, but improved upon by Abbas and Sharma, was the use of three axes of cycle, split and offset to differentiate among the demand classes. These might not be the optimum set of axes for separation and classification of demand classes. Abbas and Sharma, on the other hand, used canonical distribution for determining the axis, which is best suited for demand classification. This provides a major breakthrough in way of thinking of cycle (associated with the inbound and outbound direction), split (associated with the cross-street and main street movement) and offset (associated with the ratio of inbound and outbound movement) pattern selection parameter provided by the controller vendors. As discussed by Abbas and Sharma, the inbound, outbound and cross street movements are usually associated with a pattern selection parameter and might not be the best suited axes for classification. On the other hand, a linear combination of inbound, outbound and cross street movements as given by canonical distribution (which is a methodology to find the axis providing the maximum separation between classes) will provide the best classification. This thesis also includes this concept in its methodology.

Following chapter provides a detailed description of the methodology and concept used in determination of the TRPS factors and thresholds in this thesis.

CHAPTER III

METHODOLOGY

This chapter provides a detailed description of the methodology used for the determination of the TRPS factors and thresholds. Figure 3.1 presents the outline of research methodology. TRPS mode classifies an existing demand in one of the predetermined demand states (classes) and selects an appropriate signal timing plan. Demand states were determined by clustering the approach volumes of the network collected in the field. Clustering techniques (*22*, *23*, *24*) identify a set of “natural” groups existing in the data set. After identification of these groups (referred as demand states), timing plans were assigned to each of these states. The next step was to find the TRPS weights and thresholds in order to assign future volume scenarios to one of these states. This pattern recognition was done by the use of an artificial neural network. Each of these stages is described in detail in the following sections.

CLUSTERING

Identification of existing demand states was the first step for this research. The accuracy of TRPS mode and benefits achieved was dependent on efficient clustering of existing demands. The demand needs to be clustered due to the limitation of number of timing plans that could be put in traffic controllers, for example in a Naztec version

50/60 controller software 48 timing plans could be implemented *(6)*. A single signal timing plan was associated with a range of demand patterns. Clustering techniques identify the demand patterns having similar attributes and group them together. The use of clustering technique is illustrated by an example below.

Assume a hypothetical network having a northbound (two lane) approach and an eastbound (one lane) approach. The demand on this network can be represented by a two dimensional vector of approach volumes. Four hypothetical demands namely *A, B, C* and *D* are plotted in Figure 3.2. On inspection of the figure, *A* and *B* can be classified in one group, whereas *C* and *D* can be classified in the other. This subjective way of clustering is

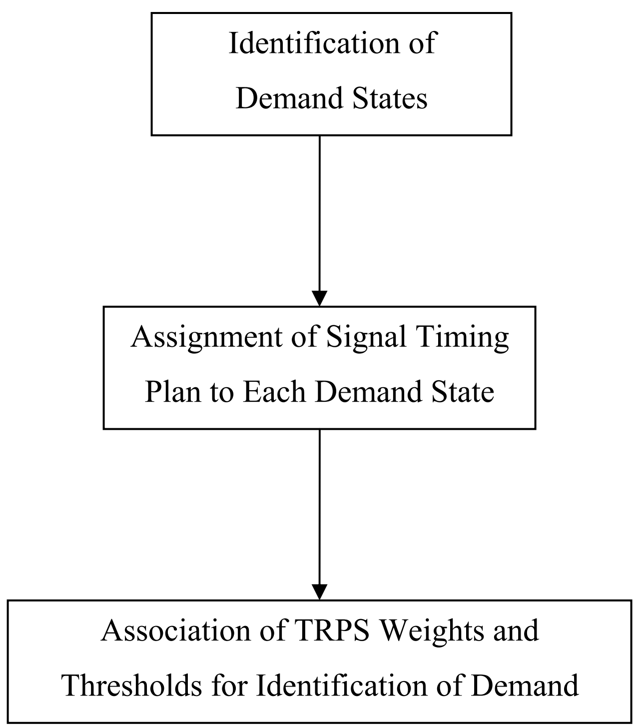


FIGURE 3.1 Flow chart of research methodology

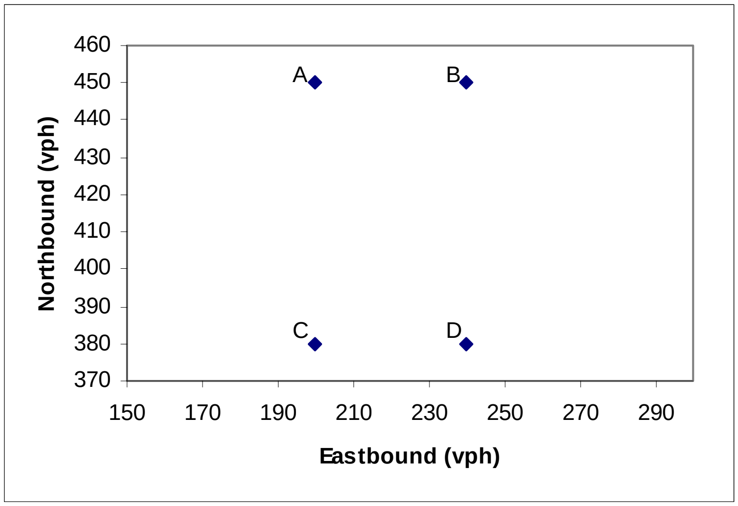


FIGURE 3.2 Hypothetical example of various demand states

mostly used in our profession. The demand is classified as A.M. peak, P.M. peak or, off-peak, etc., based on perception and judgment of the engineer. In the above example, the human perception system is used for grouping the data. Human perception techniques are effective and accurate for classification of up to three dimensions. However, in real world scenarios, where networks may have many approaches, and in turn several dimensions, automated algorithms need to be used for accuracy. Clustering algorithms essentially have three steps:

- Determination of feature vectors before collecting data
- Standardization of feature vectors
- Clustering of feature vectors with similar attributes

Data Collection and Standardization

The main aim of clustering the demand data was to associate a signal timing plan to each group. Similar volume conditions on all approach would perform efficiently under a single timing plan. Volume counts on each approach of the network, for every 15 minutes, were chosen as feature vector. Field data were collected to cover all the demand variation existing in the field. For a particular network, data should be collected during a normal day, a weekend, any special event, and during light traffic conditions.

Data so collected need to be standardized to avoid the dependence on the scale of measurement. This dependence can be explained by revisiting the first example. Based on Figure 3.2 *A* and *B* were clustered together in one group and *C* and *D* in the other. If vehicle per hour per lane was used as a unit of measurement of flow instead of vehicle per hour, and plotted, as shown in Figure 3.3, *A* and *C* will be grouped in one cluster and *B* and *D* in the other.

To avoid this dependence of clustering on the choice of measurement units and scaling, the flow needs to be standardized to corresponding flow ratio. Flow ratio is defined as ratio of volume on an approach and the saturation flow of that approach. Saturation flow is defined as the maximum number of vehicles from a lane group that would pass through the intersection in one hour under the prevailing traffic and roadway conditions if the lane group was given a continuous green signal for that hour. So, flow ratio represents the fraction of approach capacity being utilized. Usually the saturation flow volume of a single lane is in the range of 1200 to 1900 vehicles per hour of green. The highway capacity manual (*25*) provides a well-defined methodology for calculating the saturation flow of any approach. So after collecting the volume data for each approach, it was divided by its approach saturation flow to obtain the flow ratio. Flow ratio is always positive and is between 0 and 1. This conversion helped to standardize the collected data for clustering analysis.

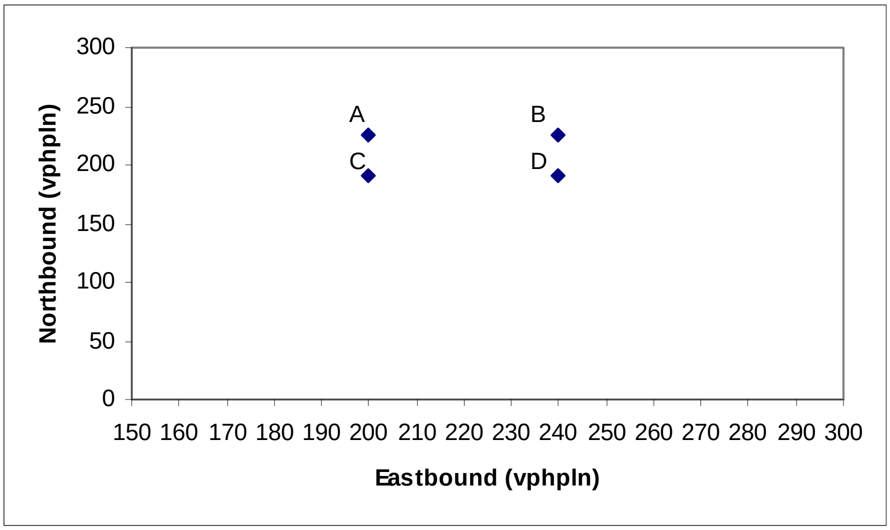


FIGURE 3.3 Plot of northbound volume (vphpln) and eastbound volume (vphpln)

Type of Clustering Algorithms

There are mainly two types of clustering algorithms: partitioning methods and hierarchical methods. Partitioning algorithms, also known as "flat clustering" algorithms, produce a set of disjoint clusters as shown in Figure 3.4. This figure shows

20 observations being clustered in three disjoint groups or partitions. A partition method constructs *k* clusters such that:

- Each group must contain at least one object
- The clusters must be mutually disjointed such that each observation falls in exactly one group

Hierarchical clustering algorithm, on the other hand, gives hierarchy of nested clusters. A tree structure, as shown in Figure 3.5, is created to show the grouping in the data. This type of clustering techniques is usually used in biological applications, like taxonomy of plants and animals. Hierarchical clustering is usually performed on small data sets.

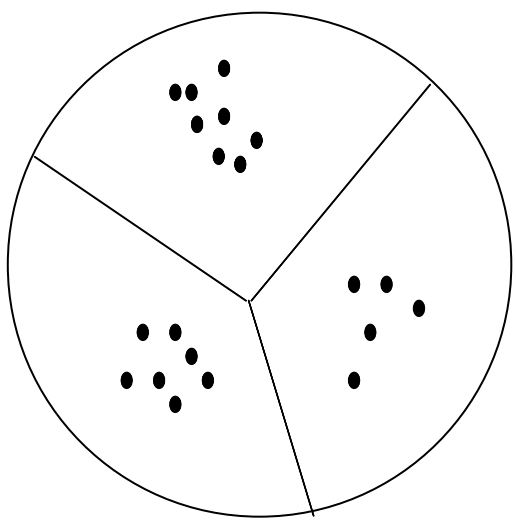


FIGURE 3.4 Sketch of partitioning clustering algorithm having 3 clusters and 20 observations *(22)*

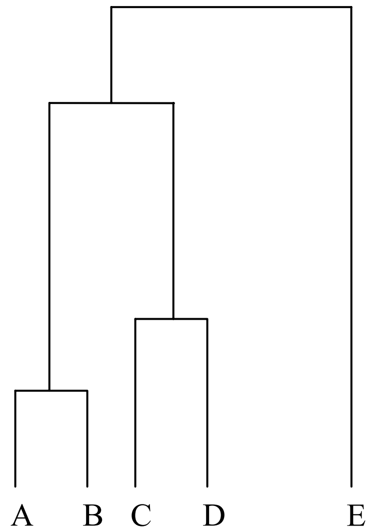


FIGURE 3.5 Sketch of hierarchical cluster having 5 observations *(22)*

The partitioning algorithm is used in this research for finding disjoint groups of demand states. K-Means clustering (22, 23, 24), a partitioning method, was used in this research. MATLAB (26) toolbox was used for K-Means clustering. The following paragraph gives a brief overview of this method.

K-Means Clustering

The K-Means clustering algorithm is a widely used method for partition clustering. The algorithm aims to minimize the overall "within cluster distance" from the patterns to the centroids. Since it is not possible to perform an exhaustive search on all the possible distributions of clusters, an iterative approach is followed. Cluster centroids, m_j, are iteratively adjusted by assigning each pattern to the closest centroid to find the local minima of the objective function. The objective function to be minimized is shown in Equation 3.1.

$$E = \sum_{j=1}^{k} \sum_{x_i \in w_j} \| x_i - m_j \|^2 \tag{3.1}$$

$$m_i = \frac{1}{N_i} \sum_{x_i \in w_j} x \tag{3.2}$$

where:

E = error (sum of squared deviations) of cluster partitions

k = number of clusters

x_i = i^{th} pattern feature vector of an element of group j

m_j = centroid feature vector of group j

w_j = j^{th} *group*

The main steps of the algorithm are as follows:

1. Define the number of clusters.
2. Initialize cluster's centroids.
3. Assign a data point to the cluster having closest centroid.
4. Calculate the new cluster centroid.
5. Repeat steps 3 and 4 unless all the data points are assigned to a cluster.

6. Calculate the error (sum of squared deviation) for given classification.
7. Reassign the data points to minimize the sum of square deviation.

The first two steps of the algorithm pose certain difficulty as we don't know the number of clusters and their centroids beforehand. Silhouette width was used for determination of number of clusters in the data. The initial value of the centroids was randomly chosen from data points.

Silhouettes were introduced by Rosseeuw (*27*) for graphical representation of each cluster. A silhouette plot shows those cluster points that lie within the cluster and those which hold only an intermediate position. The plots are used to compare the compactness and separation among the clusters. The procedure of constructing silhouettes is described below.

Each object *i* is associated with a value *s(i)* and these values are then plotted. In order to define *s(i)*, an object *i* is taken from the data set. If *i* belongs to cluster *A*, *a(i)* is defined as:

a(*i*) = average dissimilarity of *i* to all other objects of *A*.

and *d(i, C)* is defined as:

d(i, C) = average dissimilarity of *i* to all other objects of *C* where is *C* is different cluster from *A*.

After calculating *d(i,C)* for all cluster $C \neq A$, the smallest value is selected and represented by *b*(*i*).

$$b(i) = \min_{C \neq A} d(i, C) \tag{3.3}$$

The cluster *B*, having the value *b(i)*, is called the neighbor of object *i*. The value of *s(i)* is now obtained by the equation given below:

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}} \tag{3.4}$$

So, from Equation 3.4 it is evident that

$$-1 \leq s(i) \leq 1 \tag{3.5}$$

The value *s(i)* approaches 1 as the value of *b(i)*, the smallest “between” dissimilarity, is much larger than the *a(i)*, the “within” dissimilarity. A value closer to 1 thus implies that i^{th} point is more similar to the data points in its group (*A*) rather than any other group (*B*). Therefore, *i* can be said to be well classified.

In a different situation, when *s(i)* is close to 0 when *a(i)* and *b(i)* are nearly equal, which means that i^{th} point is equally close to two groups A and B. So, it is unclear whether *i* should be assigned to *A* or *B*. This represents an intermediate case of classification.

The worse situation is when *s(i)* is close to -1 where *a(i)* is much larger than *b(i)*. This implies that i^{th} point is more similar data points in B, a group other than the group to which it is assigned (*A*). In this case it might be concluded that *i* has been misclassified in *A*.

Silhouette for cluster *A* is a plot of values of *s(i)*, ranked in decreasing order for all objects in *A*. Figure 3.6 is a example of silhouette plot. A wide, box shaped, silhouette having all positive values represents a well pronounced cluster.

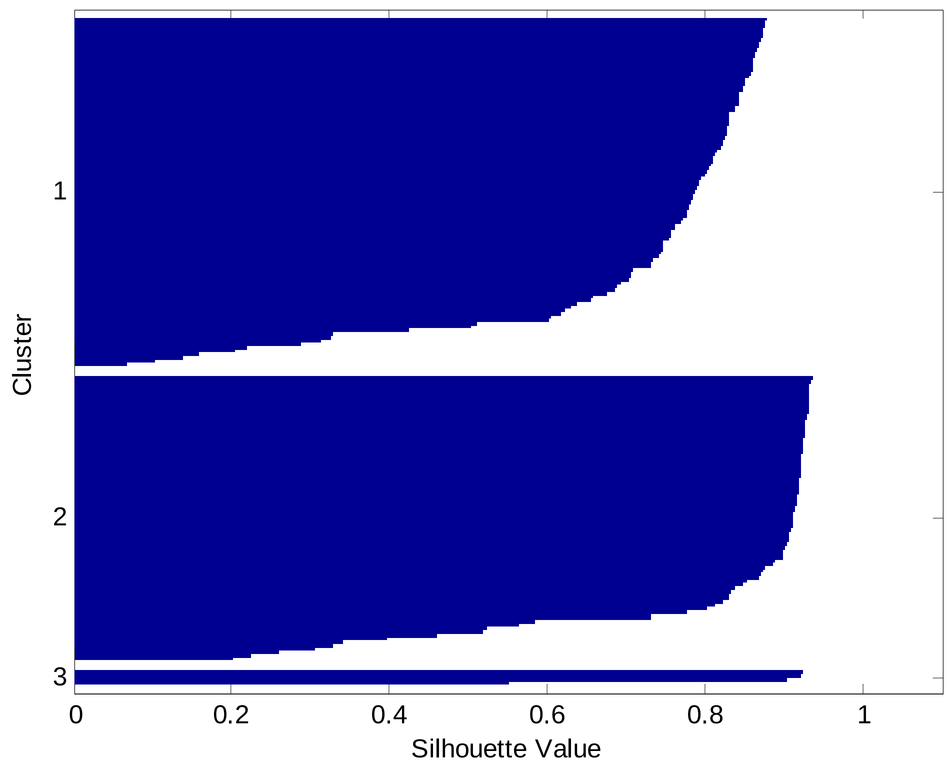


FIGURE 3.6 Silhouette plot for three clusters

The average value of the *s(i)* for all the data points is calculated. This is denoted by $\bar{s}(k)$ and is called silhouette width for the entire data set. Silhouette width is used for selecting the best value for *k* (number of clusters). *K* is selected to give the largest value of $\bar{s}(k)$. The maximum value of $\bar{s}(k)$ for all values of *k* from 0 to *n-1* (*n* is the number of data points) is termed as Silhouette coefficient (*SC*). Table 3.1 enlists the range of *SC* and proposed interpretation as given by Kaufman and Rousseeuw (*22*).

TABLE 3.1 Subjective Interpretation of Silhouette Coefficient (*SC*) *(22)*

Silhouette Coefficient, *SC*	**Proposed Interpretation**
0.71 – 1.00	A strong structure has been found
0.51 – 0.70	A reasonable structure has been found
0.26 – 0.50	The structure is weak and could be artificial; try additional methods on this data set
≤ 0.25	No substantial structure has been found

In this research, K-means clustering was used, as described above, for finding the demand state. After identifying these states we moved on to step 2 to assign a signal timing plan for each of these demand states.

A set of examples showing the K-means algorithm and plotting of silhouettes have been provided in Appendix A.

TIMING PLAN ASSIGNMENT

The signal timing plan is designed to optimally serve a demand state. There are many software programs, like SYNCHRO, PASSER II *(28)*, TRANSYT-7F that calculate an optimum signal timing plan for given demand on a network. We still lack software which can produce an optimum signal timing plan for a group of demands. Due to time constraints and scope of this research, such software was not developed. Instead timing plans were generated using SYNCHRO 5.0 for 85 percentile approach volumes for each state. Development of software which stochastically determines an optimal timing plan optimal for a set of demand patterns is highly recommended for future research.

The 85 percentile approach volumes were chosen as the design demand based on engineering judgment. The reasoning for this choice can be explained as follows. Figure 3.7 plots traffic delay versus cycle length for increasing demand scenarios. It is evident there exists an optimal cycle length for given demand that results in minimum traffic delay. If a lower cycle length than the optimum cycle length is used, delay increases at a high rate; if a higher cycle length is used, delay increases at a slower rate. It can also be noticed that for a given demand cluster a higher demand requires a bigger cycle length. It can be inferred that cycle length corresponding the maximum demand in a demand group will result in least delay for entire group. So it would beneficial to choose a higher demand on each approach. Since it is rarely possible that all the approaches will simultaneously reach their maximum, the 85 percentile value for each approach volume in a given state (cluster) is recommended to be used as design demand for a given state. This also restricts from cycle lengths from becoming very high due to certain outliers in a given state. Very high values of cycle length might cause excessive delays for cross street traffic.

Using the design volumes, a timing plan was developed and assigned to each traffic state. The next step was to simulate all the demand scenarios with each of the timing plans.

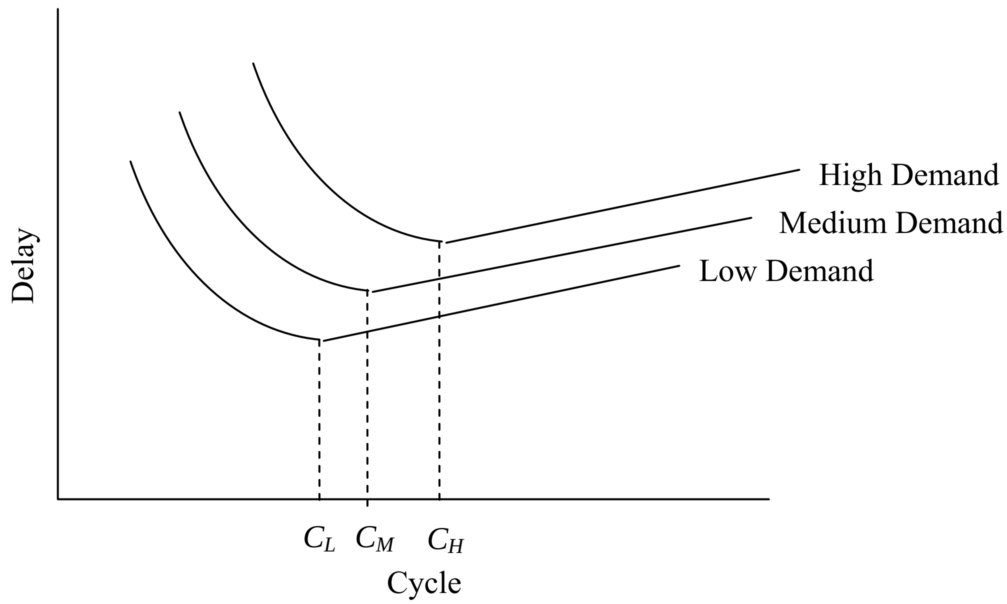


FIGURE 3.7 Plot of delay versus cycle length

PATTERN RECOGNITION

After a timing plan was associated with each state, the next step was to identify these states using system detectors. The system detector outputs for each of the states were obtained by simulating the traffic network. It is important to note that a timing plan affects the system detector outputs for a given demand. To take this into account, all the demand conditions need to be simulated with each of the timing plans. FHWA simulation program CORSIM 5.1 was used for simulation. For an arterial network, the system detectors need to be placed on all input approaches to represent the demand on the network. After collecting the system detector count and occupancy for all possible demand timing plan combinations, the next step was to use this data as input and validation data for neural network and determine the TRPS factors and threshold.

TRPS mode in traffic control is based on the principle of linear discriminant analysis *(24)* (feature extraction) for classification of data. A set of pattern select (*PS*) parameters (features) are extracted as a linear combination of system detector outputs. These *PS* parameters are then used for the classification of demand. A simple example is used to describe this concept. Let *w1* and *w2* be two hypothetical demand states represented by a two dimensional vector. Figure 3.8 shows the graphical plot of the states. The elliptical circle represents the state boundaries. *PS* parameter essentially provides the axis of maximum separability between the demand states. A single feature (*PS* parameter) in this case can provide 100% classification. It is not necessary to use all the *PS* parameters for classification. The number of *PS* parameters to be used is determined by the number of demand states and their orientations.

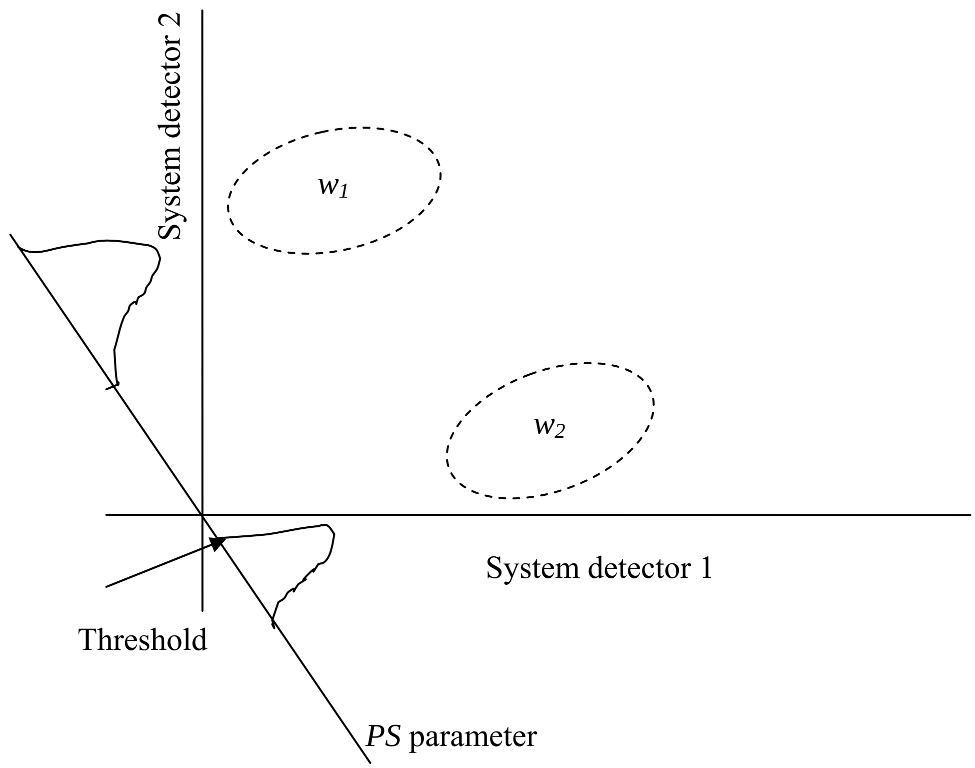


FIGURE 3.8 TRPS concept of linear discriminant analysis

ANN for Determination of TRPS Weights

An artificial Neural Network (ANN) (11, 29, 30, 31) was used for determining the orientation or weights of *PS* parameters. Following paragraphs present a brief overview of ANN.

An ANN tries to emulate the organizational structure of the brain. They constitute a set of interconnected units called neurons. The interconnections are used for sending signals from one unit to the other in either an amplified or an inhibited way. The inhibition or amplification is provided by the use of connection weights. Figure 3.9 diagrammatically shows a simplistic neural network as shown in MATLAB. A neuron receives an input vector $P = [p_1, p_2 ... p_R]'$ which passes through a set of syntactic weights $W = [w_{1,1}, w_{1,2} .. w_{1,R}]'$ and then is aggregated with a bias *(b)* to give value *n*. Activation function *f* then acts on value *n* to give the output *a*. This forms a basic building block for numerous complex ANN architectures such as multi-layer perceptron, support vector machine, etc.

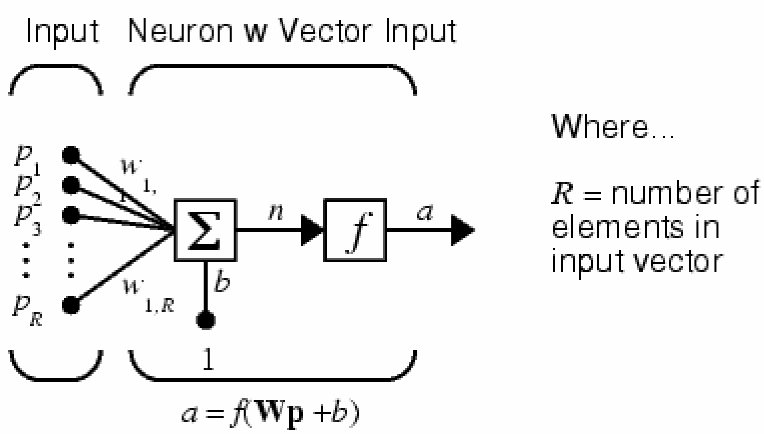


FIGURE 3.9 Simplistic diagram of ANN *(26)*

The multi-layer perceptron was used in this research for finding TRPS weights. Figure 3.10 shows the architecture of the multi-layer perceptron used. The network constituted of an input layer, two hidden layers and an output layer. The number of nodes in the input layer were equal to $2n$, where n was the number of system detectors used in the network. Weights $w_{1,1}^{1}$ to $w_{1,n}^{1}$ were TRPS weights. Linear activation function was used in the first hidden layer. The 2nd hidden layer was provided for the classification purposes of PS values. A sigmoid activation function was used in this layer. The last layer was the output layer. Numbers of neurons in the output layers were equal to the number of demand states to be classified. System detector count and occupancy data obtained in the previous step were scaled before feeding them as input. Counts and occupancies were scaled to the range of 0 to 1. The processed data were divided into two halves (randomly selected): training data and validation data. The training and validation data were composed of processed count and occupancy data together with a target value for the network. The target value was the vector representation of the demand state. For example, demand State 1 in a three state system was represented by a three dimensional vector ($[1,0,0]'$). All the weights were randomly initialized and then the network was trained using the training data. The back propagation algorithm *(11)* was used for modifying network weights to converge the system outputs to the desired target values. MATLAB neural network toolbox was used for training the network.

Weights connecting the first hidden layer to the input layer were used as system detector weights. This process could be repeated if there was a large overlap between demand states to determine weights for other PS parameters.

One important drawback in recent traffic controllers was that negative weights cannot be used. This was due to the misconception that weights implied importance of any detector and therefore could not be negative. The weights essentially are related to the slope of a hyper plane on which demand classes, when projected, attain maximum seperability. Even in the simple example shown in Figure 3.7, *PS* parameter has a negative slope and, hence, needs a negative set of weights. This problem needs to be amended by the controller vendors for achieving good seperabilty.

Another important point to note is that the neural network architecture used for this research is based on the present controller architecture of TRPS mode. This was not the most efficient method of pattern recognition; higher accuracy was attainable using multiple layer perceptron with single hidden layer having sigmoidal activation function. This statement was validated by comparing the performance of both the architecture in the data analysis chapter. The proposed modified architecture is shown in Figure 3.11. The proposed architecture was similar to present architecture until the input layer. After the input layer, the proposed network resembled MLP having a single hidden layer with sigmoidal function. Layer II might have up to 20 to 25 nodes. A portion of these, as required by a particular network, would be used for classification and other could be discarded by setting connection weights to zero. Layer III consisted of a set of nodes representing number of timing plans available in the controller. Note that proposed network would have two sets of connection weights, instead of one set of weights as in present traffic controllers. The output layer selected timing plans corresponding to the node of Layer III having highest value. This optimal timing plan would be implemented.

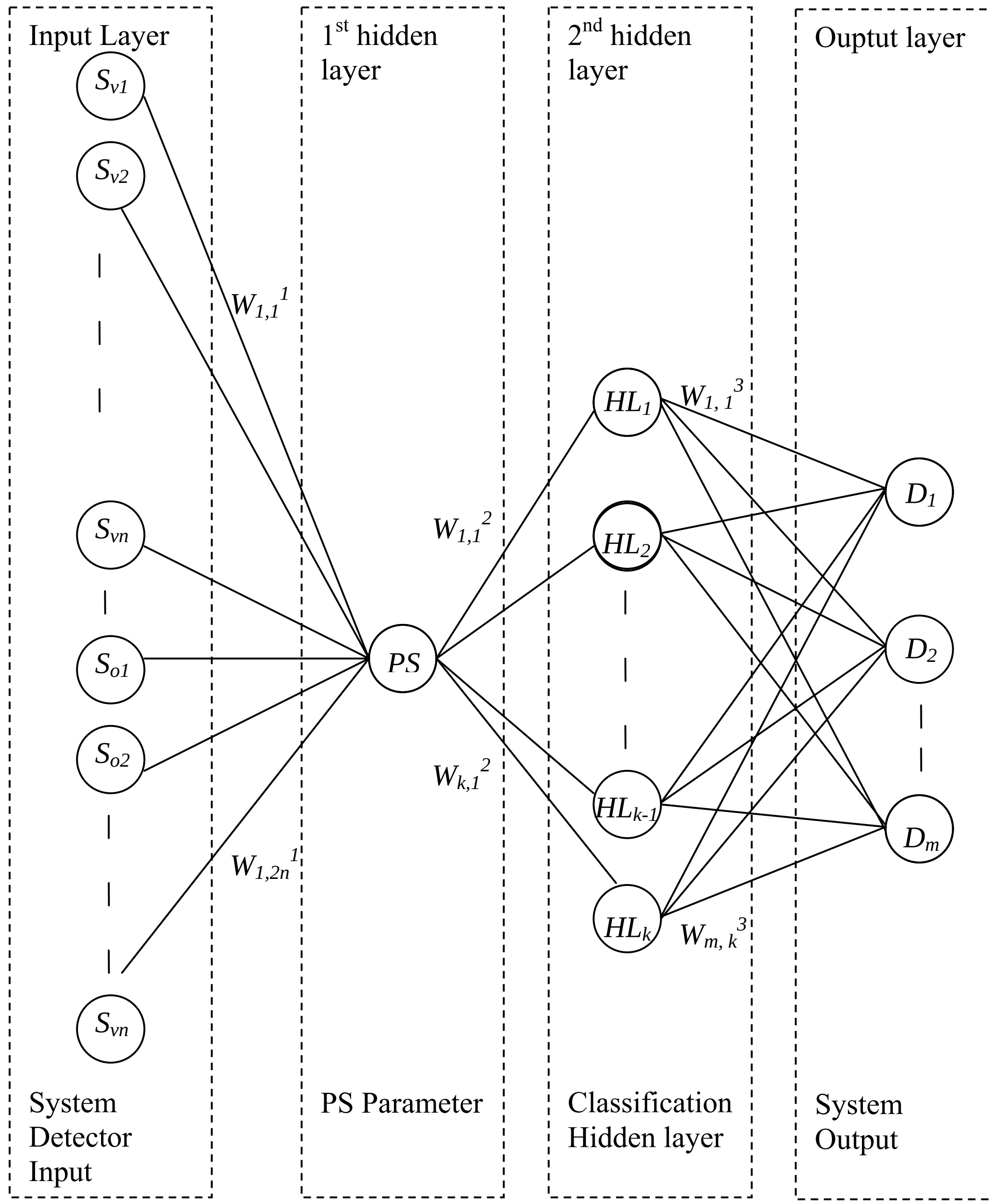


FIGURE 3.10 ANN architecture used for determination of TRPS weights

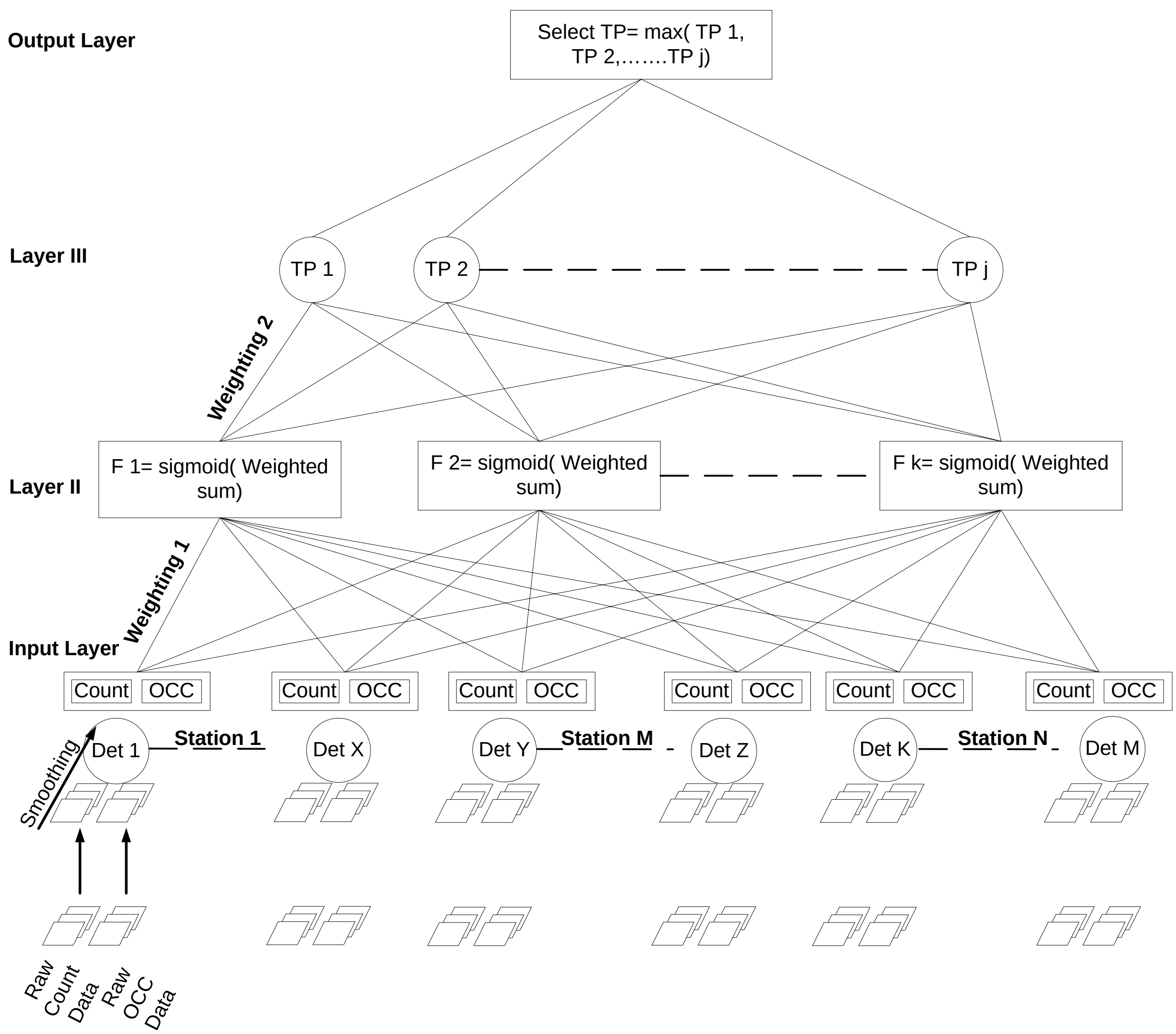
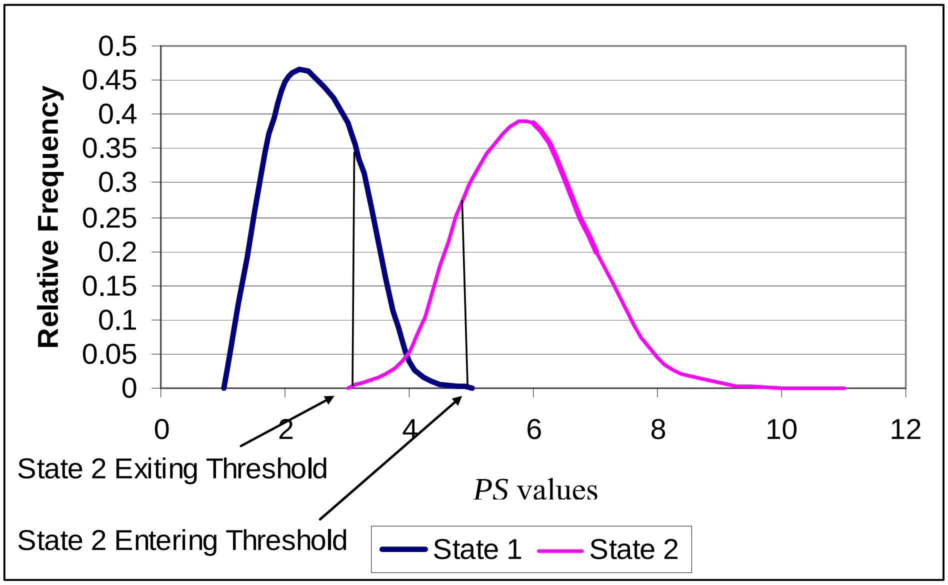


FIGURE 3.11 Proposed TRPS architecture

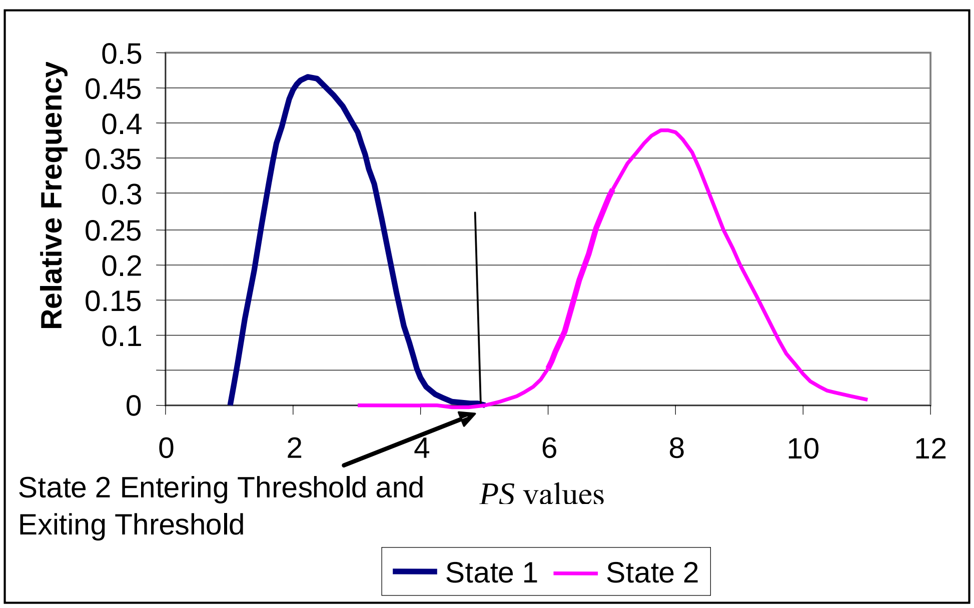
Determination of TRPS Thresholds

Using TRPS weights determined in the previous step, *PS* parameter values were calculated and plotted for all demand states. These plots were then analyzed to obtain the values of entering and exiting thresholds. Entering and exiting thresholds provide hysteresis control when there is overlap between two demand states. This concept is further illustrated with help of an example shown in Figure 3.12a. Entering threshold for State 2 is set such that the probability of occurrence of State 1 above this value approaches zero. Exiting threshold for State 2 is set such that the probability of occurrence of State 2 below this value approaches zero. In case of non-overlapping demand states, exiting and entering thresholds are the same as shown in Figure 3.12 b.

The next chapter illustrates the implementation of above methodology using data collected from Odem closed-loop arterial system in Texas.



a) States with mutual overlap



b) State without overlap

FIGURE 3.12 Conceptual illustrations of entering and exiting thresholds

CHAPTER IV

DATA ANALYSIS AND RESULTS

This chapter presents an illustrative example of implementation of the proposed methodology. The TRPS mode was set up for a closed-loop system at Odem, Texas using the proposed methodology. Figure 4.1 shows the location and ID's of the system detector as placed in the CORSIM network. These system detectors were placed 400 ft upstream of intersection to minimize the effect of braking and queuing in vicinity to the intersection.

DETERMINATION OF DEMAND STATES

Fifteen minute volume counts, from 2:45 P.M. December 1, 2002 to 2:00 P.M. December 9, 2002, were obtained from the Odem site. The approach volumes containing both normal day and holiday traffic were then divided by their respective saturation flow to give the flow ratio. These flow ratios were clustered using K-mean clustering. Sample flow rates and associated cluster are listed in Appendix B. Figure 4.2 shows the plot for Silhouette Width versus increasing number of clusters tried on the field data. Three clusters give the highest value of Silhouette Width of 0.73 (which indicates a strong structure based on Table 3.1). Three demand states low, medium and high were finalized. Figure 4.3 shows the silhouette plot for the three demand states.

There were few data points in medium demand having negative values which showed that there was some overlap between State 2 with rest of the states. But overall the clusters had high silhouette values and, hence, the clustering was acceptable.

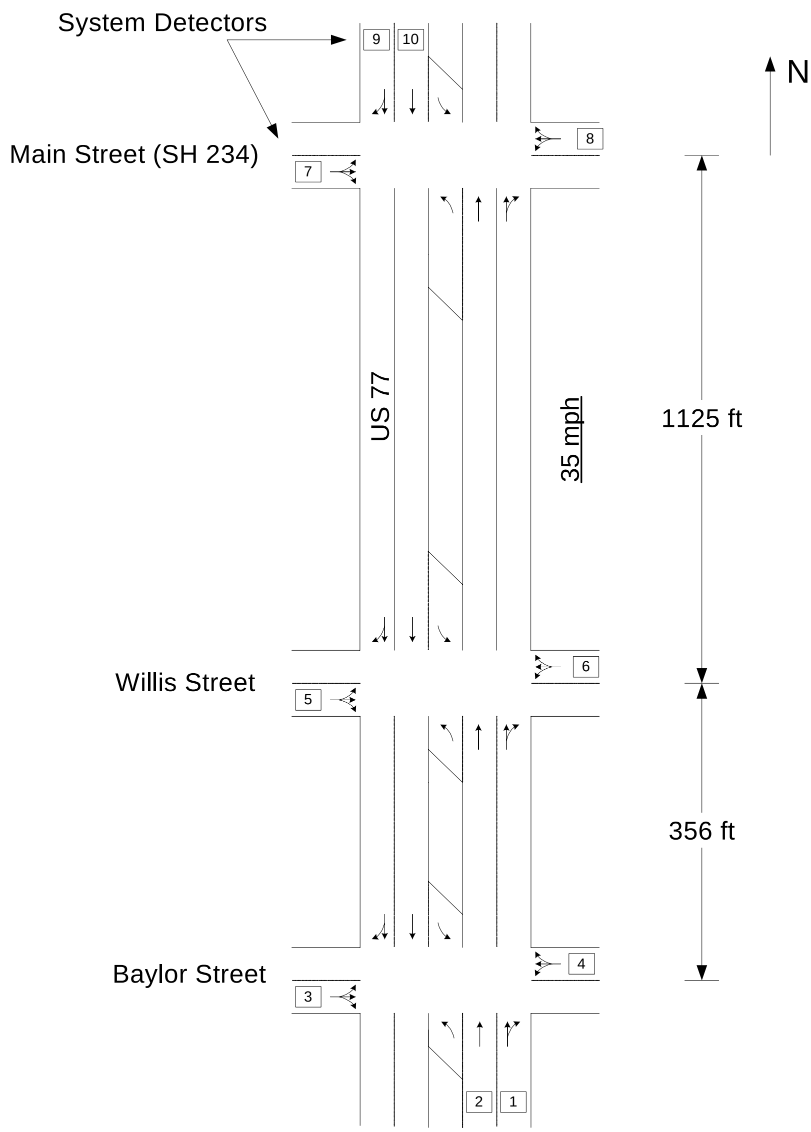


FIGURE 4.1 Odem closed-loop network

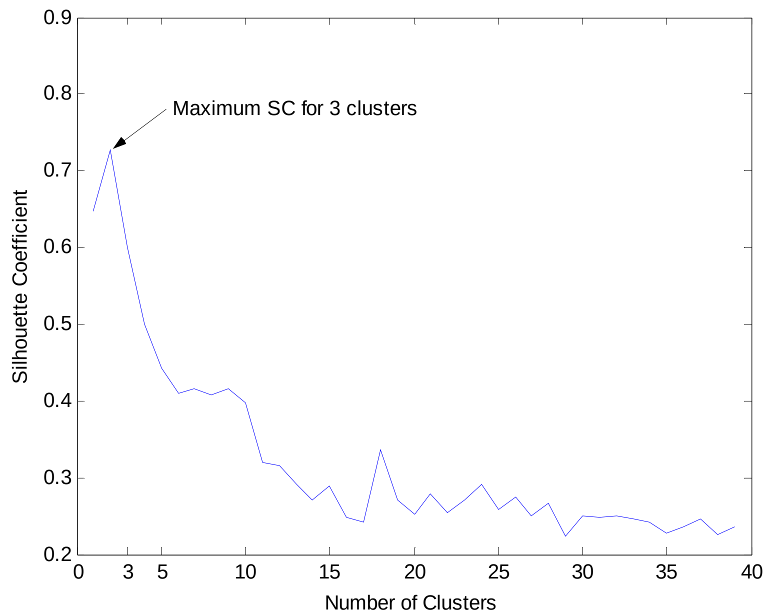


FIGURE 4.2 Silhouette widths versus number of clusters

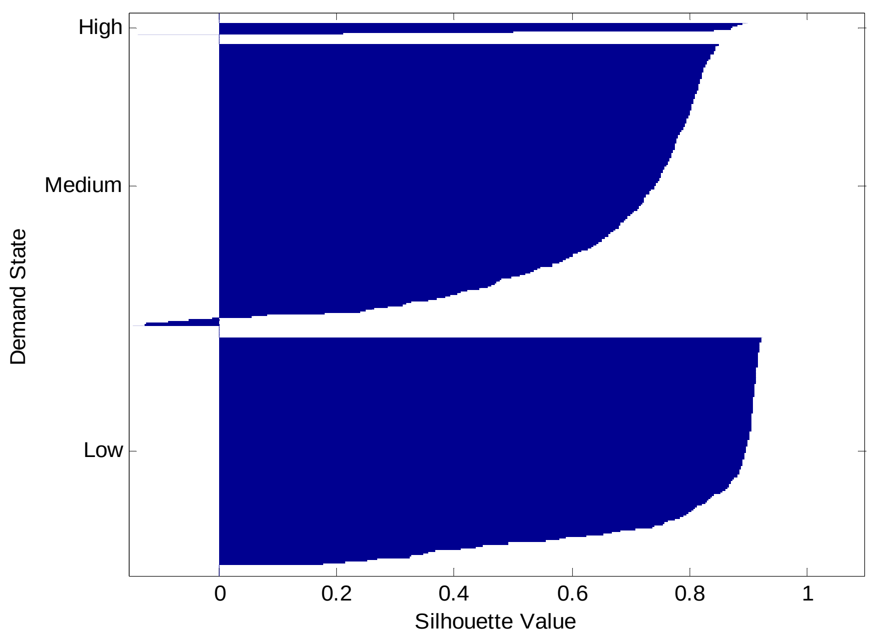


FIGURE 4.3 Silhouette plot for three demand states

TRAFFIC SIGNAL PLAN ASSIGNMENT

The next step was to assign the traffic signal plans to each of these states. SYNCHRO 5.0 was used for designing the traffic signal plans. 85% values of approach volume for each demand state was used as the design volume. Tables 4.1 and 4.2 show the design volumes and timing plan, as given by SYNCHRO, for the three demand states. Figure 4.4 presents a representative temporal variation of traffic demand on the Odem network and associated demand states. After the timing plan assignment to each demand state, CORSIM was used to simulate the network to obtain system detectors count and occupancy data. Simulations were performed for all demand state-plan combinations. Each scenario was simulated for 60 minutes. System Detectors' count and occupancy data were collected for every 15-minute intervals for all the simulated scenarios. These values were then used as training and validation data for neural network to determine TRPS detector weights as explained in next section.

TABLE 4.1 Traffic Volume States on Odem Network

Demand State	Intersection with US 77	Traffic Volumes (vph)					
		EB	NB		WB	SB	
		Total	Thru	Left	Total	Thru	Left
Low	Baylor	41	375	12	66	263	5
	Willis	37	397	21	36	292	5
	Main St	23	606	21	39	276	27
Medium	Baylor	83	640	29	108	611	17
	Willis	67	677	44	76	698	15
	Main St	38	1024	59	75	836	61
Heavy	Baylor	79	937	29	148	1509	24
	Willis	74	1034	53	98	1728	11
	Main St	38	1516	59	73	1977	54

TABLE 4.2 Designed Timing Plans for Odem Network

Demand State	Intersection with US 77	Cycle (sec)	Split (sec)						Offset (sec)
			SBL	NB	EB	NBL	SB	WB	
Low	Baylor	50	10	16	12	10	16	12	0
	Willis	50	10	16	12	10	16	12	45
	Main St	50	9	19	11	9	19	11	23
Medium	Baylor	55	9	22	11	9	22	13	0
	Willis	55	9	24	11	9	24	11	2
	Main St	55	8	27	10	8	27	10	29
Heavy	Baylor	100	8	60	13	8	60	19	0
	Willis	100	8	66	12	8	66	14	93
	Main St	100	9	70	10	8	71	11	78

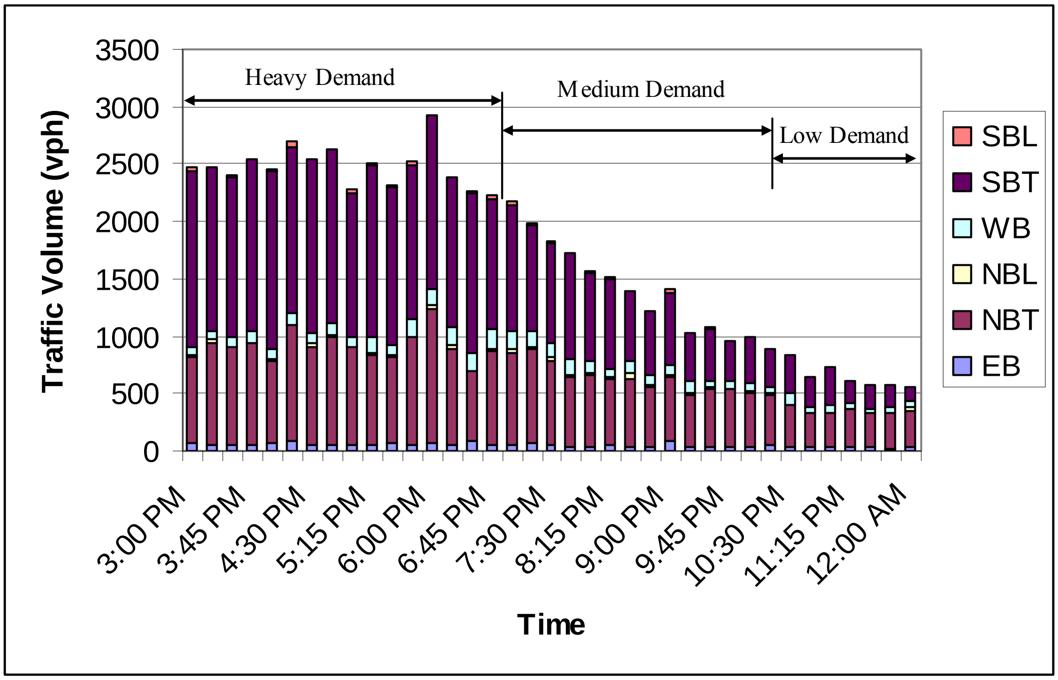


FIGURE 4.4 Assignment of different volume scenarios to demand states

DETERMINATION OF DETECTOR WEIGHTS

Figure 4.5 presents the neural network architecture used for determination system detector weights. The MATLAB neural network toolbox was used to train the artificial neural network. Count and occupancy data obtained from 10 system detectors were scaled to a scale of 0 to 1 and provided as an input to the neural network. The Count scaling factor to be input in the controller was 30, which is the maximum number of arriving vehicles expected per approach per minute (that is 2 second headway between vehicles). The Occupancy scaling factor of 100 was used. The 3-dimensional vector output was compared with target value (which is the demand state represented in vector form). For Odem network, demand states single *PS* parameter was found to be sufficient for significant (95%) classification accuracy.

Table 4.3 lists the detector weights to be used in the controller. Note that present controllers do not have the provision for having a negative weight. So the weights provided by ANN have been increased by the value of most negative weight. This option reduces the efficiency of classification.

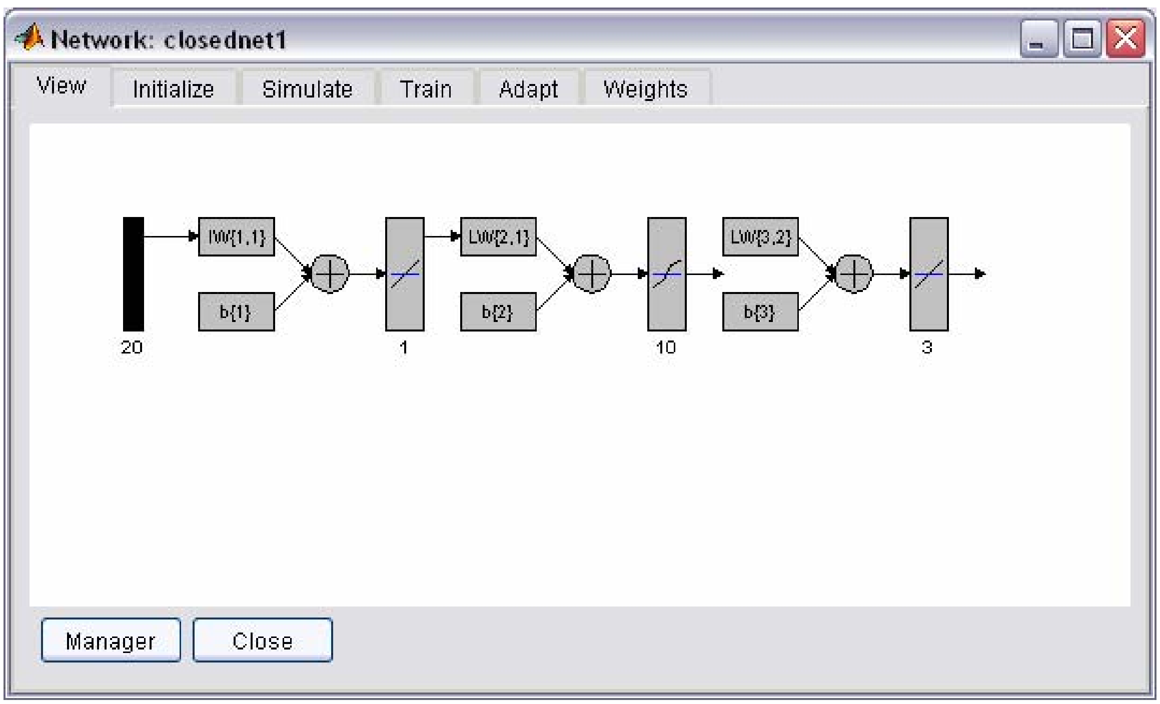


FIGURE 4.5 ANN architecture used for determination system detector weights

TABLE 4.3 System Detector Weights

Detector ID	Detector Weights	
	Count	Occupancy
1	9	0
2	7	7
3	2	4
4	5	3
5	4	6
6	8	6
7	8	9
8	1	12
9	7	2
10	3	9

DETERMINATION OF ENTERING AND EXITING THRESHOLDS

Entering and exiting thresholds were decided by finding the maximum and minimum values of the *PS* parameter occurring for each of the demand states. Table 4.4 gives the exiting and entering thresholds for three demand states. Figure 4.6 presents the plot of *PS* values obtained for each state. Relative frequency for all the demand states were plotted and shown in Figure 4.7

TABLE 4.4 Entering and Exiting Threshold for Demand States

Demand State	Threshold	
	Entering	Exiting
Low	0	0
Medium	4	3
High	9	8

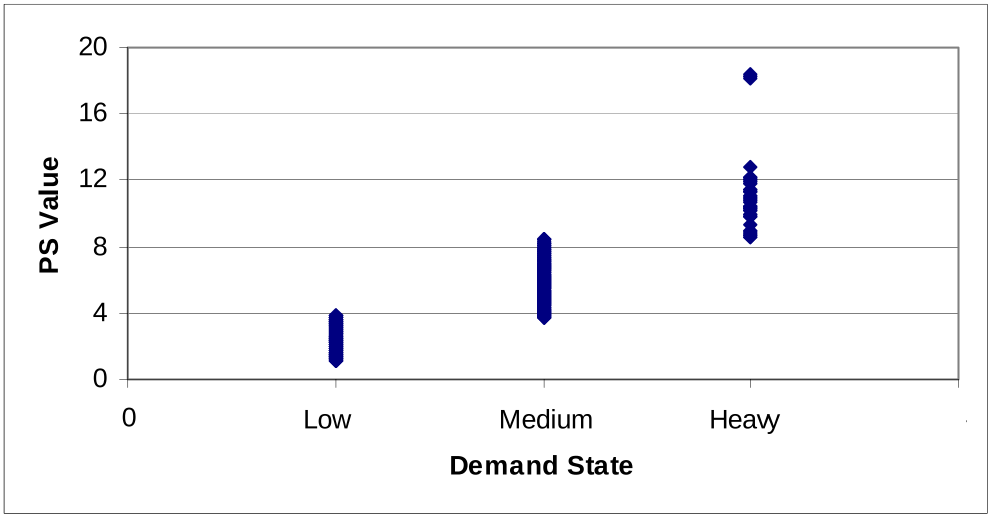


FIGURE 4.6 Plot of *PS* values for demand states

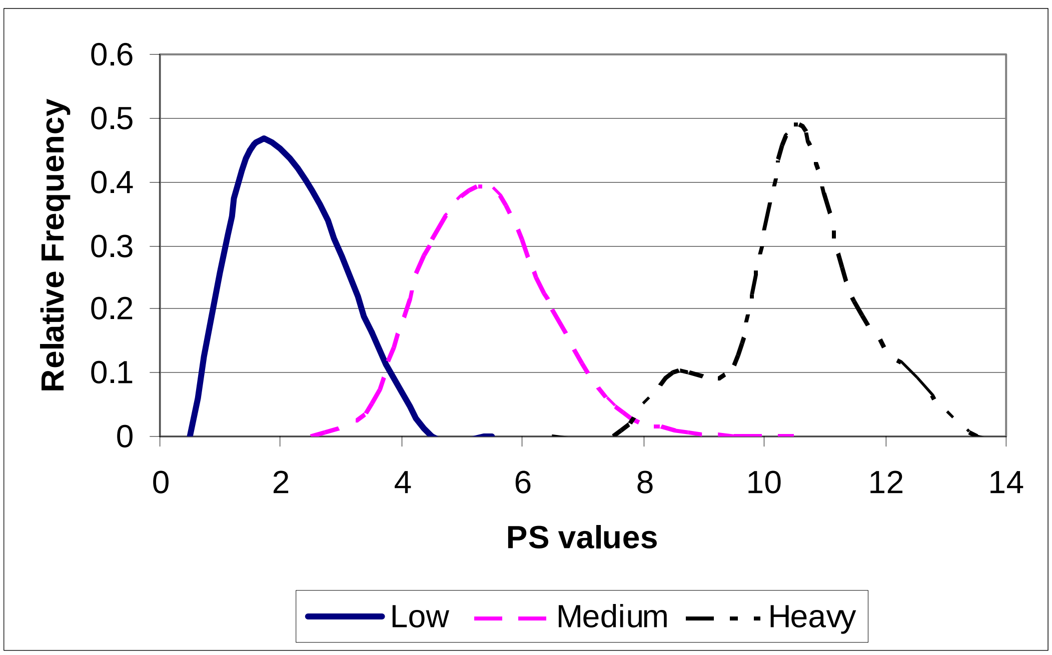


FIGURE 4.7 Plot of relative frequency versus demand states

RESULTS

The system detector weights and thresholds obtained using the training data provide classification accuracy of 94.40% on the training data and 94.38% on the validation data. The misclassification error only consisted of the demand state being misclassified as its nearest neighbor. This implies that the low demand state was never misclassified as the high demand state and vice versa. Some of the misclassification is due to controller design. It was observed that if negative system detector weights were allowed, classification accuracy increased to 96.40 %. The efficiency of the Bayesian-based classification proposed by Abbas and Sharma *(9)* on the same data set was 91.82%.

As stated earlier, the present architecture TRPS classification can be slightly modified to enhance the classification power of TRPS mode. Training the multi-layer perceptron ANN as shown in Figure 4.8 resulted in the classification accuracy being increased to 98.00% using the validation data.

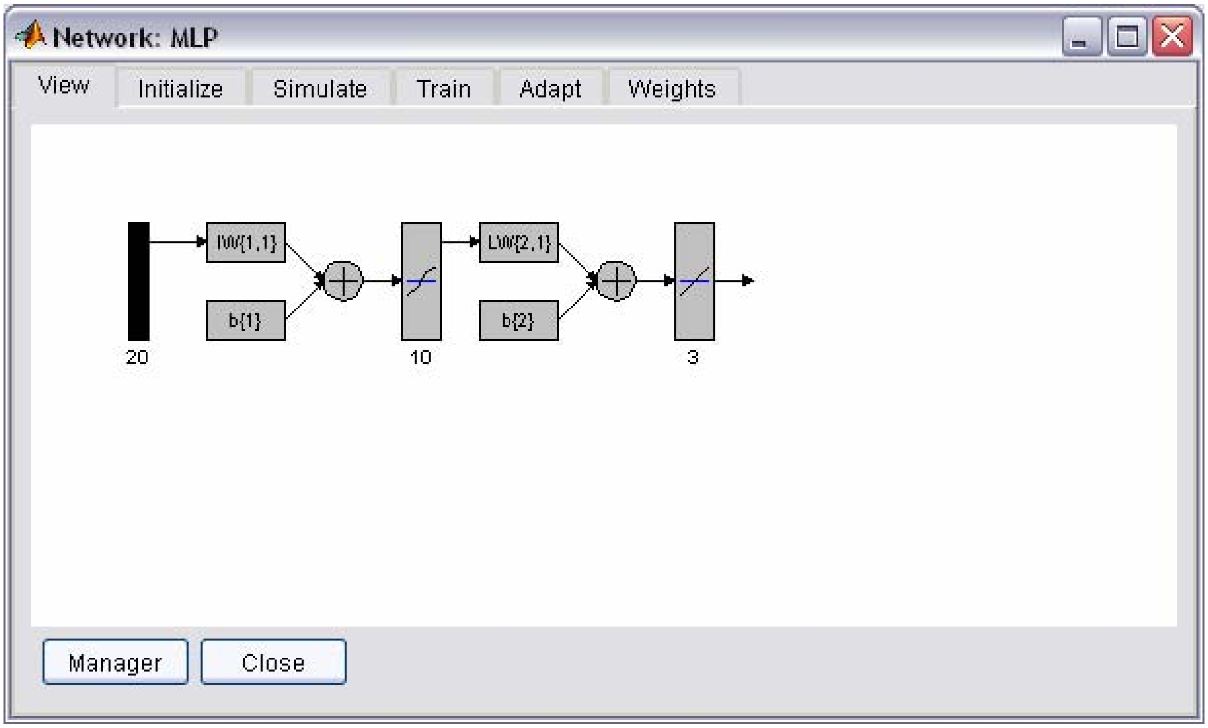


FIGURE 4.8 Proposed multi-layer perceptron ANN algorithm for TRPS mode

Table 4.5 lists the classification accuracy obtained by each methodology tested and the standard error. The standard error of the mean for binomial population proportion *(0)* is given by the following equation:

$$\sigma_{\hat{p}} = \sqrt{\frac{pq}{n}} \tag{5.1}$$

where:

$\sigma_{\hat{p}}$ = standard error of mean

p = probability of correct classification

q = probability of miss-classification

n = sample size

It was observed that classification accuracy for each methodology was significantly different from any other methodology with 95 percent level of confidence.

TABLE 4.5 Classification Accuracy and Standard Error for Different Methodologies Tested

Methodology	Population (*n*)	Correct Classification	Classification Accuracy	Standard Error	95% Confidence Interval
Bayesian-based	4566	4187	0.92	0.004	[0.909,0.925]
ANN based on Present Controller	4566	4309	0.94	0.003	[0.937,0.95]
Allowing Negative Weights in Present Controller	4566	4410	0.97	0.003	[0.961,0.971]
Proposed Architecture	4566	4473	0.98	0.002	[0.976,0.984]

Figure 4.9 presents the classification accuracy achieved on the validation data set using different methodologies. An increase in benefits is observed as we move from left to right in Figure 4.9. Each of these methodologies involved certain cost of implementation. Both Bayesian-based and MLP-based controller architecture, approaches required a statistical toolbox like MATLAB or SAS for their

implementation. In the next approach, which used negative weights, the present controller architecture need not be significantly changed. It would only require upgrading the present architecture to accept negative values for weight. The last methodology, which used a simple multi-layer perceptron architecture, required a major change in present controller architecture. The best operation was achievable using this approach. The benefits from this approach would be higher with an increase in complexity and dimensionality of the data.

In the light of above discussions, ANN based methodology using the present controller may prove most beneficial for the least cost for a simple network with low variability in demand. As the complexity of the closed-loop system increases in terms of dimensionality and variability of demand, more costly methodology using a modified controller should be considered as the benefits of a more flexible an accurate controller increase. The cost benefit analysis for implementing different methodologies was not done in this thesis, but is recommended to be done for each site.

Based on the above discussion, it can be concluded that TRPS mode can be efficiently set up using a neural network approach. TRPS mode architecture may be slightly modified to achieve even higher benefits.

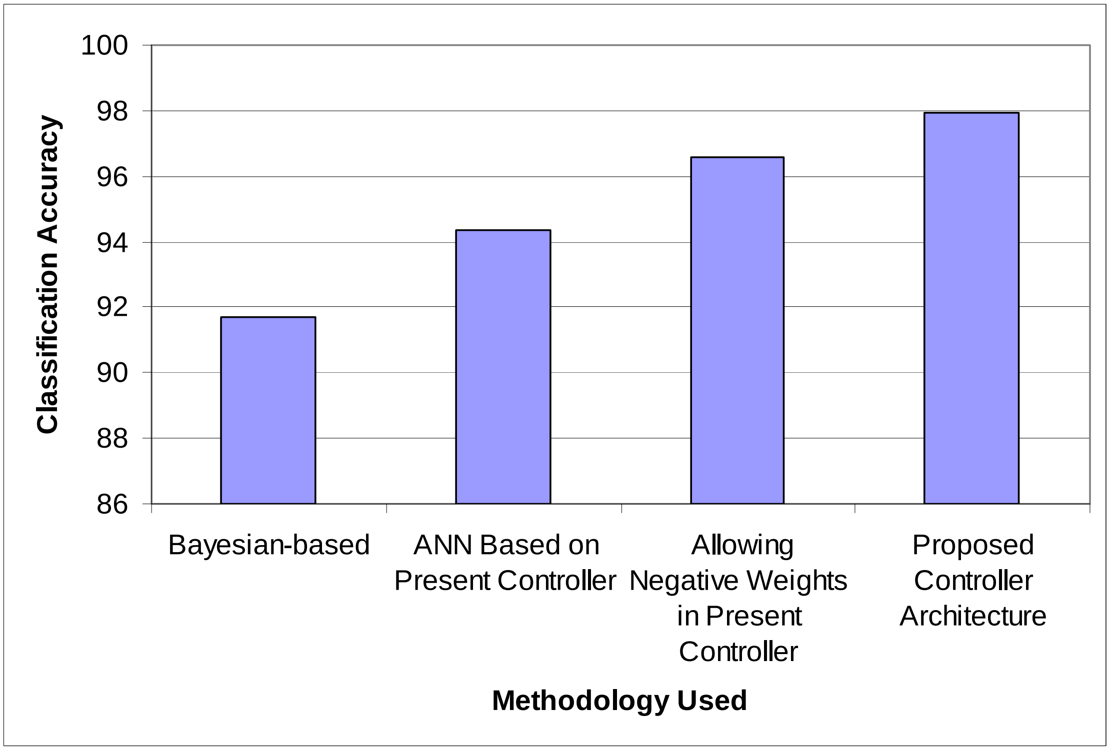


FIGURE 4.9 Classification accuracy for different classification approaches

CHAPTER V

CONCLUSIONS AND RECOMMENDATIONS

A methodology for determination of TRPS factors and threshold has been developed in this research. A three step approach was used, namely:

1. Identification of demand states using K-means clustering.
2. Assignment of signal timing plan to each demand state using SYNCHRO 5.0.
3. Determination of TRPS weights and threshold using ANN.

Essentially, this thesis provides a step-by-step description on how to set up TRPS mode in present traffic controllers after collecting the traffic volume data from the field. The methodology presented in this research is also compared against the Bayesian-based approach. Misclassification rate among the demand states is used as the measure of comparison. This thesis also recommends a set of changes in TRPS mode architecture of present traffic controllers to increase the accuracy of classification of demand states.

The scope of this research includes closed-loop arterial traffic control systems. This methodology can also extend to traffic networks, but this has not been discussed in this thesis.

CONCLUSIONS

TRPS mode can be efficiently configured using the methodology provided in this thesis. An efficiency of 94.38% was achieved for the Odem closed-loop system in Texas. This efficiency increased to 98% with some modification, as described earlier, in present traffic controller architecture.

The research methodology is transferable to any closed-loop arterial system. In contrast to a TOD mode, TRPS mode provided has built-in intelligence to adapt to changing traffic demand. TRPS mode can adapt its schedule based on the measured conditions.

RECOMMENDATIONS

The methodology provided in this research, though effective, involves a great deal of statistical analysis and complex mathematical operation for training the neural network. MATLAB was used for these analyses. This toolbox might not be available for the traffic engineer trying to setup a TRPS mode. For a wider usage of the proposed methodology and to reduce the time for implementation, it is highly recommended that automated software with a user friendly interface be developed.

Modification of TRPS mode architecture in present traffic controllers is also recommended. Figure 3.11 provided a sketch of the proposed architecture of TRPS mode for future controllers. The proposed architecture has two sets of weighing factors and the timing plans are chosen from the lookup table corresponding to node of Layer

II having the highest value. There would be no threshold factors. Some of the important points regarding connection weights are listed below:

- Negative system detector weights should be allowed
- Fractional values for system detector weights should be allowed

The proposed architecture will significantly reduce the misclassification rates associated with TRPS mode and the benefits of the proposed architecture will increase as traffic demand patterns increase in complexity and dimensionality.

REFERENCES

1. Henry, R.D., R.A. Ferlis, and J.L. Kay. *Evaluation of UTCS Control Strategies —Executive Summary*. Report No. FHWA-RD-76-149. FHWA, U.S. Department of Transportation, August 1976.

2. Kay, J.L., J.C. Allen, and J.M. Bruggeman. *Evaluation of First Generation UTCS/BPS Control Strategy —Executive Summary.* Report No. FHWA-RD-75-26. FHWA, U.S. Department of Transportation, March 1975.

3. Kay, J., R. Henry, and S. Smith. *Locating Detectors for Advanced Traffic Control Strategies: Handbook.* Report FHWA-RD-75-91. FHWA, U.S. Department of Transportation, 1975.

4. Woods, D.L., and N.J. Rowan. *Detector Location for Computerized Arterial Street Sampling Detectors.* Report FHWA/TX-95/1392-6. Texas Transportation Institute, College Station, Texas, 1995.

5. Balke, K.N., R.S. Keithireddipalli, and C.L. Brehmer. *Guidelines for Implementing Traffic Responsive Mode in TxDOT Closed-Loop Traffic Signal Systems*. Research Report 2929-3F. Project No. 7-2929. Texas Transportation Institute, College Station, Texas, August 1997.

6. *Naztec Operations Manual for Closed-Loop Master Controller*. Naztec, Inc., Sugarland, Texas, 2002.

7. *Master Controller Software*. Naztec, Inc., Sugarland, Texas, 2002.

8. Abbas, M. M., N. A. Chaudhary, A. Sharma, S. P. Venglar, and R. J. Engelbrecht. *Methodology for Determination of Optimal Traffic Responsive Plan Selection Control Parameters.* Research Report 4421-1. Texas Transportation Institute, College Station, Texas, September 2003.

9. Abbas, M. M., and A. Sharma. A Robust Bayesian Approach for Setting Traffic Responsive Control System Parameters and Thresholds (accepted). *Transportation Research Record*, 2004.

10. Huberty, C.J. *Applied Discriminant Analysis*. John Wiley & Sons, Inc. New York, 1994.

11. Haykin, S. *Neural Networks: A Comprehensive Foundation.* Second Edition. Pearson Education Inc., Upper Saddle River, New Jersey, 1999.

12. Gratner, N.H. A Prescription for Demand-Responsive Urban Traffic Control. *Transportation Research Record*, 881, 1982, pp 73-76.

13. Meyer, M.D. *A Toolbox for Alleviating Traffic Congestion and Enhancing Mobility*. Institute of Transportation Engineers, Washington, D.C., 1997.

14. Wagner, F. A. *Overview of the Impacts and Costs of Traffic Control Systems Improvements*. U.S. DOT FHWA, Washington, D.C., March 1980.

15. Fambro, D. B., S. M. Sangineni, C. A. Lopez, S. R. Sunkari, and R. T. Barnes. *Benefits of the Texas Traffic Light Synchronization (TLS) Grant Program II*. Texas Transportation Institute, College Station, Texas, 1995.

16. Image from the article, *Traffic Signals and Traffic Flow: How They Work Together*, http://www.trafficpd.com/Images/Publications.

17. Nelson, E. J., M.M. Abbas, G. E. Shoup, and D. M. Bullock. *Development of Closed-Loop System Evaluation Procedures*. FHWA/IN/JTRP. Federal Highway Administration, McLean, Virginia, May 2000.

18. Hadi, M., and K.G. Courage. *Development of Optimal Threshold Parameters for Traffic Responsive Control in Arterial Signal System*. Report No. UTC.UF.268.4, U.S. Department of Transportation, Washington, D.C., January 1990.

19. *TRANSYT-7F User's Manual*, Release 6, Federal Highway Administration, U.S. Department of Transportation, Washington, D.C., 1988.

20. *SYNCHRO 5.0 Traffic Signal Timing Software and Synchro User Guide*, Trafficware Corporation, Albany, California 2000.

21. *CORSIM User's Manual*, ITT Systems & Sciences Corporation. FHWA, U.S. Department of Transportation, Washington, D.C., March 1998.

22. Kaufman, L., and P. J. Rousseeuw. *Finding Groups in Data An Introduction to Cluster Analysis*. John Wiley & Sons Inc. New York, 1990.

23. Hartigan, J.A.. *Clustering Algorithms*. John Wiley & Sons Inc. New York, 1974.

24. Marques, de Sa J.P. *Pattern Recognition Concepts, Methods and Applications*. Springer, New York, 2001.

25. *Highway Capacity Manual*. TRB Special Report 209, Third Edition, Transportation Research Board, Washington, D.C., 1994.

26. *MATLAB 6.5*., The Mathworks. Inc., Natick, Massachusetts, 2003.

27. Rousseeuw, P.J. Silhouettes: A Graphical Aid to the Interpretation and Validation of Cluster Analysis. *Journal of Computational and Applied Mathematics*, Vol. 20, 1987, pp 53-65.

28. *PASSER II User Guide,* Developed by Texas Transportation Institute of the Texas A&M University System, College Station, Texas, 1991.

29. Bishop, C.M.. *Neural Networks for Pattern Recognition.* Clarendon Press, Oxford, 1995.

30. Nigrin, A. *Neural Networks for Pattern Recognition*. MIT Press, Cambridge, Massachusetts, 1993.

31. Cherkassky, V., and F. Mulier. *Learning from Data Concepts Theory and Methods.* John Wiley & Sons, Inc. New York, 1998.

32. Freund, J.E. *Mathematical Statistics.* Prentice-Hall, Inc. Upper Saddle River, New Jersey, 1992.

APPENDIX A

EXAMPLE FOR K-MEANS CLUSTERING

Here is a simple example of K-means clustering. Suppose we have a two dimensional data set with 5 points as shown in Table A-1. The data points are plotted and shown in Figure A-1.

TABLE A-1 Example Data Set

Data Point	***X***	***Y***
P1	22	21
P2	19	20
P3	18	22
P4	1	3
P5	4	2

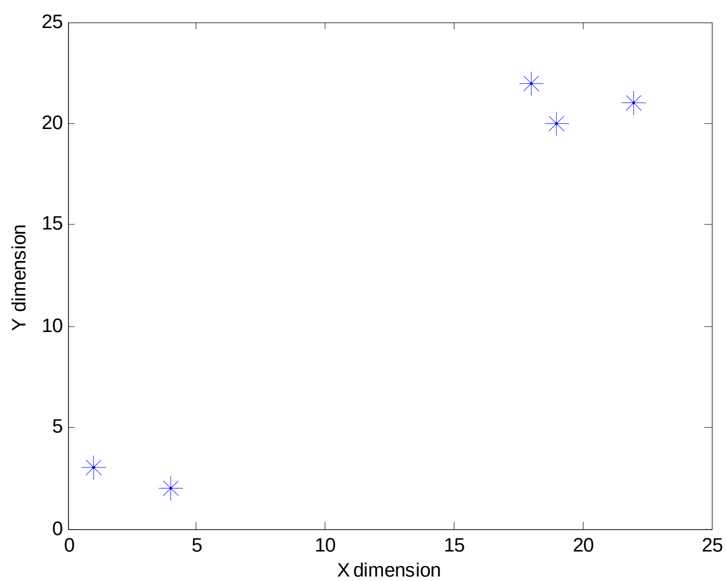


FIGURE A-1 Plot of example data set

The initialization step of the K-Means algorithm involves:

1. Assigning a k value.
2. Designating cluster centroids for each k (i.e. for each cluster).

The numbers of clusters in which the data points are classified are determined using average silhouette width. In this case we use the value of k as 2. So we would try to classify the data in two clusters. Cluster centroids are randomly selected. Suppose we choose *P3* and *P5* as the initial cluster centroids for clusters Class *A* and Class *B*. The next step is to pick a data points to their closest cluster centers. After assigning the point to a cluster, centroid values for the cluster are recalculated. Table A-2 shows these steps for the 5 points. Every data points are assigned to the class having closest centroid.

TABLE A-2 Steps for K-Means Clustering

<table>
<tr><td colspan="5">Step 1</td></tr>
<tr><td>Iteration</td><td>Class A Membership</td><td>Class B Membership</td><td>Centeroid Class A</td><td>Centroid Class B</td></tr>
<tr><td>1</td><td>P3</td><td>P5</td><td>[18,22]</td><td>[4,2]</td></tr>
<tr><td colspan="5">Step 2</td></tr>
<tr><td rowspan="2">Data Point</td><td colspan="2">Distance from</td><td colspan="2" rowspan="2">Assignment</td></tr>
<tr><td>Centroid A</td><td>Centroid B</td></tr>
<tr><td>P1</td><td>4.1</td><td>26.2</td><td colspan="2">Class A</td></tr>
<tr><td colspan="5"></td></tr>
<tr><td>Iteration</td><td>Class A Membership</td><td>Class B Membership</td><td>Centroid Class A</td><td>Centroid Class B</td></tr>
<tr><td>1</td><td>P1,P3</td><td>P5</td><td>[20,21.5]</td><td>[4,2]</td></tr>
<tr><td colspan="5">Step 3</td></tr>
<tr><td rowspan="2">Data Point</td><td colspan="2">Distance from</td><td colspan="2" rowspan="2">Assignment</td></tr>
<tr><td>Centroid A</td><td>Centroid B</td></tr>
<tr><td>P2</td><td>1.8</td><td>23.4</td><td colspan="2">Class A</td></tr>
<tr><td colspan="5"></td></tr>
<tr><td>Iteration</td><td>Class A Membership</td><td>Class B Membership</td><td>Centroid Class A</td><td>Centroid Class B</td></tr>
<tr><td>1</td><td>P1,P2,P3</td><td>P5</td><td>[19.7,21]</td><td>[4,2]</td></tr>
<tr><td colspan="5">Step 4</td></tr>
<tr><td rowspan="2">Data Point</td><td colspan="2">Distance from</td><td colspan="2" rowspan="2">Assignment</td></tr>
<tr><td>Centroid A</td><td>Centroid B</td></tr>
<tr><td>P4</td><td>26.0</td><td>4.5</td><td colspan="2">Class B</td></tr>
<tr><td colspan="5"></td></tr>
<tr><td>Iteration</td><td>Class A Membership</td><td>Class B Membership</td><td>Centroid Class A</td><td>Centroid Class B</td></tr>
<tr><td>1</td><td>P1,P2,P3</td><td>P4,P5</td><td>[19.7,21]</td><td>[2.5,2.5]</td></tr>
</table>

The next step is to check whether the points are correctly classified. Table A-3 shows the distances of each point to the centroids of class *A* and *B*. In this case there exists a correct classification in the first try. So we don't iterate further.

TABLE A-3 Euclidean Distance of Each Point to the Centroids

Data point	**Distance from**		**Assignment**
	Centroid *A*	**Centroid *B***	
P1	2.3	26.9	*A*
P2	1.2	24.1	*A*
P3	2.0	24.9	*A*
P4	26.0	1.6	*B*
P5	24.6	1.6	*B*

TABLE A-4 Within-Cluster Sum of Squares

Assigned Class	**Points**	**Square of Distance from Class Centroid**
Class *A*	*P1*	5.44
	P2	1.44
	P3	3.77
Class *B*	*P4*	2.5
	P5	2.5
Within-Cluster Sum of Squares (E)		15.65

Table A-4 shows the within-cluster sum of square (*E*) for the data set. After the first iteration one would normally reassign data points to their closest cluster centroids. The initialization in this case was good so no reassignment was necessary. Therefore, after the first iteration the cluster centroids do not change at all and so a stable *E* value (15.65) has been reached.

EXAMPLE FOR SILHOUETTE PLOT

The calculation and plotting of silhouette plot is presented here with the help of a simple example. This example is the continuation K-mean clustering example. After classification of data points, as done in previous section, Silhouette value is calculated for a data point.

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}$$

where:

$s(i)$ = silhouette value.

$a(i)$ = average dissimilarity of i to all other objects of class A.

$b(i)$ = average dissimilarity of i to all other objects of nearest cluster other than A.

For data point P1 the $a(1)$ can be calculated as follows:

$$a(1) = 1/2 * \left[\sqrt{(19-22)^2 + (20-21)^2} + \sqrt{(18-22)^2 + (22-21)^2}\right] = 13.5$$

$$b(1) = 1/2 * \left[\sqrt{(1-22)^2 + (3-21)^2} + \sqrt{(4-22)^2 + (2-21)^2}\right] = 725$$

$$\Rightarrow s(i) = \frac{725 - 13.5}{725} = 0.9814$$

Table A-5 tabulates the silhouette value for each point. A silhouette for cluster A is a plot of $s(i)$ ranked in descending order. Silhouette plot for the cluster A and B is shown in Figure A-2.

TABLE A-5 Silhouette Values for Each Data Point

Cluster	Data Point	Silhouette Value
A	*P1*	0.98709
	P2	0.98234
	P3	0.98521
B	*P4*	0.98361
	P5	0.98709

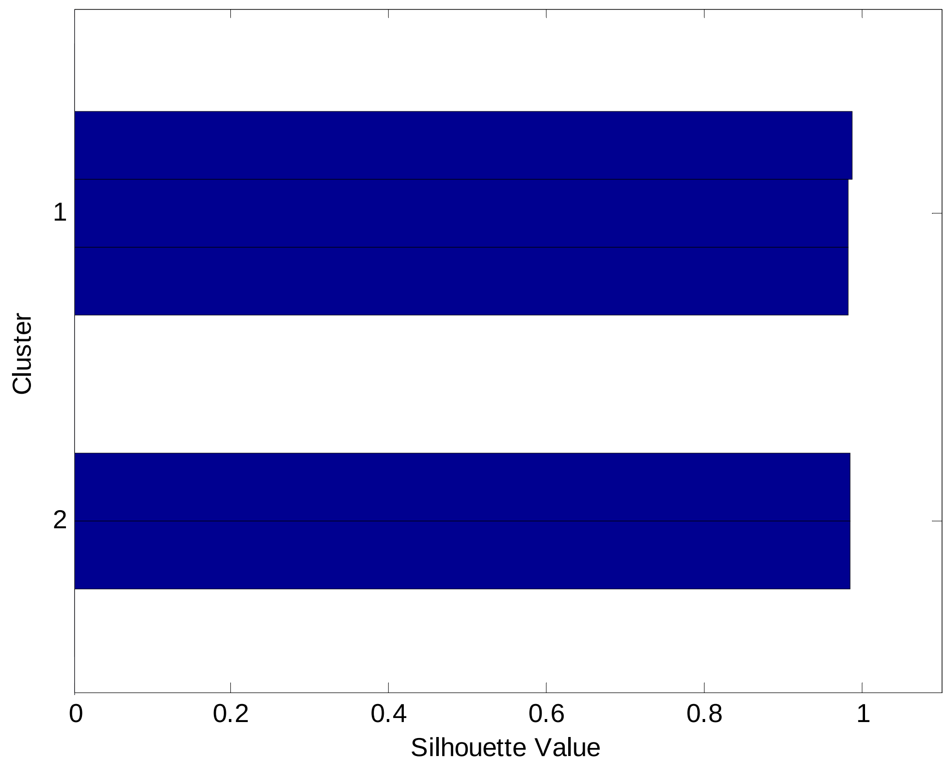


FIGURE A-2 Silhouette plot of example data set

APPENDIX B

TABLE B-1 Sample Flow Ratio for High Demand Cluster

Time	Inter-section with US 77	Flow Ratio						Cluster
		NB		SB		EB	WB	
		Thru	Left	Thru	Left	Total	Total	
3:00 PM	Baylor	0.229	0.008	0.483	0.016	0.034	0.043	High
	Willis	0.262	0.015	0.521	0.003	0.027	0.027	
	Main St	0.332	0.016	0.603	0.013	0.014	0.034	
3:15 PM	Baylor	0.272	0.016	0.452	0.000	0.026	0.043	High
	Willis	0.366	0.000	0.536	0.000	0.032	0.051	
	Main St	0.412	0.000	0.468	0.013	0.014	0.025	
3:30 PM	Baylor	0.258	0.000	0.439	0.013	0.032	0.043	High
	Willis	0.298	0.000	0.499	0.000	0.020	0.087	
	Main St	0.398	0.000	0.492	0.027	0.008	0.025	

TABLE B-2 Sample Flow Ratio for Medium Demand Cluster

Time	Inter-section with US 77	Flow Ratio						Cluster
		NB		SB		EB	WB	
		Thru	Left	Thru	Left	Total	Total	
8:30 PM	Baylor	0.186	0.013	0.241	0.013	0.022	0.058	Medium
	Willis	0.164	0.021	0.260	0.000	0.039	0.029	
	Main St	0.356	0.000	0.197	0.013	0.008	0.016	
8:45 PM	Baylor	0.175	0.011	0.248	0.005	0.025	0.037	Medium
	Willis	0.171	0.036	0.249	0.005	0.029	0.036	
	Main St	0.371	0.016	0.184	0.013	0.014	0.016	
9:00 PM	Baylor	0.181	0.026	0.191	0.003	0.019	0.057	Medium
	Willis	0.175	0.029	0.213	0.000	0.019	0.026	
	Main St	0.357	0.010	0.152	0.018	0.009	0.021	

TABLE B-3 Sample Flow Ratio for Low Demand Cluster

Time	Inter-section with US 77	Flow Ratio						Cluster
		NB		SB		EB	WB	
		Thru	Left	Thru	Left	Total	Total	
8:30 PM	Baylor	0.089	0.003	0.065	0.000	0.019	0.020	Low
	Willis	0.080	0.011	0.070	0.000	0.015	0.029	
	Main St	0.201	0.003	0.044	0.005	0.009	0.013	
8:45 PM	Baylor	0.093	0.005	0.060	0.000	0.011	0.022	Low
	Willis	0.098	0.010	0.074	0.000	0.015	0.028	
	Main St	0.170	0.002	0.066	0.004	0.009	0.014	
9:00 PM	Baylor	0.100	0.011	0.038	0.001	0.015	0.035	Low
	Willis	0.087	0.013	0.042	0.000	0.012	0.020	
	Main St	0.191	0.008	0.048	0.001	0.009	0.019	

VITA

Anuj Sharma

500, Russell Street # 9, West Lafayette, IN-47906
Ph: (979) 220-9007 (h) **E-mail:** anujpals@yahoo.co.in

EDUCATION:

- Texas A&M University**,** College Station, Texas
  Master of Science (Transportation Engineering) Graduation Date: Aug 2004.

- Regional Enginnering College, Rourkela, INDIA.
  Bachelor of Engineering (Civil Engineering)

WORK EXPERINCE:

- **Texas Transportation Institute, Texas A&M University System**.
  Graduate Research Assistant. January 2003-Present

- **Engineering Technology and Industrial Distribution, Texas A&M University**
  Student Worker. September 2002-December2002

- **Cognizant Technology Solutions, Calcutta**
  Programmer Analyst. September 2001-July 2002.

- **Indian Institute of Technology, Madras**
  Project Assistant: May 2000-July 2000.

PROJECTS AND PAPERS:

- *Abbas M.*, *Sharma A.*, A Robust Bayesian Approach for Setting Traffic Responsive Control System Parameters and Thresholds (accepted), Transportation Research Record.
- *M.M. Abbas, N.A. Chaudhary, A. Sharma, S.P. Venglar, R.J. Engelbrecht.* Methodology for Determination of Optimal Traffic Responsive Plan Selection Control Parameters, Research Report 0-4421-1. Texas Transportation Institute, College Station, Texas. February 2004.